

Composing IoT Wireless Energy Services

AMANI ABUSAFIA



THE UNIVERSITY OF
SYDNEY

Supervisor: Athman Bouguettaya

A thesis submitted in fulfilment of the requirements for the degree of
Doctor of Philosophy

The research reported in this thesis was supported by the award of a Research
Training Program scholarship to the PhD Candidate

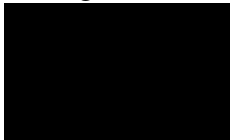
School of Computer Science
Faculty of Engineering
The University of Sydney
Australia

2024

Statement of Originality

This is to certify that, to the best of my knowledge, the content of this thesis is my own work. This thesis has not been submitted for any degree or other purposes. I certify that the intellectual content of this thesis is the product of my own work and that all the assistance received in preparing this thesis and sources have been acknowledged.

Signature:



Amani Abusafia
School of Computer Science
Faculty of Engineering
The University of Sydney
13th January 2024

Authorship Attribution Statement

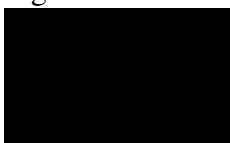
The main results presented in this thesis were published as follows:

- Chapter 3 of this thesis is published as [1]. I designed the study, analyzed the data, and wrote the drafts of the manuscript.
- Chapter 4 of this thesis is published as [2]. I designed the study, analyzed the data, and wrote the drafts of the manuscript.
- Part of chapter 5 of this thesis is published as [3]. I co-designed the study and co-wrote the drafts of the manuscript.
- Part of chapter 5 of this thesis is published as [4]. I co-designed the study and co-wrote the drafts of the manuscript.
- Part of chapter 5 of this thesis is published as [5]. I co-designed the study and co-wrote the drafts of the manuscript.
- Part of chapter 5 of this thesis is published as [6]. I co-designed the study and co-wrote the drafts of the manuscript.
- Part of chapter 5 of this thesis is published as [7]. I co-designed the study and co-wrote the drafts of the manuscript.

- Chapter 6 of this thesis is published as [8]. I co-designed the main techniques and co-wrote the drafts of the manuscript.
- Chapter 7 of this thesis is published as [9]. I co-designed the main techniques and co-wrote the drafts of the manuscript.
- Chapter 9 of this thesis is published as [10]. I designed the study, analyzed the data, and wrote the drafts of the manuscript.
- Part of Chapter 10 of this thesis is published as [11]. I designed the study, analyzed the data, and wrote the drafts of the manuscript.
- Chapter 12 of this thesis is published as [12]. I designed the study, analyzed the data, and wrote the drafts of the manuscript.
- Part of chapter 13 of this thesis is published as [13]. I designed the study and wrote the drafts of the manuscript.
- Part of chapter 13 of this thesis is published as [14]. I co-designed the study and co-wrote the drafts of the manuscript.

In addition to the statements above, in cases where I am not the corresponding author of a published item, permission to include the published material has been granted by the corresponding author.

Signature:



Amani Abusafia
School of Computer Science
Faculty of Engineering
The University of Sydney
13th January 2024

As a supervisor for the candidature upon which this thesis is based, I can confirm that the authorship attribution statements above are correct.



Athman Bouguettaya
School of Computer Science
Faculty of Engineering
The University of Sydney
13th January 2024

*In Memory of My Deeply Missed Grandparents
Dedicated to My Wonderful Parents, Amna and Mohammed,
and to Palestine, in the hope of its lasting freedom and peace.*

Acknowledgements

My PhD journey has been a transformative experience. It has been a period of profound self-discovery, growth, and affirmation of my passion for research and academia. This thesis not only stands as a monument to my personal efforts but also pays tribute to the unwavering support and faith bestowed upon me by others. It is a testament to the countless individuals who have showered me with encouragement and unwavering belief throughout my academic journey.

I want to express my utmost gratitude to the Almighty Allah. I firmly believe that his guidance, mercy, and blessings have been the foundation of my journey in this life.

I want to thank my supervisor, Professor Athman Bouguettaya, for his continuous support during my PhD. His contributions to developing my critical thinking, writing, and presentation skills have been outstanding. He taught me to be a persevering, dedicated, and meticulous researcher throughout the years.

I am beyond grateful to Dr. Abdallah Lakhdari, whose exceptional supervision made this research possible. His guidance, empathy, and assistance have been invaluable and empowered me to be a better researcher and academic. I want to acknowledge and express my gratitude to all the members of the Sensors, Clouds, and Services Lab for their help and constant feedback during my PhD journey.

Special thanks to Professor Zaher Al Aghbari and Professor Ibrahim Kamel, who introduced me to the world of research and served as my master's research advisor. I am incredibly grateful to every teacher who taught me during any stage of my academic life. They have been my continuous source of inspiration, guidance, and support. Special thanks to my mentor, Sindi Mabaso-Koyana, whose wisdom and advice will always remain with me. To my students at the University of Sharjah, who encouraged me to pursue my PhD, and to my students at the University of Sydney, whose hard work and enthusiasm have always been inspiring. To all the incredible women who

I met through Zonta, who are committed to making a difference in this world. I also thank Janelle Cruickshank for her editorial assistance in proofreading my thesis.

I am grateful to my friends who supported me during my failure and celebrated my success. To Sarah, Reem, Tasneem, Salam, Ashwaq, Arwa, Hend, Manar, Nafisa, Maytha, Ayesha, and many others, thank you for your love, support, and belief in me, sometimes even more than I believed in myself. Thank you for the intellectual conversations that inspired and helped me through tough times. To Saja, her mother Fadwa, and the rest of their family; to Doaa and Moawea, thank you for being my family in a country where I knew nobody. I am truly indebted to you for your unlimited help and support.

I also want to express my boundless and heartfelt gratitude to my parents, Amna and Mohammed. To my mother, who instilled in me the love of knowledge, and to my father, who supported me in whatever I wanted to do, regardless of societal expectations. My parents have laid the foundation on which I now stand. They have always showered me with unconditional love and care and have consistently striven to push me towards doing my best. Their support is indeed beyond repayment. I am also thankful to my beloved sister Suha, my dear brothers Ahmed, Mohammed, and Omar, and my sisters-in-law, Hiba and Isra, for their love, care, continuous encouragement, and prayers. To my niece Haya and my nephews, Abdullah, Omar, Asem, and Fares, your love is what makes this life enjoyable. And finally, to my entire family in Palestine, I live to make you proud, and I hope one day we meet to celebrate this achievement under the banner of freedom and peace.

While the acknowledgment section cannot include all the individuals who have supported my work, I want to extend my profound appreciation to everyone who has played a role in this research. While your contributions were indispensable, any errors that may be present in this thesis are solely my responsibility.

List of Publications

- (1) **Amani Abusafia**, Athman Bouguettaya, and Sajib Mistry, “Incentive-Based Selection and Composition of IoT Energy Services.”, In *IEEE International Conference on Services Computing (SCC)*, 2020. (Core 2020 Ranking: A)
- (2) **Amani Abusafia**, and Athman Bouguettaya, “Reliability Model for Incentive-Driven IoT Energy Services.”, In *EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking, and Services (EAI Mobi-Quitous)*, 2020. (Core 2020 Ranking: A)
- (3) Abdallah Lakhdari, **Amani Abusafia**, and Athman Bouguettaya, “Crowd-sharing Wireless Energy Services.”, In *IEEE International Conference on Collaboration and Internet Computing (CIC)*, 2020. Vision paper.
- (4) Zishan Qin, Dipankar Chaki, Abdallah Lakhdari, **Amani Abusafia**, and Athman Bouguettaya, “Occupancy Estimation from Thermal Images.”, In *International Conference on Service-Oriented Computing (ICSOC)*, 2021. (CORE A)
- (5) Jessica Yao, **Amani Abusafia**, Abdallah Lakhdari, and Athman Bouguettaya, “Wireless IoT Energy Sharing Platform.”, In *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, 2022. Demo paper. (CORE A*)
- (6) **Amani Abusafia**, Athman Bouguettaya, and Abdallah Lakhdari, “Quality of Experience Optimization in IoT Energy Services.”, In *IEEE International*

Conference on Web Services (ICWS), 2022. (CORE A)

- (7) Balsam Alkouz, **Amani Abusafia**, Abdallah Lakhdari, and Athman Bouguettaya, “In-Flight Energy-Driven Composition of Drone Swarm Services.”, In *IEEE Transactions on Services Computing (TSC)*, 2022. (Impact Factor: 11.019)
- (8) Ruochun Wang, **Amani Abusafia**, Abdallah Lakhdari, and Athman Bouguettaya, “The nudging effect on tracking activity.”, In *ACM international joint conference on pervasive and ubiquitous computing (UbiComp)*, 2022. Poster paper. (CORE A*)
- (9) Riccardo Menon, Abdallah Lakhdari, **Amani Abusafia**, Qijun He, and Athman Bouguettaya, “DIY-IPS: Towards an Off-the-Shelf Accurate Indoor Positioning System.”, In *ACM International Conference on Mobile Computing And Networking (MobiCom)*, 2022. Demo paper. (CORE A*)
- (10) Yisheng Alison Zheng, **Amani Abusafia**, Abdallah Lakhdari, Shing Tai Tony Lui, and Athman Bouguettaya, “IMAP: Individual huMAN mobility Patterns visualizing platform.”, In *ACM International Conference on Mobile Computing And Networking (MobiCom)*, 2022. Demo paper. (CORE A*)
- (11) Pengwei Yang, **Amani Abusafia**, Abdallah Lakhdari, and Athman Bouguettaya, “Towards peer-to-peer sharing of wireless energy services.”, In *International Conference on Service Oriented Computing (ICSOC)*, 2022. Demo paper. (CORE A)
- (12) **Amani Abusafia**, Abdallah Lakhdari, and Athman Bouguettaya, “Service-Based Wireless Energy Crowdsourcing.”, In *International Conference on Service Oriented Computing (ICSOC)*, 2022. Vision paper. (CORE A)

- (13) **Amani Abusafia**, Athman Bouguettaya, and Abdallah Lakhdari “Maximizing Consumer Satisfaction of IoT Energy Services.”, In *International Conference on Service Oriented Computing (ICSOC)*, 2022. (CORE A)

- (14) Pengwei Yang, **Amani Abusafia**, Abdallah Lakhdari, and Athman Bouguettaya, “Monitoring Efficiency of IoT Wireless Charging.”, In *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, 2023. Demo paper. (CORE A*)

- (15) Jiajie Li, **Amani Abusafia**, Abdallah Lakhdari, and Athman Bouguettaya, “Activity-based Profiling for Energy Harvesting Estimation.”, In *ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, 2023. Poster paper. (CORE A*)

- (16) Pengwei Yang, **Amani Abusafia**, Abdallah Lakhdari, and Athman Bouguettaya, “Energy Loss Prediction in IoT Energy Services.”, In *IEEE International Conference on Web Services (ICWS)*, 2023. (CORE A)

- (17) Yisheng Zheng, Abdallah Lakhdari, **Amani Abusafia**, Shing Tai Tony Lui, and Athman Bouguettaya, “CrowdWeb: A Visualization Tool for Mobility Patterns in Smart Cities.”, In *IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2023. Demo paper. (CORE A)

- (18) **Amani Abusafia**, Athman Bouguettaya, Abdallah Lakhdari, and Sami Yangui, “Context-Aware Trustworthy IoT Energy Services Provisioning.”, In *International Conference on Service Oriented Computing (ICSOC)*, 2023. (CORE A)

- (19) **Amani Abusafia**, Abdallah Lakhdari, and Athman Bouguettaya, “Flow-Based Energy Services Composition.”, In *IEEE Transactions on Services Computing (TSC)*, 2023, (Impact Factor: 11.019)

Abstract

Energy Service refers to the wireless delivery of energy among nearby IoT devices. Energy services may be harvested from natural resources such as kinetic activity or body heat. *Crowdsourcing* is an efficient way to leverage energy services to create a self-sustained environment. Moreover, crowdsourcing energy offers *convenient* and *ubiquitous* power access for IoT users. A key aspect to unlocking the full potential of the energy service ecosystem is to design an *end-to-end* Service Oriented Architecture (SOA). We identify three key components of the SOA: energy *provider*, energy *consumer*, and *super-provider*. In this architecture, providers advertise services, consumers submit requests, and the super-provider (i.e., microcell's owner) manages the exchange of energy services between providers and consumers. Existing research addresses challenges related to the consumer's perspective. This thesis addresses challenges from the provider's and super-provider's perspectives. In particular, we investigate composing energy requests to overcome providers' resistance. This thesis also applies energy services in drone swarm delivery, aiming to improve delivery efficiency by minimizing the number of stops and recharging time duration. As for the super-provider's perspective, this thesis focuses on two aspects: (1) efficient energy services provisioning and (2) Quality of Experience (QoE) based energy service provisioning. Therefore, this thesis examines utilizing partial services and a recommendation-based approach to efficiently balance energy distribution in a smart city. Moreover, a QoE-aware framework is presented to leverage the provisioning of energy services to ensure the highest consumer satisfaction. The framework addresses challenges related to ensuring service availability, such as providers' provisioning preferences and trust. In summary, this thesis makes several contributions toward the adoption and optimization of energy services in IoT environments.

Contents

Statement of Originality	ii
Authorship Attribution Statement	iii
Acknowledgements	vi
List of Publications	viii
Abstract	xii
Contents	xiii
List of Figures	xxii
List of Tables	xxvii
Chapter 1 Introduction	1
1.1 Motivation.....	5
1.2 Research Gaps and Key Challenges	7
1.3 Research Objectives	10
1.4 Thesis Contributions.....	11
1.4.1 Incentive-based Service Composition Framework	11
1.4.2 Reliability Model for Incentive-driven IoT Energy Services	12
1.4.3 Energy Services Demonstration.....	12
1.4.4 In-flight Energy Services in Drone Swarm Delivery.....	13
1.4.5 Flow-based Energy Service Composition	13
1.4.6 Recommendation-based Crowdsourcing of Energy Services	14
1.4.7 QoE-aware Energy Service Composition	14
1.4.8 Preference-aware Energy Service Composition.....	15
1.4.9 Constraint-aware Energy Service Composition	15

1.4.10	Context-aware Energy Trustworthy Service Composition	15
1.5	Thesis Outline	15
Chapter 2	Literature Review	19
2.1	Crowdsourced IoT Services	19
2.1.1	Incentives in crowdsourcing	19
2.1.2	Reliability assessment in IoT services	20
2.1.3	Trust assessment in IoT services	21
2.2	Wireless Energy Sharing	22
2.2.1	Technology and applications	22
2.2.2	Energy Sharing Techniques	22
2.3	Energy Services Selection and Composition	23
2.3.1	Services Selection and Composition	24
2.3.2	Energy Services	25
2.4	Quality of Experience	26
2.5	Conclusion	27
Chapter 3	Incentive-based Selection and Composition of IoT Energy Services	28
3.1	Introduction	28
3.2	Motivating Scenario	28
3.3	System Model and Definition	30
3.4	Incentive Model	33
3.5	Composition Framework for Energy Service Requests	37
3.5.1	Selection of Energy Service Requests	38
3.5.2	Incentive-based Composition of Energy Service Requests	39
3.6	Experimental Results and Discussions	40
3.6.1	Evaluation of the Composition Framework	40
3.6.2	Evaluation of the Incentive Model	43
3.7	Conclusion	46
Chapter 4	Reliability-based Energy Service Composition	47

4.1	Introduction	47
4.2	Motivating Scenario	48
4.3	System Model and Definitions	50
4.4	Reliability Model	53
4.5	Composition Framework for Energy Requests	59
4.5.1	Selection of Energy Service Requests	60
4.5.2	Reliability-based Composition of Energy Requests	62
4.5.3	Adaptive Reliability-based Composition of Energy Requests	62
4.6	Experiments and Discussions	63
4.6.1	Dataset Description	64
4.6.2	Evaluation of the Composition Framework	65
4.7	Conclusion	69
Chapter 5	Energy Service Demonstration	71
5.1	Peer-to-peer Sharing of Wireless Energy Services	72
5.1.1	Wireless Energy Services Demonstration	72
5.1.1.1	System Overview	73
5.1.1.2	Demo Scenario	74
5.1.1.3	Demo Setup	74
5.1.1.4	Preliminary Experiments	75
5.1.2	Wireless Energy Estimation Prediction	76
5.1.2.1	Motivating Scenario	77
5.1.2.2	Wireless Energy Estimation Framework	79
5.1.2.3	Evaluation	82
5.2	Activity-based Profiling for Energy Harvesting Estimation	83
5.2.1	Activity-based Profiling Framework	85
5.2.1.1	Evaluation Metrics	87
5.2.2	The Nudging Effect on Tracking Activity	88
5.2.2.1	Study Method	88
5.2.2.2	Results	89
5.3	Conclusion	91

Chapter 6	Energy Services in Drone Swarm Delivery	92
6.1	Introduction	92
6.2	Motivating Scenario	98
6.3	Related Work	100
6.3.1	Swarm-based Drone Services	101
6.4	System Architecture and Composition Model	104
6.4.1	Problem Formulation	107
6.5	Swarm-based Drone Services Composition Framework	108
6.5.1	Swarm-based Drone Service Pre-composition	108
6.5.1.1	Formation Selection	108
6.5.1.2	Support Drones Selection	109
6.5.1.3	Support Drones Positioning	111
6.5.2	Swarm-based Drone Services Composition with In-flight Energy Sharing	113
6.5.2.1	In-flight Energy Sharing Composition	116
	Priority-Based (PB) Energy Sharing	117
	Fairness-Based (FB) Energy Sharing	119
6.5.2.2	Formation Re-ordering	119
6.6	Experiments and Results	121
6.6.1	Dataset and Experiments Setup	121
6.6.2	Effectiveness	123
6.6.3	Runtime Efficiency	127
6.7	Conclusion	127
Chapter 7	Flow-based Energy Service Composition	129
7.1	Introduction	129
7.2	Motivating Scenario	131
7.3	System Model and Problem Formulation	134
7.3.1	Energy Service Model	134
7.3.2	Energy Request Model	135
7.3.3	Energy Utilization and Problem Formulation	135

7.4	Wireless Energy Sharing Framework	137
7.4.1	Phase 1: Proxy Management	138
7.4.2	Phase 2: IoT Coordinator Management	139
7.4.2.1	Bipartite Graph Generation	139
7.4.2.2	Network Flow Generation	141
7.4.2.3	Maximum Flow Matching Composition	142
7.4.2.4	Energy Services Composition Assessment	144
	Energy Utilization	145
	Fulfillment Rate	145
	Consumers and Providers' Satisfaction	146
	Global Consumers and Providers' Satisfaction	147
7.4.2.5	Partial Bipartite Graph Generation	148
7.4.2.6	Partial Maximum Flow Matching Composition	150
7.5	Experiments Results	150
7.5.1	Dataset Description	151
7.5.2	Evaluation of the Composition Framework	151
7.5.2.1	Energy Utilization Evaluation	152
7.5.2.2	Fulfillment Rate Evaluation	154
7.5.2.3	Consumers' Satisfaction Evaluation	156
7.5.2.4	Providers' Satisfaction Evaluation	158
7.5.2.5	Computation Efficiency Evaluation	161
7.6	Conclusion	162
Chapter 8	Recommendation-based Crowdsourcing of Energy Services	163
8.1	Introduction	163
8.2	Motivating Scenario	164
8.3	Preliminaries	166
8.3.1	Problem Definition	169
8.4	Recommendation-based Energy Services Framework	170
8.4.1	Mobility Patterns Extraction	171
8.4.2	Mobile Energy Services Modelling	171

8.4.3	Heuristic-Based Recommendation	171
8.4.3.1	Complexity Analysis.....	173
8.4.4	Recommendation Assessment	173
8.5	Evaluation.....	175
8.5.1	Dataset Description	175
8.5.2	Evaluation of the Recommendation Framework	177
8.5.2.1	Effectiveness Evaluation	177
8.5.2.2	Computation Efficiency Evaluation.....	181
8.6	Conclusion	181
Chapter 9	QoE-based Energy Services Composition	183
9.1	Introduction	183
9.1.1	Motivating Scenario	185
9.2	Preliminaries	188
9.2.1	Problem Definition.....	189
9.3	Quality of Experience Model.....	190
9.4	Quality of Experience Framework	192
9.4.1	Microcell Energy Demand Advertisement	192
9.4.2	Energy Services Composition	193
9.4.2.1	Partial-based Energy Services Composition	193
9.4.2.2	Demand-based Energy Services Composition.....	194
9.4.3	Assessing Quality of Experience	196
9.5	Evaluation.....	196
9.5.1	Dataset Description	197
9.5.2	Evaluation of the Composition Framework.....	198
9.5.2.1	Quality of Experience Evaluation	198
9.5.2.2	Threshold Impact Evaluation	201
9.5.2.3	Computation Efficiency Evaluation.....	202
9.6	Conclusion	202
Chapter 10	Preference-aware Energy Service Composition	204

10.1	Introduction	204
10.1.1	Motivating Scenario	205
10.2	System Model	208
10.2.1	Preliminaries	208
10.2.2	Providers' Flexible Temporal Provisioning Preferences	209
10.2.3	Super-Provider Preferences	210
10.2.4	Problem Definition	212
10.3	Quality of Experience Framework	214
10.3.1	Providers' Temporal Preferences Ranking	214
10.3.1.1	A CP-Net-based preferences ranking	215
10.3.1.2	Time slots ranking	216
10.3.2	QoE-Aware Energy Services Composition	217
10.3.2.1	Importance-based Energy Services Composition	217
10.3.2.2	Heuristic-based Energy Services Composition	218
10.4	Experiments Results	220
10.4.1	Dataset Description	221
10.4.2	Evaluation of the Composition Framework	222
10.4.2.1	Quality of Experience Evaluation	222
10.4.3	Cost Evaluation	225
10.4.3.1	Energy Utilization Evaluation	226
10.4.3.2	Computation Efficiency Evaluation	228
10.5	Conclusion	228
Chapter 11	Constraints-aware Energy Service Composition	230
11.1	Introduction	230
11.1.1	Motivating Scenario	232
11.2	System Model	234
11.2.1	Energy Service and Request Models	235
11.2.2	Super-Provider Preferences	236
11.2.3	Quality of Experience (QoE) Model:	238
11.2.4	Problem Definition	239

11.3	Energy Service Composition Framework	241
11.3.1	Energy Services Expression of Interest (EoI) Generation	242
11.3.2	Energy Services Composition	245
11.3.2.1	MILP Formalization of Energy Service Composition	246
	Decision variables	246
	Objective function and constraints	246
11.3.3	Energy Services Provisioning Composition	248
11.4	Experiments Results	250
11.4.1	Dataset Description	250
11.4.2	Evaluation of the Composition Framework	252
11.5	Conclusion	255
Chapter 12	Context-aware Trustworthy Energy Service Composition	256
12.1	Introduction	256
12.2	Motivating Scenario	257
12.3	System Model	259
12.3.1	Energy Service Model	260
12.3.2	Super-Provider Preferences	261
12.3.3	Problem Definition	262
12.4	Trust Assessment Model	263
	Providers' Trustworthiness	266
12.5	Energy Service Composition Framework	266
12.5.1	Context-Aware Trust Assessment	267
12.5.1.1	Context Model	267
12.5.1.2	History Filtering:	271
12.5.1.3	Trust Assessment:	271
12.5.2	Trust-based Energy Services Composition	271
12.5.3	Assessing Quality of Experience and Cost	272
12.6	Evaluation	272
12.6.1	Dataset Description	272
12.6.2	Evaluation of the Composition Framework	273

12.6.2.1	Quality of Experience Evaluation	274
12.6.2.2	Computation Efficiency Evaluation.....	276
	Lessons Learned:	276
12.7	Conclusion	277
Chapter 13	Conclusion and Future Work	278
13.1	Open Challenges and Future Directions	280
13.1.1	Enabling Technologies:.....	281
13.1.1.1	Energy Loss	281
13.1.1.2	Miniature Delivered Energy	281
13.1.1.3	Limited Power Transfer Range.....	281
13.1.2	Human-in-the-Loop	282
13.1.2.1	Provisioning Resistance	282
13.1.2.2	Modeling Human Mobility	283
13.1.2.3	Modeling Usage Patterns.....	283
13.1.2.4	Determining Services Availability.....	284
13.1.3	Deployment of Energy Services Ecosystem.....	284
13.1.3.1	Reliability	284
13.1.3.2	Security	285
13.1.3.3	Privacy	285
	Bibliography	286

List of Figures

1.1	IoT energy services environment	3
1.2	Energy-service system oriented architecture	4
1.3	Usage-time using wire vs. wireless charging technologies	6
1.4	Thesis roadmap	12
3.1	Wireless energy sharing motivating scenario	29
3.2	Example of energy requests and energy service	32
3.3	The charging time measured based on the capacity of lithium-ion batteries [93]	33
3.4	Users' temporal preferences to share energy	35
3.5	Average importance of incentive attributes based on user preferences for energy sharing	36
3.6	The composition framework for energy service requests	38
3.7	Average total reward for IB, BF, and FCFS approaches.	41
3.8	Average of providers' remaining energy for IB, BF, and FCFS approaches	42
3.9	Participation rate of users in sharing energy with and without a reward	44
3.10	Participation rate of users in sharing energy with and without a reward	44
3.11	Participation rate of users in sharing energy with and without a reward	45
4.1	Reliability-based wireless energy sharing system	48
4.2	Example of energy requests with reliability	49
4.3	Example of voluntary disruption	54
4.4	Example of involuntary disruption	55
4.5	Example of random behavior	57
4.6	The composition framework for ER services	60
4.7	The average of the reliability score	66
4.8	The average of total rewards	67

4.9	The average of providers' remaining energy	68
4.10	The average execution time	69
5.1	Wireless energy charging scenario	73
5.2	Used charging technologies	74
5.3	Comparison among different charging technologies	75
5.4	Comparison of wireless charging with different battery levels for consumer	76
5.5	The energy services environment	77
5.6	Example of the motivating scenario	78
5.7	Wireless energy estimation framework	79
5.8	Hardware experiments setup for data collection	80
5.9	Training datasets visualization	81
5.10	Box-plot of Root Mean Square Error (RMSE)	83
5.11	Energy Harvesting Estimation System	84
5.12	Activity-Based Profiling System Architecture	85
5.13	Active rates of both groups each month	90
5.14	Average active rates difference between experiment and control groups each month	90
5.15	Active rates distribution in each month for (A) Experiment group (B) Control group	90
6.1	Swarm-based delivery services with in-flight energy-sharing	99
6.2	Effect of in-flight recharging on swarm-based drone services composition	99
6.3	Swarm-based drone services system architecture	103
6.4	Swarm-based drone service composition framework	108
6.5	Different swarm formations A: Column, B: Front, C: Echelon, D: Vee, E: Diamond	109
6.6	Support drones positioning	112
6.7	Drones' energy-sharing requests and service	117
6.8	Number of successful requests using in-flight energy-sharing	124
6.9	Delivery Times of Various Energy Sharing Methods	125
6.10	Delivery times under different positioning settings	125

6.11	Nodes time of various energy-sharing methods	127
6.12	Execution time of various energy-sharing methods	128
7.1	Crowdsourcing IoT energy services scenario	130
7.2	Crowdsourcing wireless energy example: (A) Phase 1: The proxy aggregates its wearables' spare energy as an energy service (B) Phase 2: The IoT coordinator composes services to requests.	132
7.3	Wireless energy services framework	138
7.4	Bipartite graph generation example	140
7.5	Maximum flow matching composition	144
7.6	Example of a bipartite graph and partial bipartite graph	148
7.7	Example of network flow and partial network flow	149
7.8	The average of the energy utilization EU vs. number of requests vs. (A) number of services (B)	153
7.9	Distributions of the energy utilization EU score vs. number of requests	154
7.10	Distributions of the energy utilization EU score vs. number of services	154
7.11	The average of the fulfillment rate FR vs. number of (A) requests and (B) services	155
7.12	The mean of consumer satisfaction CS vs. number of (A) requests and (B) services	156
7.13	The standard deviation score of consumer satisfaction CS vs. number of (A) requests and (B) services	157
7.14	The entropy score of consumer satisfaction CS vs. number of (A) requests and (B) services	157
7.15	The mean of providers' satisfaction PS vs. number of (A) requests and (B) services	159
7.16	The standard deviation score of providers' satisfaction PS vs. number of (A) requests and (B) services	160
7.17	The entropy score of providers' satisfaction PS vs. number of (A) requests and (B) services	161
7.18	The average execution time vs. number of (A) requests and (B) services	161

8.1	Crowdsourcing IoT energy services in the microcells of a Smart city with different energy demands	165
8.2	Recommendation-based energy services framework	170
8.3	Wireless Energy Dataset Collection Experiment Using the App Developed by [5]	177
8.4	Average FR of different number of microcells	178
8.5	Average FR of different number of providers	178
8.6	Average C of different number of microcells	179
8.7	Average C of different number of providers	179
8.8	EU of different number of microcells	180
8.9	EU of different number of providers	180
8.10	Average EU when changing the responsive rate	181
8.11	Computation Time	181
9.1	IoT energy services environment	185
9.2	(A) Microcell energy demand and providers services (B) Greedy energy provisioning approach	186
9.3	Quality of experience driven service composition framework	192
9.4	The average of satisfaction ratio	199
9.5	The average of energy utilization	200
9.6	The average of fulfillment ratio	200
9.7	The average of quality of experience	201
9.8	The average of $\{QoE, \mathcal{SR}, \mathcal{FR}\}$ using PB composition	201
9.9	The average of QoE using PB composition with various thresholds in a microcell	202
9.10	The average execution time of all compositions	202
10.1	IoT energy services environment	205
10.2	(A) Microcell energy demand and providers temporal preferences (B) Energy provisioning approaches	206
10.3	a Provider's temporal preferences	210
10.4	Quality of experience aware service composition Framework	214

10.5	Provider's CP-Net-based temporal preferences ranking	216
10.6	The average of QoE vs. number of requests	223
10.7	The average of QoE vs. number of services	224
10.8	The average of QoE with different providers' temporal preferences	225
10.9	The average of cost vs. number of requests	225
10.10	The average of energy utilization vs. number of requests	227
10.11	The average of the energy utilization EU vs. number of services	227
10.12	The average execution time vs. number of requests	228
11.1	IoT energy services environment	232
11.2	Example of the motivating scenario	233
11.3	Quality of experience aware service composition framework	241
11.4	Provider's PCP-Net-based provisioning preferences	244
11.5	The average of QoE vs. number of requests	252
11.6	The average of cost vs. number of requests	253
11.7	The average of energy utilization vs. number of requests	254
12.1	IoT energy services environment	258
12.2	Example of how different trust assessments yield different trust scores	259
12.3	Trust-based energy service composition framework	267
12.4	Context-aware trust assessment model	270
12.5	The average of QoE using different history	274
12.6	The average of QoE using different trust assessments	275
12.7	The average of QoE in trustworthy environment	275
12.8	The average of QoE in neutral environment	276
12.9	The average of QoE in untrustworthy environment	276
12.10	The average of cost	276
12.11	The average of computation cost	276

List of Tables

3.1 Experiments variables	41
4.1 Experiments variables	65
5.1 Evaluation Metrics of clustering algorithms	87
6.1 Summary of notations	106
6.2 Redundancy values for different failure probabilities	112
6.3 Experimental Variables	122
7.1 Parameters of the experiments' settings	152
7.2 Statistics of the experiments' environment	152
8.1 Experiments Variables	177
9.1 Experiments variables	199
10. Parameters of the experiments' settings	221
10. Statistics of the experiments' environment	222
11. Statistics of the experiment's environment	251
12. Experiments Variables	274

CHAPTER 1

Introduction

Internet of Things (IoT) is a paradigm that enables everyday objects (i.e., *things*) to connect to the internet and exchange data [15][16]. IoT devices usually have augmented capabilities, including sensing, networking, and processing [15]. The number of connected IoT devices is expected to reach 125 billion in 2030 [17]. This potential pervasiveness of IoT provides opportunities to *abstract* their capabilities using the *service paradigm* as *IoT services* [18][19]. IoT services are defined by their functional and non-functional attributes. The functional attributes define the purpose of the service, such as sharing internet access using WiFi. The non-functional attributes are the properties that assess the Quality of Service (QoS), e.g., signal strength, reliability, etc. For example, an IoT device owner may offer their WiFi as a hotspot (i.e., service provider) to other nearby IoT devices (i.e., service consumers). A multitude of novel IoT services may be used to enable intelligent systems in several domains, including smart cities, smart homes, and healthcare [19][18]. Examples of IoT services are computing offloading [20], WiFi hotspots [21], environmental sensing [22], and energy services [23]. IoT services may be *exchanged* between IoT devices as *crowdsourced* IoT services. For example, an IoT device may be charged by crowdsourcing wireless energy services from nearby IoT devices [24]. These crowdsourced IoT services present a convenient, cost-effective, and sometimes the only possible solution for resource-constrained devices [24]. Our focus is on the use of energy services.

Energy service, also known as *Energy-as-a-Service (EaaS)*, refers to the *wireless power transfer* among nearby IoT devices [25][23]. Energy services is IoT service with energy delivered from an energy provider (e.g., a smart shoe) to an energy consumer

(e.g., a smartphone). We consider a particular set of IoT devices named *wearables*. Wearables refer to anything worn or hand-held like smart shirts, smartwatches, and smartphones [26]. Wearables may harvest energy from natural resources such as kinetic activity, solar power, or body heat [27][28][29]. For instance, a smart shoe using a PowerWalk kinetic energy harvester may produce 10-12 watts of on-the-move power¹. In this respect, wearing a PowerWalk harvester may generate energy to charge up to four smartphones from an hour's walk at a comfortable speed. Crowdsourcing energy services offer a *green* solution as it reduces carbon footprint by utilizing spare and renewable energy [14]. Additionally, energy services present a *convenient* and *ubiquitous* power access for IoT users [30][31][32]. Indeed, devices do not need to be tethered to a power point or use power banks. In addition, crowdsourcing energy offers an *ubiquitous* power access for IoT users as they may be charged anytime and anywhere, even while moving [23][32]. Moreover, there are multiple domains where energy services are necessary due to the lack of power outlets. These domains include disaster control, entertainment, and crisis response [14].

Energy services may be deployed through the newly developed "Over-the-Air" wireless charging technologies [23][33]. Several companies are developing charging technologies that enable IoT devices to charge wirelessly over a distance, including, WiTricity², Xiaomi³, Energous⁴, Powercastco⁵, and, Cota⁶. For example, WiTricity started based on the work of [34], where they succeeded in transferring 60 W of power wirelessly to power a light bulb. Additionally, Energous developed a device that can charge up to 3 watts of power within a 5-meter distance. Another example is Xiaomi's Mi Air charger which transmits energy wirelessly to nearby IoT devices. Although current technology may not provide efficient energy delivery [32], technological advances are expected to enable devices to exchange larger amounts of energy [35]. Recent studies have suggested using a central source to charge IoT devices wirelessly [36]. However,

¹bionic-power.com

²witricity.com

³mi.com

⁴energous.com

⁵powercastco.com

⁶ossia.com

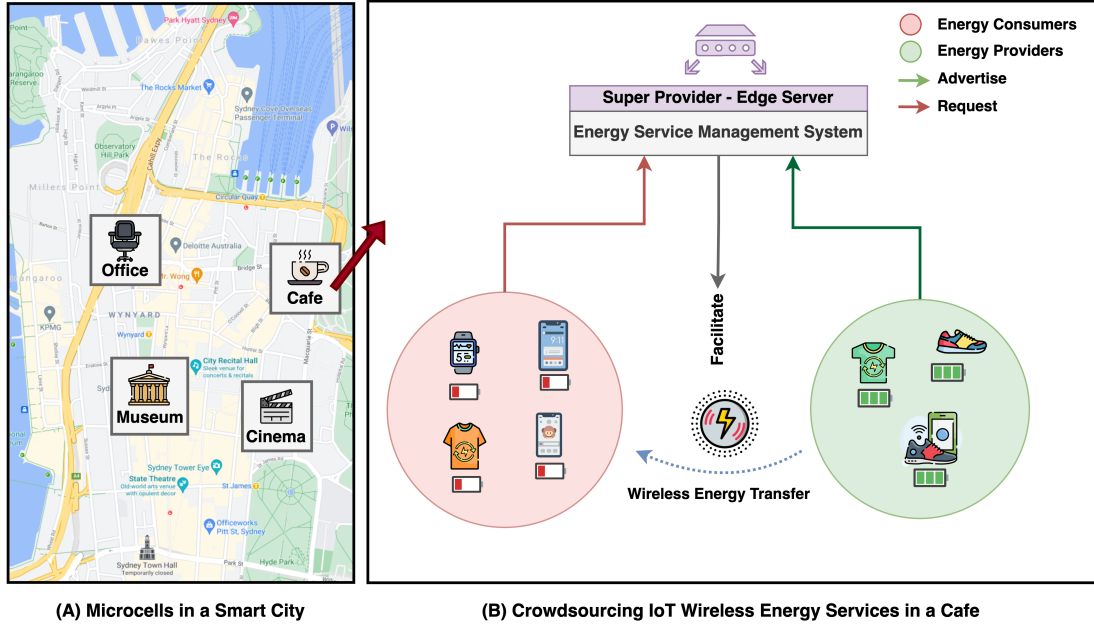


FIGURE 1.1. IoT energy services environment

a central source usually requires a high-frequency magnetic field to transfer energy over distance [37]. Studies have shown that a strong magnetic field may harm human health [38]. Therefore, crowdsourcing energy services using multiple weaker energy-emitting sources reduces health risks by sharing low-frequency magnetic fields [36].

The proposed crowdsourced IoT energy environment consists of IoT users congregating and moving across *microcells* [23][39]. A microcell is any confined area in a smart city where people may gather (e.g., coffee shops, restaurants, movie theaters, libraries) (see Fig. 1.1(A)). IoT users are assumed to act as energy providers or consumers (see Fig. 1.1(B)). A key aspect to unlocking the full potential of the energy service ecosystem is to design an *end-to-end* Service Oriented Architecture (SOA). We identify three key components of the SOA: energy service *provider*, energy service *consumer*, and *super-provider* (see Fig. 1.2). In this architecture, providers advertise services, consumers submit requests, and the super-provider (i.e., the microcell's owner and IoT coordinator) manages the exchange of energy services between providers and consumers. The energy crowdsourcing environment's deployment relies on

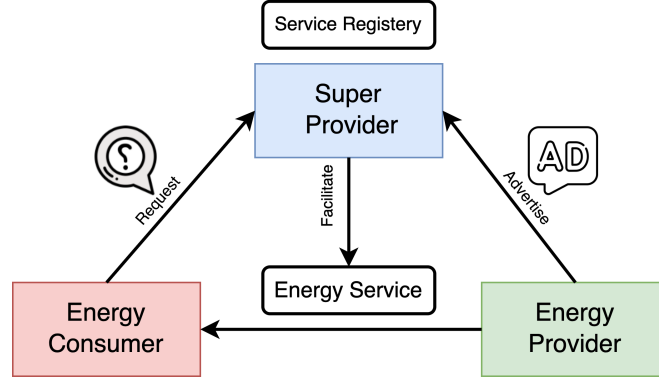


FIGURE 1.2. Energy-service system oriented architecture

the energy providers' willingness to participate [14]. We assume that the IoT coordinator (i.e., super-provider) offers effective incentives in the form of credits to encourage providers' participation. The credits may be used to receive energy when the providers act as consumers in the future [24]. The IoT coordinator is assumed to be deployed one hop away from the energy providers and consumers (e.g., a router at the edge) to minimize the communication overhead and latency while advertising energy services and requests. Additionally, the participation of IoT users in the crowdsourced energy ecosystem relies on the security and trust of the deployed ecosystem. Novel security modules and new privacy-preserving trust models have been developed for crowdsourced IoT environments [40]. These aspects are outside the scope of this work. This thesis focuses on crowdsourcing energy services.

The main objective of this thesis is to establish an end-to-end energy services ecosystem. Establishing the ecosystem requires addressing challenges associated with each actor in the energy service oriented architecture (see Fig. 1.2). To the best of our knowledge, existing research mainly focused on the composition of energy services from a consumer's perspective [14]. In contrast, this thesis addresses challenges from both the provider's and super-provider's perspectives. In particular, we investigate composing energy requests for a provider in a crowdsourced IoT environment. In that respect, we address provider resistance and assess consumer reliability to compose energy requests. Additionally, we demonstrate several prototypes to prove the feasibility

of deploying energy services. Additionally, we consider the impact of energy request composition in a different IoT environment, specifically drone swarm delivery.

From the super-provider's perspective, we focus on two aspects: (1) efficient energy services provisioning, i.e., determining strategies to allocate energy services that maximize energy utilization, and (2) Quality of Experience (QoE) based energy service provisioning. In this regard, we propose using energy services as a tool to ensure consumers' quality of experience and address related challenges.

1.1 Motivation

Crowdsourcing energy services provides numerous benefits to both the environment and IoT users. Each of the benefits is discussed in detail below.

- **Enabling Sustainable IoT Ecosystem:** The proliferation of IoT devices leads to a significant increase in energy consumption [41][42]. IoT devices' global energy demand is predicted to reach 46 TWh by 2025 [42]. This tremendous energy demand accounts for 6% to 8% of the global carbon footprint generated by information technologies [43][44]. Crowdsourcing energy services reduces carbon footprint by reducing dependence on fossil fuels to charge IoT devices. In addition, crowdsourcing energy services relies on renewable or spare energy resources. Renewable energy may be harvested from natural resources such as body heat, solar energy, or kinetic movement [29].
- **Extending Battery Endurance:** IoT devices are constrained by the size of their batteries [45][46]. The small size of the battery results in a limited energy storage capacity. This limited capacity hinders the capabilities of IoT devices. Increasing the battery capacity for IoT devices faces several challenges, including safety, weight, cost, and recycling [47]. Other charging methods, such as carrying a power cord and searching for a power outlet, are

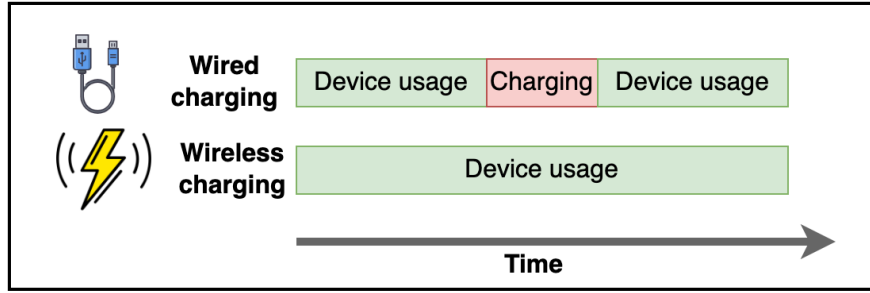


FIGURE 1.3. Usage-time using wire vs. wireless charging technologies

inconvenient for users. Hence, crowdsourcing energy services becomes an attractive solution to improve battery endurance [48].

- **Unlimited Usage Time:** Crowdsourcing energy services enables the recharging of IoT devices without interrupting their usage time. In contrast, wired charging may require IoT users to stop their usage while charging. Hence, wireless recharging of IoT devices may result in unlimited usage time, i.e., constantly using them to provide or access services (see Fig. 1.3). Thus, crowdsourcing energy services prolongs the IoT device's usage lifetime, especially when the external energy supply is unavailable.
- **Spatial Freedom:** Crowdsourcing energy services provides spatial independence for IoT users. As previously mentioned, IoT devices are frequently required to be tethered to power points or carry power banks to be charged. IoT devices' last obstacle to achieving their *spatial freedom* consists of charging wires and pads. The envisioned environment is expected to enable wireless power transfer delivery up to five meters [14].
- **Ubiquitous Power Access:** As aforementioned, energy services offer spatial freedom for IoT devices. Therefore, crowdsourcing energy services offers *ubiquitous* power access to charge IoT devices anytime and anywhere. The *ubiquity* of wireless charging facilitates access to energy for IoT users.
- **Flexible Contracts:** Unlike traditional services, crowdsourced energy services environments do not have lock-in contracts [25]. The flexible contracts enable

consumers and providers to request or offer full or partial services according to their preferences [25]. In addition, providers and consumers may extend their stay-time to offer or receive an energy service [49].

- **Convenience Charging:** The *ubiquity* of wireless charging offers a convenient alternative to charge IoT devices [32]. In addition, as previously discussed, the *flexibility* to participate in crowdsourcing energy services without any long-term contract offers convenience for IoT users. IoT users might offer or request energy services without any lock-in contract. They can also move around freely while providing or receiving energy [50]. This convenience ensures that the use of this ecosystem will expand widely.
- **Business Edge:** As mentioned, energy services offer a convenient solution to charge IoT devices. Thus, energy services may be used as a complementary service to provide customers with the best quality of experience when visiting a business [33]. For example, energy services, like WiFi, may charge customers' wearables in a cafe. Such a service will increase customers' satisfaction with a business. Customer satisfaction is the main factor in ensuring that customers keep coming back to businesses, i.e., maintaining or increasing foot traffic [51]. For example, a case study showed that "Sacred", a cafe in London, had a noticeable increase in foot traffic after installing wireless charging points⁷. The *foot traffic* in a business has a direct impact on its revenue [52].

1.2 Research Gaps and Key Challenges

This thesis aims to develop an end-to-end energy services ecosystem. To achieve that, challenges related to each actor in the energy service system-oriented architecture need to be addressed (see Fig. 1.2). To the best of our knowledge, existing research mainly focuses on the composition of energy services from a consumer's aspect [14]. This thesis addresses challenges from a provider's and super-provider's perspectives.

⁷air-charge.com

- **Provisioning resistances:** The limited availability of energy in the crowd-sourced IoT environment may lead providers to resist offering their spare energy [48]. A provider's resistance is influenced by their available energy and the size of the requested energy. As a result, a context-aware incentive model is needed to address the providers' resistance to offering their energy service.
- **IoT users' commitment:** Energy services offer flexibility as they do not require lock-in contracts [49]. For instance, providers may offer full or partial services according to their preferences [25]. However, this flexibility may impact the commitment of IoT users. In addition to the no-lock contract, there is uncertainty in consumers' and providers' commitment to sharing energy due to mobility and usage behavior [14]. For example, a provider may terminate the energy transfer by leaving the transfer range. They may also stop the delivery of energy due to excessive device usage. Such disruptions in services may impact the participation of IoT users in the ecosystem. In such cases, real-time service replacement is not always guaranteed [49]. Therefore, a reliability model and a *trust* management framework are necessary to evaluate the uncertainty in consumers' and providers' commitment.
- **Dynamic IoT environment:** IoT environments are highly dynamic, i.e., IoT devices move freely between microcells. In the case of energy sharing, movements in and out of microcells may result in several challenges, including:
 - Disrupting the energy transfer: Moving IoT devices within a microcell may result in disconnecting the wireless energy transfer between them. Disconnection in the delivery of the energy transfer may result in its failure [50]. The disruption also may reduce the amount of shared energy, impacting the consumer's experience or the provider's reward. In such cases, a re-composition is required to ensure that the provider gets an optimal reward and that consumers get their required energy. However, a re-composition may not be guaranteed because an optimal replacement depends on other service requests' availability.

- Imbalance between energy availability and demand: Energy service providers and consumers may have different spatio-temporal preferences. These preferences' incongruity may severely impact the energy-sharing process within a microcell [39]. For instance, an energy consumer might not find their requested energy in a particular location at a specific time.
- **Context-dependent IoT environment:** The energy services environment is highly context-sensitive. For instance, energy scarcity may result in providers offering less energy than advertised [49]. Therefore, using existing trust metrics will likely result in low trust scores for most providers. Therefore, a new context-aware trust assessment framework is needed to better utilize the scarce energy services [14].
- **Quality of experience:** In this thesis, we proposed using energy services as a tool to enhance consumers' quality of experience in a microcell. Using quality of experience as a metric to assess consumers' satisfaction imposes several challenges, including:
 - Modeling quality of experience: Modeling the quality of experience in the energy service environment presents challenges due to several factors. First, resources in a crowdsourced environment are typically limited, meaning they cannot always meet all consumers' requirements. Second, given the limited resources and the shareable nature of energy services, these services may often be provided partially. For instance, a single service could be divided into smaller parts and shared among multiple consumers. Therefore, the consumers' experience with these partial services may differ from their experience with complete services.
 - Modeling providers' provisioning preferences: Providers may be flexible when provisioning their service, i.e., they can come at multiple time slots. However, they may prefer certain times over others, e.g., a provider may prefer to come in the morning rather than the evening. Intuitively, assigning providers to their preferred time slots will necessitate fewer rewards and potentially increase the certainty of their commitment.

Therefore, accurately modeling the flexible preferences of providers is a challenge that remains to be addressed.

- Cost: Typically, the super-provider would use rewards to encourage providers' participation. However, the super-provider's budget may not accommodate all energy demands, requiring careful allocation strategies. Traditional resource allocation algorithms may not work effectively here due to the uncertainty in provider availability and the super-provider's context-sensitive perspective. For instance, a super-provider may prioritize smaller requests with a higher weight in the super-provider's revenue, such as allocating a larger budget for a time slot that includes high-profit consumers with lower energy demand.

1.3 Research Objectives

In light of the challenges mentioned above, this thesis aims to provide a service composition framework for IoT energy services by achieving the following goals:

- Develop energy service composition frameworks to address factors that impact providers' provisioning resistance:
 - Design a context-aware incentive model to determine effective rewards.
 - Design a reliability-based model to assess consumers' commitment to receive energy.
 - Develop a framework to maximize energy utilization in different IoT contexts to study the impact of the context on the approach.
- Develop energy service composition frameworks to ensure efficient energy services provisioning for multiple IoT consumers.
 - Design a framework to maximize energy utilization between multiple providers to multiple consumers within a microcell.
 - Design a framework to maximize energy utilization across microcells within a smart city.

- Design quality of experience model and QoE-based energy service composition approaches to ensure consumers' satisfaction with energy services.
 - Design a quality of experience model from an energy service sharing perspective.
 - Design a provisioning preference model to reflect the providers' spatio-temporal preferences accurately.

1.4 Thesis Contributions

We present the roadmap to the key contributions of this thesis in Fig. 1.4. Then, we briefly discuss each contribution. Contributions detailed in Subsections 1.4.1 through 1.4.4 are aligned with the first research objective. Similarly, the contributions in Subsections 1.4.5 through 1.4.6 correspond to the second research objective. Lastly, the contributions in Subsections 1.4.7 through 1.4.10 are associated with the third research objective.

1.4.1 Incentive-based Service Composition Framework

We propose an incentive model that considers the characteristics that affect the provider's willingness to participate. An incentive model relies on *intrinsic* or *extrinsic* rewards to motivate participation. Intrinsic reward is an internal satisfaction motivated by altruistic purposes. In contrast, extrinsic reward is motivated by materialistic compensations [53]. This thesis focuses on computing the *extrinsic* reward using an incentive model. *Our incentive model will not consider intrinsic rewards as they are self-motivated.* The incentive model will consider the *contexts* of the providers and consumers that may affect the provider's willingness, such as the amount of requested energy and the required time to complete the energy service. *We aim to compose the best set of energy requests using our incentive model.*

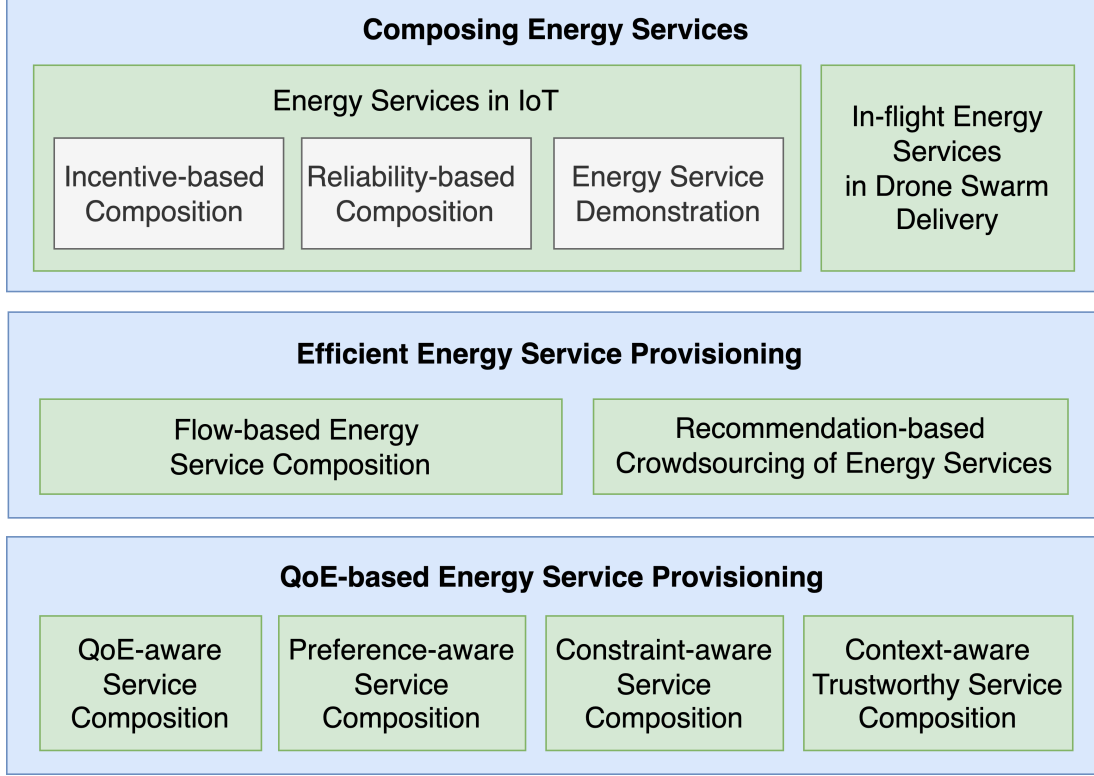


FIGURE 1.4. Thesis roadmap

1.4.2 Reliability Model for Incentive-driven IoT Energy Services

We propose a novel *reliability* model for composing energy service requests. The proposed model is based on consumers' behavior and history of energy requests. The reliability model ensures the maximum incentives to providers. Adaptive and priority scheduling compositions are proposed to compose the most reliable energy requests while maximizing providers' incentives.

1.4.3 Energy Services Demonstration

We developed several demonstrations and frameworks and conducted preliminary experiments to provide : (1) a proof of concept for deploying an energy services ecosystem for Internet of Things (IoT) devices, where users share spare or harvested energy; and (2) comprehensive real-world datasets to evaluate the effectiveness of the proposed models and composition frameworks. These demonstrations and experiments can be

grouped into two main components: a wireless energy transfer setup and an energy harvesting estimation model. The platform collects real energy transfer datasets and demonstrates the transfer of wireless energy between smartphones to share energy. The energy harvesting estimation model may be used to predict the amount of energy a provider may harvest. The predicted amount may be used in forecasting the provider service availability. To design an energy harvesting model, we designed an activity profiling framework to estimate harvested kinetic energy. The framework is the first step to building the energy harvesting model.

1.4.4 In-flight Energy Services in Drone Swarm Delivery

We explore the impact of composing energy services in a different IoT environment with a dynamic nature and varying goals, specifically, drone swarms. We propose a framework for swarm-based drone delivery services with in-flight energy recharging. The framework aims to reduce the delivery time of multiple packages by minimizing the number of stops for recharging at intermediate stations. We propose to use *support drones* to recharge other drones in the swarm during their flight using energy services. In this respect, we propose two *energy services composition* methods, namely, Priority-based and Fairness-based methods. A re-ordering strategy for delivery drones is proposed to facilitate the in-flight energy-sharing process. An enhanced A* algorithm is implemented to compose the optimal services in terms of delivery time.

1.4.5 Flow-based Energy Service Composition

We propose a novel spatio-temporal service composition framework for crowdsourcing *multiple* IoT energy services to cater to *multiple* energy requests. We define a new energy service model to leverage the *wearable-based* energy and wireless power transfer technologies. We reformulate the problem of spatio-temporal service composition to provision multiple energy requests as a *matching problem*. We leverage the *fragmented* nature of energy to offer *partial* services to maximize the utilization of energy services.

We propose *EnergyFlowComp*, a modified Maximum Flow matching algorithm that efficiently provisions IoT energy services to accommodate multiple energy requests. Moreover, we propose *PartialFlowComp*, an extension of the *EnergyFlowComp* approach that considers the *partial-temporal overlap* between services and requests in provisioning. We conduct an extensive set of experiments to assess the effectiveness and efficiency of the proposed framework.

1.4.6 Recommendation-based Crowdsourcing of Energy Services

We propose a novel *recommendation-based* approach to proactively and efficiently distribute energy providers to balance energy distribution in a smart city. Our approach utilizes the mobility patterns of providers as a tool to identify their spatio-temporal preferences, that is, the times and places of their most frequent visits. The proposed approach leverages the provider's preferences to *reduce the cost of incentivizing* providers while effectively distributing them. We evaluated the effectiveness and efficiency of our approach using a real dataset.

1.4.7 QoE-aware Energy Service Composition

We propose a novel *Quality of Experience (QoE)* aware framework to crowdsource IoT energy services efficiently. The proposed framework leverages the provisioning of energy services as an auxiliary to increase consumers' *satisfaction*. A novel QoE model is developed as a metric to assess the consumers' satisfaction with the provisioning of energy services. Two novel composition algorithms, namely Partial-Based (PB) and Demand-Based (DB) approaches, are proposed to ensure the highest QoE for consumers. Both approaches leverage the providers' *flexibility* and *shareable* nature of energy services to allocate services and optimize the QoE efficiently.

1.4.8 Preference-aware Energy Service Composition

We extend the previous QoE framework by proposing a *CP-Net-based preference model* to capture providers' provisioning preferences. Two novel composition algorithms, namely Importance-based and Heuristic-based approaches, are proposed to ensure the highest QoE for consumers. Moreover, the Heuristic-based approach minimizes the cost of service provisioning. Experiments are conducted to evaluate the proposed approaches' effectiveness and efficiency.

1.4.9 Constraint-aware Energy Service Composition

We extend the previous preference-aware framework by considering the constraints of providing energy services. The constraints include the fluctuating availability of IoT energy service providers, coupled with the super-provider's limited reward budget. We propose a *PCP-Net-based preference model* to capture providers' fine-grained provisioning preferences and availability. We present a new Mixed-Integer Linear Programming (MILP) formulation for QoE-based energy service provisioning. We use several solvers to find the optimal service composition by leveraging providers' preferences.

1.4.10 Context-aware Energy Trustworthy Service Composition

We propose a novel *context-aware trust assessment* model to evaluate the trustworthiness of providers. Our model adapts to the dynamic nature of energy service providers and the super-provider's need to maintain QoE by selecting trustworthy providers. The proposed model evaluates providers' trustworthiness in various contexts, considering their behavior and energy provisioning history. Additionally, a trust-adaptive composition technique is presented for optimal energy allocation.

1.5 Thesis Outline

The rest of the thesis is organized as follows:

Chapter 2 presents a comprehensive background of crowdsourcing energy services from three areas: crowdsourcing IoT services, crowdcharging wireless energy, and service selection and composition. We describe the related work to our research in each of these domains. We also closely examine the quality of experience and how it has been modeled in the literature.

In Chapter 3, we propose a context-aware incentive model to encourage providers to participate. The incentive model considers several factors in determining the reward amount. We also propose an incentive-based energy requests composition approach.

Chapter 4 presents a reliability model to evaluate energy service requests. The proposed model is based on consumers' behavior and history of energy requests. Adaptive and priority scheduling approaches are presented to compose the most reliable energy requests, ensuring the maximum incentives for providers.

Chapter 5 proposes two components to demonstrate the feasibility of deploying the energy service ecosystem. The first component is a platform demonstrating the wireless energy transfer between smartphones. Additionally, the platform monitors and collects data from the energy-sharing process. Moreover, it includes a wireless energy estimation framework. The second component is an activity-based profiling framework employed to estimate harvested energy. In addition, it includes a study on the impact of incentives on consistent activity reporting.

Chapter 6 explores the impact of composing energy services for drone swarm delivery. We propose a novel framework for swarm-based drone delivery services with in-flight energy recharging. We propose the use of *support drones* to recharge other drones in the swarm during their flight using energy services. We also presented two *energy services composition* methods, namely, Priority-based and Fairness-based methods.

In Chapter 7, we propose a spatio-temporal service composition framework to crowdsource *multiple* IoT energy services to cater to *multiple* energy requests. We propose *EnergyFlowComp*, a modified Maximum Flow matching algorithm that efficiently provisions IoT energy services to accommodate multiple energy requests. Moreover, we

propose *PartialFlowComp*, an extension of the *EnergyFlowComp* approach that considers the *partial-temporal overlap* between services and requests in provisioning.

Chapter 8 proposes a *recommendation-based* approach to efficiently distribute energy providers to balance energy distribution in a smart city. Our approach utilizes the mobility patterns of providers as a tool to identify their spatio-temporal provisioning preferences. The proposed approach leverages the provider's preferences to *reduce the cost of incentives* while effectively distributing them.

In Chapter 9, we propose *Quality of Experience (QoE)* aware framework to crowd-source IoT energy services. A novel QoE model is developed to assess consumers' satisfaction with energy services. Partial-Based (PB) and Demand-Based (DB) approaches are proposed to ensure the highest QoE for consumers. Both approaches leverage the providers' *flexibility* and *shareable* nature of energy services to allocate services and optimize the QoE.

Chapter 10 extends the previous QOE framework by proposing a *CP-Net-based preference model* to capture providers' provisioning preferences. Importance-based and Heuristic-based approaches are proposed to ensure the highest QoE for consumers.

In Chapter 11, we extend the previous preference-aware QoE-based framework by considering the constraints of provisioning energy services. We propose a novel *PCP-Net-based preference model* to capture providers' fine-grained provisioning preferences and availability. We present a new Mixed-Integer Linear Programming (MILP) formulation for QoE-based energy service provisioning and use several solvers to find the optimal service composition.

Chapter 12 proposes a *context-aware trust assessment* model to evaluate the trustworthiness of providers. The proposed model evaluates providers' trustworthiness in various contexts, considering their behavior and energy provisioning history. Additionally, a trust-adaptive composition technique is presented for optimal energy allocation.

In chapter 13, we conclude the thesis by summarizing our work and discussing directions for future research.

CHAPTER 2

Literature Review

Crowdsourcing energy services comes from the intersection of three areas: crowdsourcing IoT services, wireless energy sharing, and service selection and composition. This chapter highlights the challenges in crowdsourcing IoT services in Section 2.1. Next, we present the current techniques for wireless energy sharing in Section 2.2. Then, we discuss spatial-temporal service selection and composition approaches in Section 2.3. Moreover, as previously stated, we propose the use of energy services as a tool to ensure consumers' Quality of Experience (QoE) in microcells (i.e., coffee shops). Therefore, in the last section (Section 2.4, we closely examine QoE and how it has been modeled in the literature. This chapter is designed to foster a comprehensive understanding of the background of crowdsourcing energy services.

2.1 Crowdsourced IoT Services

This section presents the primary challenges associated with the crowdsourcing of Internet of Things (IoT) services - a fundamental pillar of crowdsourcing energy services. The key areas of investigation encompass incentives, reliability assessment, and the assessment of trust in IoT services.

2.1.1 Incentives in crowdsourcing

Several studies used incentives to engage users in crowdsourcing and crowdsensing tasks [54][53]. There are a variety of incentive-based approaches, including auction-based, reputation-based, services as incentives, social incentives, and gamification. An

auction-based system is a platform where a user (i.e., auctioneer) posts a task on the system, then other users (i.e., bidders) bid to complete the task [55]. The system then selects a user (i.e., winner) from the bidders based on the objective of the task requester. Reputation-based incentives are used to enhance the quality of the completed task or service [55]. Ratings of the users based on their completed tasks motivate them to provide better quality. Services were used as incentives to motivate users to provide services, in order to request them later as consumers [56]. Social incentives leverage social networks and events among users to promote global cooperation [57]. Gamification turns tasks into playable games to attract and motivate users to provide their services [58].

To the best of our knowledge, there is no previous work done on incentivizing users to provide wireless energy services [48]. In this regard, we propose a context-aware incentive model to address the resistance to providing energy services. Additionally, we propose incentive-based selection and composition approaches.

2.1.2 Reliability assessment in IoT services

Reliability of services has been widely explored in the literature, including web and mobile sensing services [59][60][61]. A proposed approach to determine the reliability of web services suggested two methods: neighborhood-based and model-based [59]. The neighborhood-based approach uses past failure data of similar neighbor services to predict the Web service's reliability. The model-based approach uses a factor model built on the data of service failure; it then uses this factor model to make a reliability prediction. Another approach to evaluate the reliability of mobile sensing services proposes the use of a link reliability factor [60]. The factor determines the reliability of routes between a provider and a requester. Measuring the reliability in mobile sensing services included the reliability of providers to produce accurate data [62]. Another work suggested measuring the reliability and accuracy of the provided service by the repetition of performing a task and a majority voting system [62]. Another approach proposed a reliability assessment method to assign sensing tasks to vehicles [63]. The

reliability method evaluates vehicles based on previously completed sensing tasks in terms of acceptable task performance and accuracy of sensed data.

To the best of our knowledge, all the aforementioned approaches are not applicable to the energy services environment because of the environment's context, such as the mobility of IoT users and the IoT's limited resources. In addition, composing reliable energy service requests while using incentives is yet to be addressed [48]. In this thesis, we propose a reliable incentive-driven composition to increase participation in provisioning energy.

2.1.3 Trust assessment in IoT services

Trust assessment in crowdsourced IoT service environments is fairly new. Most of the proposed approaches rely on either previous experiences [64] or social relations [65][66] to assess trust. The work in [64] proposed a framework that assesses service providers based on their reputations collated by a central authority. A framework was proposed in [65] that aims to eliminate the privacy risks associated with public Wi-Fi hotspots. The proposed framework lacks generality as it can only be used for Wi-Fi hotspot services. An approach for evaluating the trust between users in social networks is presented in [66], where behavioral interactions are used to indicate the level of trust (i.e., conversations between users and message propagation). However, the aforementioned approaches may not be suitable for IoT crowdsourcing environments. Trust assessment in IoT crowdsourcing environments is challenging because of specific characteristics, such as the diversity and anonymity of IoT users and devices and the absence of a central managing authority [40]. These characteristics make it more difficult to acquire an accurate measure of trust. Recent solutions proposed to assess trust from a usage-based perspective [40][67].

These frameworks are not applicable in IoT energy environments due to the energy scarcity, fluctuating behavior of IoT users, and influence of super-provider expectations on trust scores [50]. *To the best of our knowledge, no solution assesses the*

trustworthiness of providers' provisioning while considering the super-providers' expectations and requirements. We propose a context-aware trust assessment model and a trust-adaptive composition technique for optimal energy allocation.

2.2 Wireless Energy Sharing

This section presents the concept of wireless energy sharing, which is divided into two parts. The first part, titled 'Technology and Applications', examines the technologies enabling the wireless transfer of energy and their practical applications in daily life. The subsequent part, named 'Energy Sharing Techniques', explores the various strategies employed to share energy efficiently and effectively.

2.2.1 Technology and applications

Wireless energy charging is growing rapidly in the fields of wireless sensor networks, the internet of things, mobile social networks, vehicular ad hoc networks, and UAV networks [48][14]. The advent of wireless charging makes the harvested energy from IoT devices more flexible and convenient to be easily shared. Energy sharing helps create self-sustained systems. Different techniques have been developed for wireless charging in the internet of things and wireless sensor networks [68]. The most common methods are magnetic inductive coupling, magnetic resonant coupling, and microwave radiation. These techniques are used in wireless sensor networks by deploying charger robots in the network to charge the low-battery sensors [69]. A new radio wave-based uncoupled wireless charging paradigm has enabled sharing and accessing harvested energy from IoT devices [70].

2.2.2 Energy Sharing Techniques

Energy sharing is used for several purposes in IoT environments, including utilizing energy, balancing energy distribution, and energy sharing for content delivery [48].

Energy utilization intends to charge from other devices' batteries to reduce reliance on outlets [71]. Balancing energy distribution is accomplished in a mobile network by exchanging energy among devices [72] or by using a star network design [73]. Additionally, energy sharing is used in content delivery where nodes forward messages to a destination node [74]. The message is transferred with energy to encourage the nodes to carry the content to the destination node. A reliable energy supply method was proposed to charge low-battery IoT devices using a mobile charger [69]. Another approach proposed the use of drones to charge rechargeable sensor networks wirelessly [75].

Wireless energy sharing has been proposed in other fields, such as electric vehicles (EV). A recent study presents a privacy-preserving algorithm for EV charging, which uses BMNN assignments and homomorphic encryption to match demand and supply efficiently while maintaining user privacy in dynamic environments [76]. Another research introduces a decentralized peer-to-peer trading mechanism for bidirectional energy exchange between EVs and wireless charging-discharging lanes (WCDLs) [77]. The mechanism focuses on optimizing cost functions with a privacy-preserving protocol. Further, a blockchain-based electricity trading system for EVs proposes a cooperative game algorithm for transaction pricing and user matching [78]. Lastly, a study explores bi-directional energy transfer among EVs on repetitive routes [79]. The study converts the energy distribution challenge into linear programming problems.

2.3 Energy Services Selection and Composition

In this section, we explore the intricacies of selecting and composing energy services. To provide a more comprehensive understanding, we have divided this topic into two subsections: 'Services Selection and Composition' and 'Energy Services'.

2.3.1 Services Selection and Composition

Selecting and composing services is one of the techniques used in service computing. Service composition is utilized in different domains such as cloud computing [80], social networks [81], and the Internet of Things [82][83]. Services are composed according to their *functional properties* and consumer preferences (QoS). Most service composition methods convert the composition into a resource scheduling or an optimization problem. Resource scheduling in service composition has been extensively researched [84]. The fundamental parameters of resource scheduling algorithms comprise the target to optimize, e.g., computing or storage resources in cloud computing, and the scheduling priority policy based on time. For example, the priority for the Short Job First (SJF) scheduling algorithm is that the shortest jobs are to be scheduled first [85]. Service composition is mostly driven by resource utilization maximization or minimization of scheduling time [86][87]. Different optimization algorithms have been utilized for service composition, such as integer programming, genetic algorithm (GA), and particle swarm optimization (PSO) [86][87].

The composition of crowdsourced services should consider two aspects, the spatio-temporal features of services and their mobility of IoT devices [23]. Usually, crowdsourced services are from different sources (mobile and static devices). The functional properties of the provided services should conform to the spatio-temporal features of the query. Consumer preferences have to be met by the QoS of the provided services. Crowdsourcing energy as a service is converted to a QoS-aware service composition problem [25][14]. The spatio-temporal features of energy services are considered QoS attributes (i.e., start and end time, duration, and location of an energy service) [23]. Composing energy services relies on finding the optimal selection of nearby services that fulfill a consumer's required energy within their query duration. Existing composition techniques may not be applicable to compose energy services due to the uniqueness of the crowdsourced IoT environment. For example, an IoT user may consume only a part of the advertised energy service [14].

To the best of our knowledge, existing literature in service composition does not apply to energy services due to its dynamic nature and limited resources [14][13]. In this thesis, we proposed several models and composition approaches to address challenges in composing energy services, such as provision resistance, energy utilization, and mobility.

2.3.2 Energy Services

Energy services have been introduced as a ubiquitous alternative solution to charge IoT devices [14][48]. Several studies have addressed challenges related to fulfilling the requirements of energy consumers [25][49][50]. Energy Services and energy query models are proposed to capture the attributes of each service [23]. A spatio-temporal composability model is proposed to select the best set of energy services to fulfill an energy consumer's request. A temporal composition algorithm is proposed to compose energy services to fulfill a consumer's energy requirement [25]. The algorithm proposes the use of partial services and fractional knapsack to maximize the provided energy. An elastic composition was proposed to address the reliability of highly fluctuating energy providers [49]. The composition uses the concepts of soft and hard deadlines to extend the stay of a consumer and select more reliable services. Mobility pattern impact on IoT energy services was addressed in [50][39]. The intermittent behavior of energy services was addressed by a fluid approach [50]. The approach uses the indoor mobility patterns of the crowd to predict the intermittent disconnections in energy services and then replace or tolerate these disconnections. A proactive composition approach is proposed to avoid the under-provision of energy requests in the microcells [39]. The approach uses the outdoor mobility patterns and the energy distribution of energy consumers to determine when and where to request energy. However, the authors assume prior knowledge of the energy distribution of the microcells. This assumption is yet to be addressed. In addition, they assume that consumers will always visit a microcell that has oversupplied energy.

To the best of our knowledge, existing literature in energy services addresses issues from a consumer perspective [14][13]. Challenges related to providers and super-providers are yet to be addressed. In this thesis, we proposed several models and composition approaches to address challenges from both provider and super-provider perspectives, including provision resistance, energy utilization, trust, and reliability.

2.4 Quality of Experience

In this thesis, we propose the use of energy services as a tool to ensure consumers' Quality of Experience (QoE) in microcells (i.e., coffee shops). Therefore, in this section, we take a close look at QoE and its representation in the existing literature.

Quality of experience has several definitions in the literature based on the field of research, such as networking, multimedia, and human-computer interaction [88][89][11]. All existing definitions emphasize the assessment of the quality of an application or service based on end-user perception. Moreover, most studies have concentrated on evaluating user experience quality for multimedia applications. For example, a method was proposed to evaluate gaming QoE under system-influencing factors, such as delay, packet loss, and frame rates [90]. Another study introduced a model-based approach to measure QoE for VoIP services [89]. The model uses five quantifiable quality measures, including integrality, retainability, availability, usability, and instantaneousness. Another study proposed 'Kaleidoscope' as an automated solution to evaluate Web features [91]. Improving the QoE for restaurant customers using augmented reality was proposed by [92]. This study introduced an interactive smart table to enhance QoE in terms of entertainment, socialization, food selection, and ordering. In conclusion, the existing research focuses on assessing the QoE of a service based on the perception of the end-users.

To the best of our knowledge, assessing the QoE in energy services is not yet explored. In addition, using energy services as a tool to enhance QoE in microcell-based services

is yet to be addressed. This thesis is the first attempt to compose energy services with the objective of enhancing the QoE for consumers.

2.5 Conclusion

This chapter discussed the details of crowdsourcing energy services and provided a broad overview of the relevant concepts. We presented a review of crowdsourcing challenges and the proposed approaches to address them in Section 2.2. Section 2.2 presented a detailed review of existing techniques used for crowdcharging wireless energy. Section 2.3 presented a review of the existing spatial-temporal energy service selection and composition approaches. Finally, we presented a review of existing techniques to assess the quality of experience in Section 2.4.

Incentive-based Selection and Composition of IoT Energy Services

3.1 Introduction

In this chapter, we propose an incentive model that considers the characteristics that affect the willingness of the provider to participate. An incentive model relies on *intrinsic or extrinsic* rewards to motivate participation. Intrinsic reward is an internal satisfaction motivated by altruistic purposes. In contrast, extrinsic reward is motivated by materialistic compensations [53]. This work uses an incentive model to compute the *extrinsic* reward. *Our incentive model will not consider intrinsic rewards as they are self-motivated.* The incentive model will consider the *contexts* of the providers and consumers that may affect the provider's willingness, such as the amount of requested energy and the required time to complete the energy service. *Was e aim to compose the best set of energy requests using our incentive model.*

3.2 Motivating Scenario

We consider an *incentive-based composition for energy service requests as a motivating scenario* (see Fig. 3.1). Assume that a geographic area, such as a building or a mall, is divided into *microcells*, where a microcell is a confined area such as a coffee shop or a restaurant (see Fig. 3.1(A)). Assume several IoT devices (providers and consumers) are distributed in a microcell (see Fig. 3.1(B)). The devices are assumed to have wireless energy transmitters and receivers such as Energous¹. The distance

¹www.energous.com

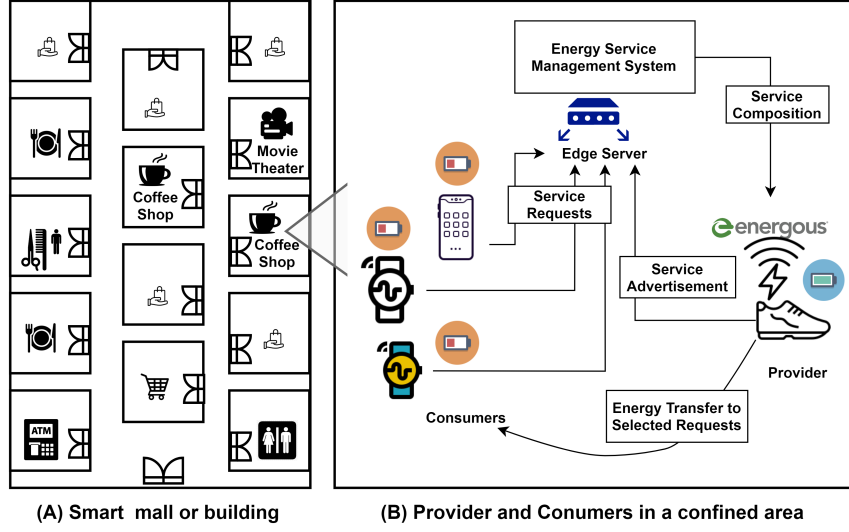


FIGURE 3.1. Wireless energy sharing motivating scenario

between devices exchanging energy may reach the maximum distance (15 feet using Energous) to ensure successful wireless transmission. An owner of a device such as a smart shoe with spare energy would like to share the energy with available IoT devices. Other devices may request energy to charge themselves. All local energy requests and advertisements are processed at the edge, e.g., a router associated with the microcell. Consumers may have different requirements regarding requested energy, charging speed, time availability, and location. The provider may resist offering their spare energy even if they had the intention because of multiple reasons. This may include the fear of needing excess energy at a later time. The provider may get motivated to give energy if an incentive is offered. Providers may be incentivized differently. For example, a provider may prefer providing the service in the shortest time. A second provider may desire an extrinsic reward (e.g., monetary). Another provider may want to give all their energy to contribute to a green environment. *We aim to select and compose energy service requests that will maximize the extrinsic reward of the provider.*

We focus on *composing energy service requests to maximize the reward of the energy provider*. We assume a single energy provider may share their energy with multiple energy consumers within a *specific time interval*. The *composition* of energy service

requests includes selecting the optimal requests that maximize the provider's reward. The reward for every energy request is calculated using an incentive model. We assume a *static environment* where the provider and consumers do not move during the composition. To the best of our knowledge, no approach addresses the problem of resistance in providers for participation while composing energy service requests [48]. The main contributions of this chapter are:

- A Spatio-temporal selection of energy service requests.
- A scheduling-inspired incentive-driven composition of energy service requests.
- A novel incentive model considering consumers' EaaS requests and providers' QoS that increases providers' participation.

3.3 System Model and Definition

We present the service-oriented architecture (SOA) for incentive-based wireless energy services (see Fig. 3.1). The provider publishes its services to the edge to advertise them. Similarly, consumers submit their energy requests. When the provider submits its service advertisement, the service management system determines the reward for each energy request using an incentive model. Additionally, the system selects and composes the set of energy requests that will maximize the provider's reward. Then the system will notify the provider of the composition. Then, the provider will share their energy according to the composition with the selected consumers. Finally, the system will provide a reward for the provider in terms of credits.

We propose a formal model of our incentive-based energy service request composition. We consider the scenario of energy sharing in a microcell during a particular time interval T . We use the following definitions to formulate the problem.

Definition 1: Energy-as-a-Service (ES). We adopt the definition of ES in [25]. An ES is defined as a tuple of $\langle E_{ID}, E_{OwnerID}, F, Q \rangle$, where:

- E_{ID} is a unique energy service identifier

- $E_{OwnerID}$ is a unique owner identifier
- F is the function of sharing energy by an IoT device owner $E_{OwnerID}$ via an IoT device d
- Q is a tuple $\langle P_{EC}, P_{Loc}, P_{ST}, P_{ET} \rangle$ where each attribute donates a QoS property of ES as following:
 - P_{EC} is the energy capacity the provider can share
 - P_{Loc} is the location of the provider $\langle x, y \rangle$
 - P_{ST} is the start time of the provider's stay in the microcell
 - P_{ET} is the end time of the provider's stay in the microcell

Definition 2: Energy Service Request (ER). An ER request is a tuple of $\langle ER_{ID}, ER_{OwnerID}, C_{BL}, C_{RE}, C_{ST}, C_{ET}, C_{Loc} \rangle$ where:

- ER_{ID} is a unique energy service request identifier
- $C_{OwnerID}$ is a unique consumer (owner) identifier
- C_{BL} is the consumer's battery level
- C_{RE} is the amount of requested energy by the consumer
- C_{ST} is the start time of the consumer's stay
- C_{ET} is the end time of the consumer's stay
- C_{Loc} is the location of the consumer

Definition 3: Incentive. The energy service provider receives an incentive reward R by giving energy. Each incentive is earned as stored credit points. The reward will be computed using an incentive model. Rewards will be determined and provided by the system. The rewards come in the form of credits to providers. We assume the system has a limit for the amount of requested energy. The provider may use the rewarded credits later to increase the limit to request energy as a consumer.

Definition 4: Incentive-based energy requests composition problem. We assume that, in a microcell, there exists an energy service ES and a set of n energy requests $ER = \{ER_1, ER_2, \dots, ER_n\}$ as shown in Fig. 3.2. The ES will be advertised by a provider P . The ES will be represented using the aforementioned Definition 1.

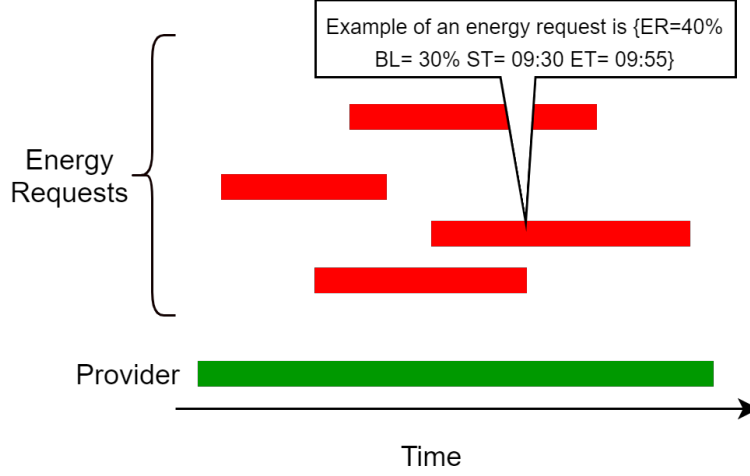


FIGURE 3.2. Example of energy requests and energy service

The energy requests are sent by consumers C . Each energy request (ER) is described using the aforementioned Definition 2. We formulate the composition of energy service requests into a service composition problem. Composing energy service requests needs to consider the spatio-temporal features of the service and requests. Composing energy requests for a provider's ES requires the composition of energy requests $ER_i \in ER$ where $[C_{ST_i}, C_{ET_i}] \subset [P_{ST}, P_{ET}]$, $\sum C_{RE_i} \leq P_{EC}$ and the provider reward $= \sum R$ is the maximum. We use the following assumptions and definitions to formulate the problem.

- The IoT devices are equipped with wireless energy transmitters and receivers.
- The composition considers the scenario of a single provider and multiple consumers.
- The provider has a fixed energy size during the composition.
- The provider and consumers may have different time windows but consumers' time windows must fall within the provider's time window $T_c \in T_p$.
- The provider and consumers have fixed locations $\langle x, y \rangle$ for the whole duration of the energy service.
- The provider transfers energy within their range (15 feet using Energous) to one consumer at a time.
- There is no energy waste while sharing.

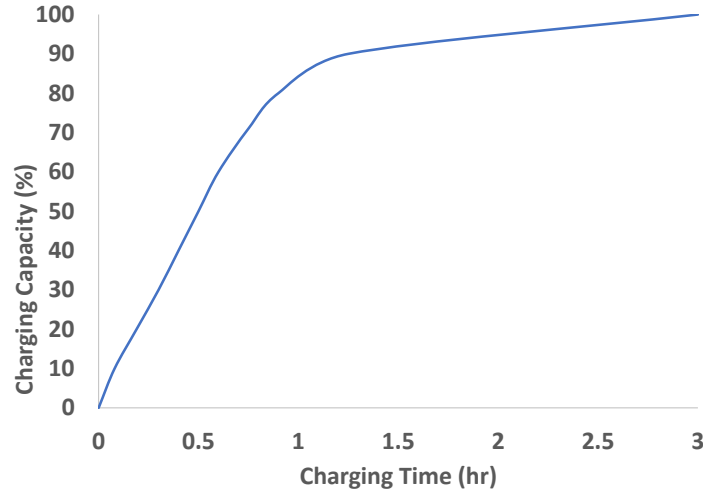


FIGURE 3.3. The charging time measured based on the capacity of lithium-ion batteries [93]

- The consumers' preferences, i.e., the reward they are willing to pay, will be part of their profile.

3.4 Incentive Model

We define the incentive model to compute the reward R . The incentive model considers the attributes of the consumers and provider that may influence the resistance of the provider to give energy. The below attributes will be used to compute the reward.

Attributes of Consumers' Context :

- *Battery level (C_{BL})*: The battery level affects the charging speed as shown in Fig. 3.3. A high charging speed encourages providers to give energy. The reward increases if the charging speed decreases. The charging speed is affected by the battery level of the consumer. If the battery level is less than 80%, the charging speed is high compared to the battery level between 80% and 99% [93]. Additionally, consumers with low battery levels will be at risk of shutting down. The reward of giving energy to a consumer with a low

battery should be more than if their battery level is high. We define the reward of the battery level ($Reward_{BL}$) as follows: **if** $C_{BL} < 20\%$ **or** $C_{BL} > 80\%$ **then** $Reward_{BL} = 1$ **otherwise** $Reward_{BL} = 0.5$.

- *Requested Energy (C_{RE})*: The amount of requested energy is considered in determining the reward. We only compute the reward of requested energy if the provider can afford it. The reward of the Requested Energy ($Reward_{RE}$) is defined as follows: **if** $P_{EC} \geq C_{RE}$ **then** $Reward_{RE} = C_{RE}/P_{EC}$
- *Stay Time*: If a consumer stays for a short time, they will rush to get the requested energy. For example, if two consumers have the same energy request and similar attributes but differ in availability time, priority will be given to the one available for a shorter time. Therefore, the shorter the stay time of a consumer, the higher the reward. We defined the reward of stay time as $Reward_{ST} = |C_{time} - P_{time}|/P_{time}$, where P_{time} represents the total minutes of the provider time and C_{time} represents the total minutes of the consumer time.

Attributes of Provider's Context

- *Time of Provision*: The time of providing energy influences the provider's willingness. For example, the provider may be motivated to give the energy at the start of his stay than at the end. This may occur due to having plenty of energy in the morning compared to the end of the day. We categorize the time of provision into four time periods TP that is $TP_1 = [09:00 \text{ a.m.} - 11:00 \text{ a.m.}]$, $TP_2 = [11:00 \text{ a.m.} - 01:00 \text{ p.m.}]$, $TP_3 = [01:00 \text{ p.m.} - 03:00 \text{ p.m.}]$, $TP_4 = [03:00 \text{ p.m.} - 05:00 \text{ p.m.}]$.

We use the crowdsourcing platform Amazon Mechanical Turk (MT_{urk}) to compute users' preferences at the time of provision². The use of MTurk is appropriate as incentives are human-centric. Moreover, MTurk was used by other works as a platform to validate incentive models [94]. We designed a

²www.mturk.com

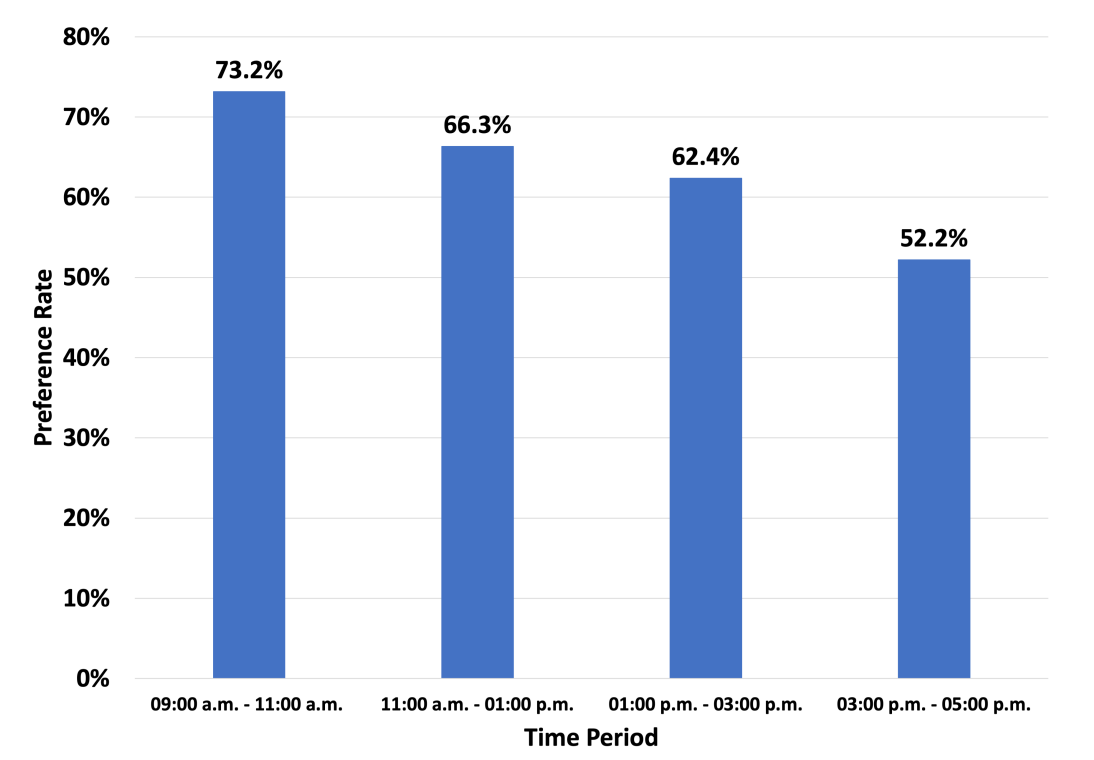


FIGURE 3.4. Users' temporal preferences to share energy

questionnaire that consists of a set of scenarios. Each scenario starts by describing the energy-sharing environment. We adopt an energy-sharing environment where users can share energy through wireless technologies. MTurK workers are asked to consider themselves as energy service providers. We asked MTurk workers to select their preferred time to accept energy requests from 09:00 a.m. to 05:00 p.m. Fig. 3.4 represents the users' temporal preference for sharing energy. The x-axis represents the defined periods. The y-axis shows the percentage of users who selected a certain time. For example, the figure shows that 73.2% of participants preferred the period from 09:00 a.m. to 11:00 a.m. We normalized the time preference statistics to compute the reward of the Time of Provision ($Reward_{TP}$) as follows: we compute the resistance rate of each period by subtracting the participation rate from 100,

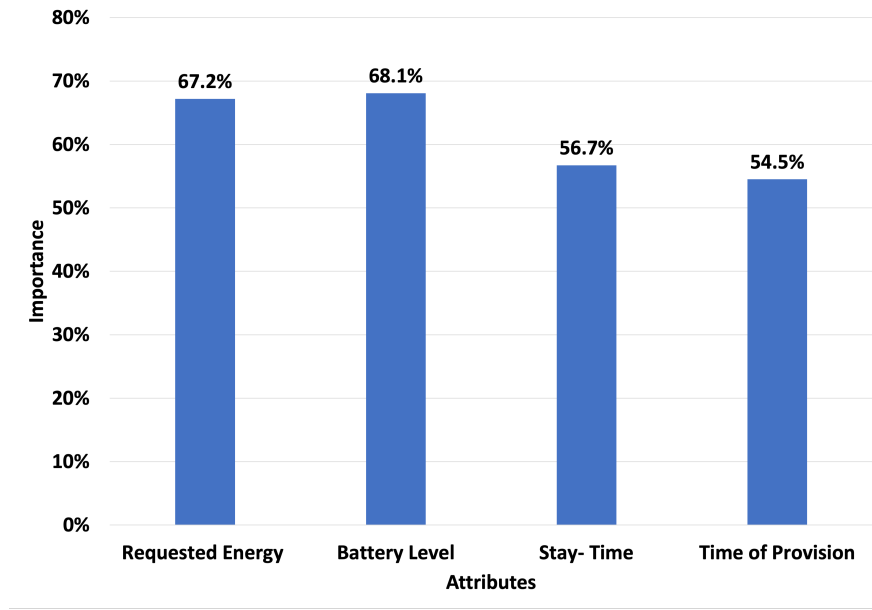


FIGURE 3.5. Average importance of incentive attributes based on user preferences for energy sharing

i.e., the resistance rate of $TP_1 = 100\% - 73.2\% = 26.8\%$. Then we normalized the resistance rate as the weight of the reward for TP_1 . The normalization is computed by dividing the resistance rate of TP_1 by the total resistance rates of all TP_s . We followed the same steps to calculate the reward of the Time of Provision ($Reward_{TP}$), which were as follows: **if $TP_i \in TP_1$ then $Reward_{TP} = 0.18$ if $TP_i \in TP_2$ then $Reward_{TP} = 0.23$ if $TP_i \in TP_3$ then $Reward_{TP} = 0.26$ if $TP_i \in TP_4$ then $Reward_{TP} = 0.21$**

Energy Request Total Reward: The total reward of an energy service request ($Reward_{ER}$) is the summation of all the rewards as follows:

$$Reward_{ER} = \sum_{i \in n} w_i \times Reward_i \quad (3.1)$$

where $n = \{BL, RE, ST, TP\}$ and w is the weight of each reward. w is computed using MTurk as previously indicated. We asked users to scale the importance of each attribute when they accept energy requests. The scale value is between 0% and 100%. 0% indicates an unimportant attribute, while 100% indicates a highly important attribute. Fig. 3.5 represents the average of the importance of each attribute. Then, we

normalized the attributes' importance to define their weights. We computed the normalization by dividing the importance of an attribute by the total importance of all the attributes. Therefore, the weight of each reward w is $\{0.27, 0.28, 0.23, 0.22\}$ in the same order of set n .

Provider Total Reward:

The total reward of the provider is the summation of the reward of the selected energy service requests as follows:

$$ProviderReward = \sum_{j=1}^m RewardER_j \quad (3.2)$$

where m is the number of selected energy requests ER that will receive the energy from the provider. The energy requests will be selected using our composition framework.

3.5 Composition Framework for Energy Service Requests

The framework of composing energy service requests will involve two phases: the selection of energy service requests and the composition of energy service requests (see Fig. 3.6). The first phase consists of selecting the energy service requests that are composable with the energy service. Then calculating the reward for each selected energy service request using the incentive model. The second phase is composing the energy service requests that maximize the reward for the provider. Our composition algorithm is inspired by Priority-based scheduling algorithms [95].

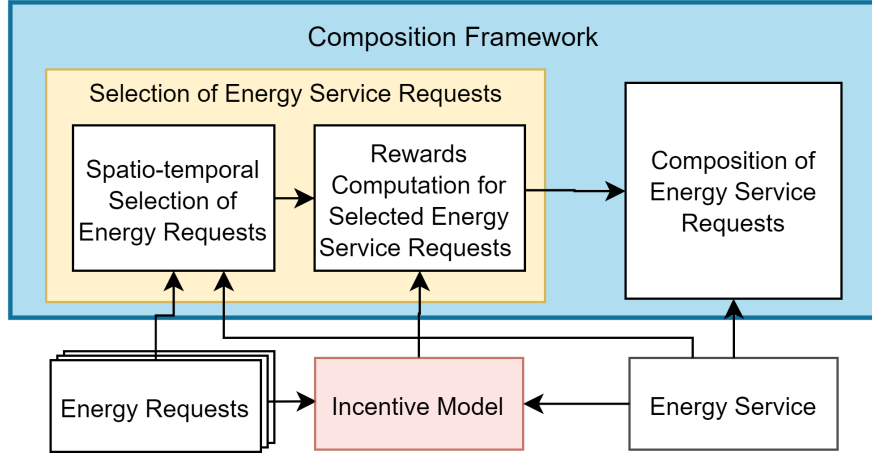


FIGURE 3.6. The composition framework for energy service requests

3.5.1 Selection of Energy Service Requests

The selection phase aims to select all the energy requests that are composable with the energy service of the provider. The selection phase consists of two stages: Spatio-temporal selection of energy requests and rewards computation for the selected energy requests (see Fig. 3.6). The first stage aims to select nearby energy service requests that the provider can serve. The selection is accomplished using the spatial and temporal features of *ES*. Spatial composability service is required to allow wireless energy transfer to occur between the consumers and the provider. An energy request is composable temporally if its duration falls within the time window of the energy provider. The last step in the first stage is to check if the *ES* can provide the requested energy. The second stage focuses on computing the reward for each selected energy service request. We compute the reward using the incentive model discussed in Section 3.4.

Phase 1 in Algorithm 3.1 describes the selection phase of energy service requests (Lines 1-7). For every energy service request, the algorithm checks if the time window of the energy service request falls in the time interval of the energy service (Lines 1-2). Line 3 calculates the distance between the energy request and the energy service. Line 4 checks if the distance between the energy request and energy service is below the maximum needed distance to transfer energy (15 feet using Energous). Line 5 checks

Algorithm 3.1 Incentive-based Composition of Energy Service Requests**Input:** $ES, ER, MaxEnergyDistance$ **Output:** $ER_{composition}, Reward$ **Phase 1: Selection of Energy Service Requests**

```

1: for  $ER_i \in ER$  do
2:   if  $C_{ST} \geq P_{ST}$  and  $C_{ET} \leq P_{ET}$  then
3:      $distance = \sqrt{(C_{loc.x} - P_{loc.x})^2 + (C_{loc.y} - P_{loc.y})^2}$ 
4:     if  $distance \leq MaxEnergyDistance$  then
5:       if  $C_{RE} \leq P_{EC}$  then
6:          $ER_{i.Reward} = Calculate\_Reward(ES, er_i)$ 
7:          $NearbyER.add(ER_i, ER_{i.Reward})$ 

```

Phase 2: Composition of Energy Service Requests

```

8:  $NearbyER_{sorted} = sort(NearbyER, starting_{time} : ascending, Reward : descending)$ 
9:  $Reward = 0$ 
10:  $Provider_{ST} = P_{ST}$ 
11:  $Provider_{EC} = P_{EC}$ 
12: for  $er_i \in NearbyER_{sorted}$  do
13:   if  $C_{ST} \geq Provider_{ST}$  and  $Provider_{EC} \geq C_{RE}$  then
14:      $Reward = Reward + er_{i.Reward}$ 
15:      $Provider_{ST} = C_{ET}$ 
16:      $Provider_{EC} = Provider_{EC} - C_{RE}$ 
17:      $ER_{composition}.add(er_i)$ 
18: return  $ER_{composition}, Reward$ 

```

if the energy service capacity is enough to provide the requested energy. Line 6 calculates the reward value of the energy request using the previously defined incentive model (Section 3.4). Line 7 adds the energy request with its reward value to the set of nearby energy service requests.

3.5.2 Incentive-based Composition of Energy Service Requests

The composition of energy service requests phase aims to compose the energy requests that incentivize the provider to share their energy. This is accomplished by selecting energy requests that maximize the provider's reward. Phase 2 of Algorithm 3.1 describes the composition of energy service requests (Lines 8-18). Line 8 sorts the nearby energy requests in ascending order based on the start time, and then in descending order based on the reward. Lines 12-14 check for each request if the available

energy of the provider can serve the requested energy, then it sums up the ER's reward as part of the provider's reward using equation (3.2). Line 15 updates the available energy of the provider by subtracting the energy request C_{RE} from the Provider energy P_{EC} . Line 16 updates the provider's time so that no energy requests overlap, as the provider can only share energy with one consumer. Line 17 adds the request to the set of composed energy requests.

3.6 Experimental Results and Discussions

We conducted two sets of experiments to test the proposed composition framework and incentive model. The first set of experiments evaluated the performance of the proposed Incentive-based composition algorithm. We evaluated the execution time and energy utilization of the proposed composition against other approaches. The second set of experiments evaluated the incentive model's attributes' relevance to users' resistance to sharing energy. Moreover, the experiments showed the effect of using rewards on increasing participation in sharing energy.

3.6.1 Evaluation of the Composition Framework

We evaluated the performance of the proposed Incentive-Based composition algorithm (IB) against the Brute Force approach (BF) and the First Come First Served (FCFS). We then conducted two sets of experiments to evaluate the execution time and energy utilization of each algorithm.

We used a dataset published by IBM for a coffee shop chain in New York City to mimic the crowd's behavior within microcells ³. The dataset consists of transaction records of customers' purchases in coffee shops for one month. Each coffee shop consists of, on average, 560 transactional records per day and 16,500 transaction records. We use the IBM dataset to simulate the spatio-temporal features of energy requests. Our experiment uses the consumer ID, transaction date, time, location, and coffee shop

³<https://ibm.co/2O7IvxJ>

TABLE 3.1. Experiments variables

Variables	Value
Total Energy Requests for coffee shop 1 in April	16,830
Energy Services	2,000
Duration of Services	10 - 200 minutes
Duration of Energy Requests	5 - 30 minutes
Provided Energy	50 - 100 %
Requested Energy	1 - 100 %
Battery Level	1 - 80%

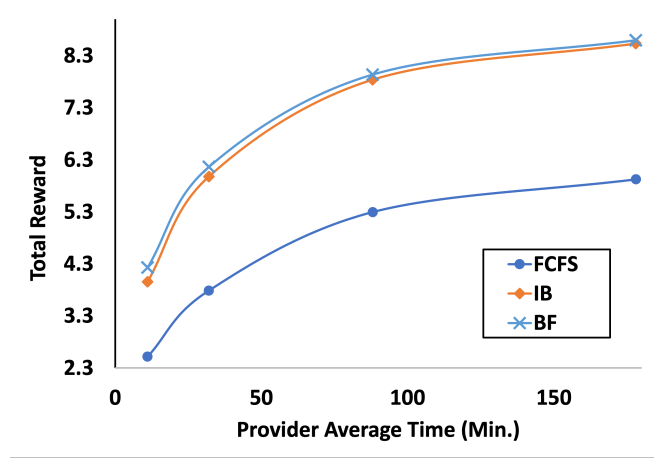


FIGURE 3.7. Average total reward for IB, BF, and FCFS approaches.

ID from each record in the dataset to define the spatio-temporal features of energy services and requests, e.g., start and location of energy service or a request. To model the energy aspect, we randomly assigned a battery level C_{BL} and requested energy capacity C_{RE} for each request, with values ranging from 1% to 100%. Furthermore, we generated 2,000 energy service advertisements with random energy capacity P_{EC} , location P_{Loc} , and time $P_{ST,ET}$. For each energy service, we used our framework to compose the energy requests that maximize the reward of the service's provider. Table 3.1 summarizes the experimental variables.

In the first experiment, we compared the average total reward of the proposed Incentive-Based composition algorithms (IB) against the Brute Force approach (BF) and First Come First Served (FCFS) [95]. We considered the Brute Force approach as the baseline. For the Brute Force approach, we retrieved all the possible compositions

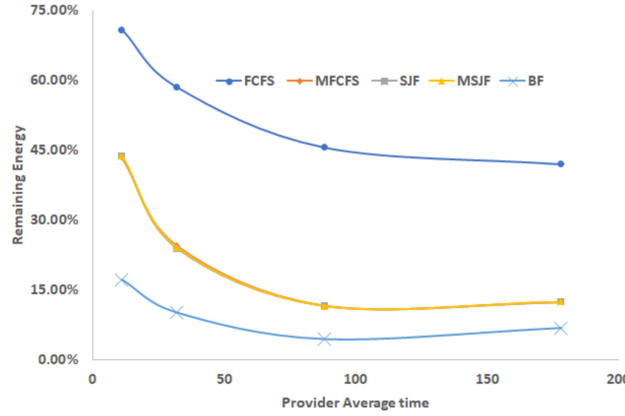


FIGURE 3.8. Average of providers' remaining energy for IB, BF, and FCFS approaches

of the energy service requests. Then, we selected the composition that had the maximum total reward. In the case of FCFS, the approach selects the energy requests based on their start time regardless of their reward value [95]. Fig. 3.7 presents the average total reward for each algorithm—the x-axis in Fig. 3.7 represents the average of the provider's staying time. The proposed algorithm (IB) performed better than the FCFS in terms of total reward. Our proposed algorithm maximizes, on average, the rewards by 22% compared to the FCFS as it considers the reward value of each energy request. The Brute Force (BF) gave the best result, but the proposed algorithm gave close results.

In the second experiment, we compared the average of the remaining energy of the providers after composition using the proposed Incentive-Based composition algorithms (IB) against the Brute Force approach (BF) and the First Come First Served (FCFS). The experiment tested which approach has better energy utilization. Fig. 3.8 shows the average remaining energy of the providers for each algorithm. The x-axis in Fig. 3.8 represents the average of the provider's staying time. The graph shows that the proposed algorithm (IB) performed better than the FCFS in utilizing the provider's energy. Our proposed algorithm minimizes, on average, the remaining energy by 31% as it considers the reward value of each energy request. As previously mentioned, the reward value considers the energy request's size.

3.6.2 Evaluation of the Incentive Model

We present a set of experiments to evaluate the effectiveness of the incentive model. As indicated in Section 3.4, we used MTurk to validate the attributes' impact on users' participation in sharing energy. Moreover, MTurk is used to show the effect of using rewards on increasing participation in sharing energy. We designed a questionnaire that consists of a set of scenarios ⁴. These scenarios are presented to MTurk workers to test each attribute, such as energy request size and time of provision. Each scenario started by describing the energy-sharing environment. We adopted an energy-sharing environment where users can share energy through wireless means. MTurK workers were asked to consider themselves as energy service providers. Moreover, workers were asked to accept or reject requests by varying the attributes to measure the resistance of each attribute.

All the described experiments in this subsection were designed with the following structure: Given a certain scenario, if a worker rejected a request, we asked them if an extra reward would make them accept that request. If a worker still rejected the request, we asked them if the reward amount would change their decision. A total of 175,000 questionnaires were answered by workers at MTurk. All the experiments in this subsection compared the participation rate of users in sharing energy without an incentive (No reward), with an incentive (Reward), and with deciding the amount of incentive (Amount of reward).

In the first experiment, we asked Mturk workers to accept or reject energy requests with different energy request sizes. The energy request size varied from 10% to 90% of the provider battery by increasing 10% for each question. Fig. 3.9 presents the participation rate of users in sharing energy with different requests. The x-axis represents the energy requests' size. The y-axis shows the percentage of users who accepted the energy request. The figure reflects the effect of the energy request size attribute on

⁴tinyurl.com/vnz6ayl

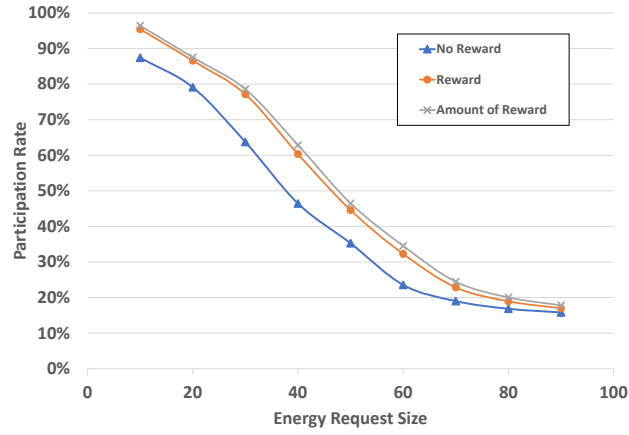


FIGURE 3.9. Participation rate of users in sharing energy with and without a reward

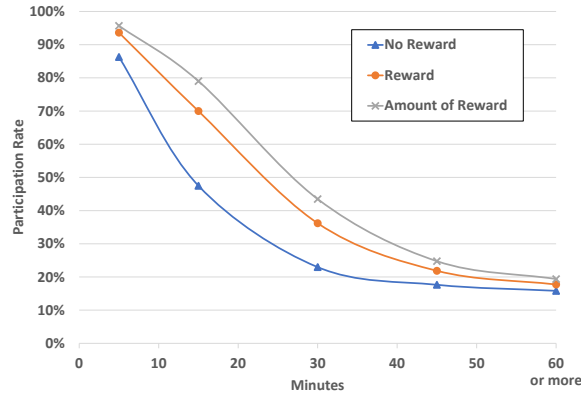


FIGURE 3.10. Participation rate of users in sharing energy with and without a reward

the willingness to share energy. Additionally, the figure shows that providing a reward increased users' participation in sharing energy.

In the second experiment, we asked Mturk workers to accept or reject energy requests with different charging times. The charging time was used to test the effect of the battery level attribute. As previously indicated, the battery level affects the speed of charging, which affects the time to charge a device. The energy request charging time varied between 5 to 60 minutes. Fig. 3.10 presents the participation rate of users in



FIGURE 3.11. Participation rate of users in sharing energy with and without a reward

sharing energy with different requests. The x-axis represents the energy requests' charging time. The y-axis shows the percentage of users who accepted the energy request. The figure reflects the effect of the charging time on sharing energy. Additionally, the figure shows that giving a reward increased users' participation in sharing energy.

In the third experiment, we asked Mturk workers to assume they were in a coffee shop and willing to share energy. Additionally, we asked them to select their preferred period to share energy. If a period was not selected, we asked them if an extra reward would make them accept the rejected period. If a worker still rejected the request, we asked them if the reward amount would make them change their mind. Fig. 3.11 presents the participation rate of users in sharing energy with different requests. The x-axis represents the time to provide energy for a request. The y-axis shows the percentage of users who accepted the energy request. The figure shows that giving a reward increased users' participation in sharing energy.

3.7 Conclusion

We proposed an Incentive-based energy requests composition framework. A new incentive model was designed that considers the attributes that affect the willingness of providers to share their energy. An incentive-driven composition of energy service requests was proposed. The approach selects the energy service requests that maximize the provider's reward to overcome their resistance. The efficiency of the proposed approach was tested against Brute Force (BF) and First Come First Served (FCFS) approaches. Experimental results showed that the proposed composition outperformed the FCFS approach in maximizing the reward and outperformed the BF approach in execution time, while providing similar rewards for the provider. Future direction is to improve the framework to accommodate different incentive preferences.

Reliability-based Energy Service Composition

4.1 Introduction

IoT environments are highly dynamic, i.e., IoT devices move freely between microcells. In the case of energy sharing, movements in and out of microcells may disrupt the energy transfer. Energy disruption, caused by movement, may be categorized as *voluntary* or *involuntary*. Voluntary disruption is an *explicit* decision by the consumer to discontinue the energy transfer. Involuntary disruption refers to an *unintended* temporary discontinuity in the energy transfer by the consumer. In the case of voluntary disruption, a selected energy request may be discontinued for several reasons, including canceling the request or leaving the range of energy transfer. In the case of involuntary disruption, a consumer may unintentionally leave intermittently, e.g., moving in and out of the microcell. In crowdsourcing energy environments, rewards, such as credit points, are used to encourage providers to share more energy [96]. The disruption of a selected request may reduce the amount of shared energy and, thereby affecting the provider's reward. Therefore, a re-composition is required to ensure the provider receives an optimal reward. However, a re-composition may not be guaranteed because an optimal replacement is dependent on the availability of other consumers' requests.

In this chapter, reliability is proposed as a measure to gauge the level of commitment from energy consumers. *Commitment* is defined as the *likelihood* of a consumer completing the consumption of an energy request. We will use past consumers' history to assess the level of commitment. We propose a *consumer-side* reliability model using

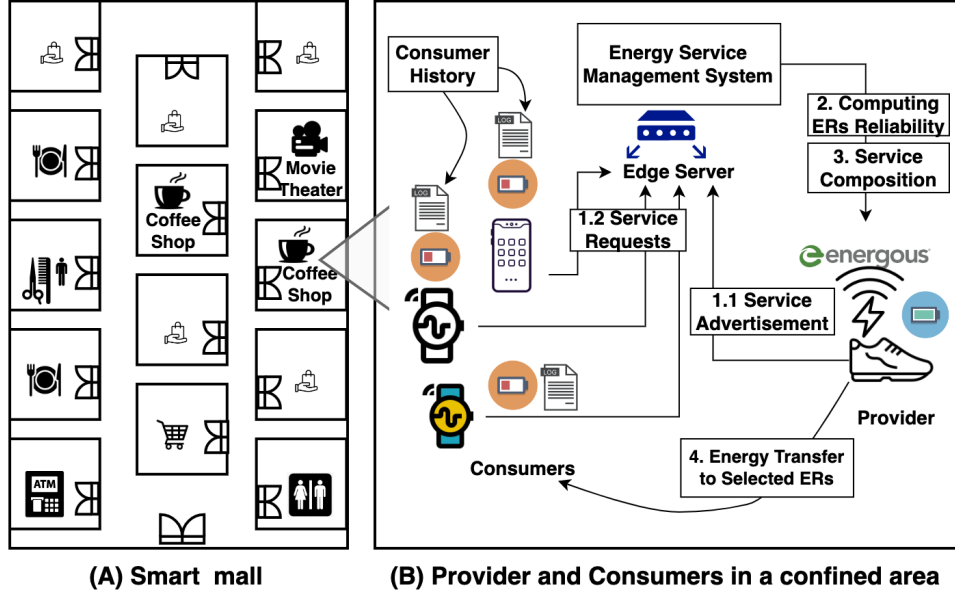


FIGURE 4.1. Reliability-based wireless energy sharing system

the level of consumers' commitment as a key indicator for computing reliability. We focus on selecting and composing the best set of energy requests using our reliability model. The main contributions of this chapter are:

- A novel reliability model considering consumers' history and mobility model.
- A spatio-temporal reliable selection of energy requests.
- A scheduling reliability-based incentive-driven composition of energy service requests.
- An adaptive composition to maximize the reliability of consumers and the provider's reward.

4.2 Motivating Scenario

We consider the scenario as shown in Fig. 4.1. Assume that a mall is split into geographical *microcells* such as cafes, restaurants, movie theaters, etc. (see Fig. 4.1(A)). We also assume there are several IoT devices and wearables (provider and consumers) in a microcell (see Fig. 4.1(B)). The devices are assumed to have wireless energy

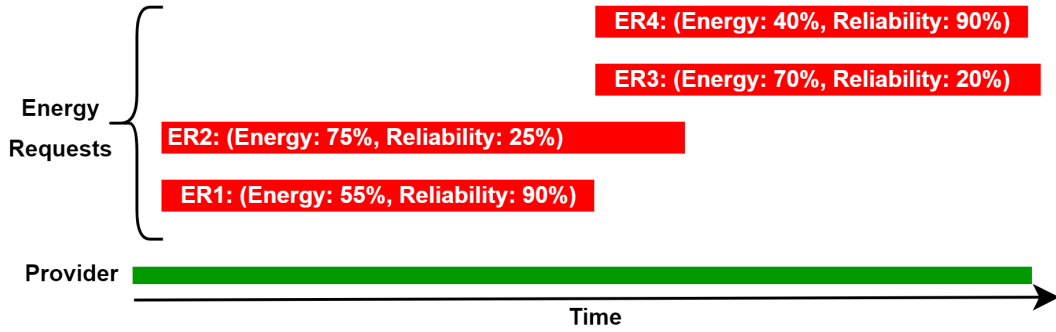


FIGURE 4.2. Example of energy requests with reliability

transmitters and receivers such as "Energous"¹. All local energy requests and advertisements are submitted and handled at the edge, e.g., a router associated with the microcell. We assume the system limits the amount of requested energy by a consumer. We also assume that the IoT coordinator uses a reward system to encourage providers to share energy. Rewards come in the form of stored credits to providers. The provider may use the collected credits later to increase the limit on the amount of requested energy when they are in the consumer role. We assume that each provider may also be a consumer and vice-versa. A provider receives a reward based on the amount of shared energy. Assume a *device owner*, e.g., a *smart shoe*, wishes to share their spare energy with other IoT devices in the microcell. In this scenario, the provider is assumed to be able to receive requests from multiple energy consumers via the edge. We assume the provider can transfer energy to one consumer at a time. A service consumer may not commit to receiving all the requested energy. In this respect, the reliability of such consumers is paramount: Unreliable consumers may hinder the provider from utilizing their spare energy and thus reduce their reward. The unreliability of consumers might be voluntary, e.g., intentionally canceling the request, or involuntary, e.g., moving in and out of the microcell.

Fig. 4.2 represents an example of received energy requests with different reliability. The first and second energy requests ($ER1$ and $ER2$) require 55% and 75% of the provider energy, respectively. $ER1$ has a 90% reliability. $ER2$ has a 25% reliability.

¹<https://www.energous.com/>

Selecting $ER2$ will give a larger reward for the provider than $ER1$, but $ER2$ is more likely to be canceled. If the request is canceled, the provider's reward may be reduced. It is *challenging* to replace a canceled request as there is no guarantee to find another available request. Evaluating the reliability of the energy requests before selection may enhance the chances of obtaining the most committed ERs and, therefore, offer the best reward for the provider.

We focus on composing reliable energy service requests to maximize the reward of the energy provider. We assume a single energy provider may provide a *specific* time interval, during which they may share their energy with multiple consumers. The *composition* of energy service requests includes selecting the most reliable requests that maximize the provider's reward. The reliability of energy requests is calculated using a reliability model. Note that we assume a provider is static during the composition. Additionally, we assume consumers might move during the composition. To the best of our knowledge, no algorithm addresses the problem of energy requests' reliability in composing energy service requests to incentivize the provider.

4.3 System Model and Definitions

We propose a formal model of our reliable composition of energy requests. We consider the scenario of energy sharing in a microcell during a particular time interval T . We use the below definitions to formulate the problem.

Definition 1: Energy-as-a-Service (EaaS). We adopt the definition of EaaS in [25]. An EaaS is defined as a tuple of $\langle E_id, E_Pid, F, Q \rangle$, where:

- E_id is an energy service's unique identifier
- E_Pid is a unique provider's identifier
- F is the function of sharing energy by an IoT device
- Q is the non-functional , Quality of Service (QoS), attributes including:
 - P_ec is the capacity of energy the provider may share

- P_{loc} is the spatial location of the provider $\langle x, y \rangle$
- P_{st} is the start time of the provider's stay
- P_{et} is the end time of the provider's stay

Definition 2: Energy Service Request (ER). We extend the existing definition of the ER model presented in Chapter 3 by considering the reliability of the ER. An ER request is defined as a tuple of $\langle ER_{id}, ER_{Cid}, F, Q \rangle$, where:

- ER_{id} is an energy request's unique identifier
- E_{Cid} is a unique consumer's identifier
- F is the function of receiving energy by an IoT device
- Q is the non functional QoS attributes including:
 - C_{re} is the amount of requested energy by the consumer
 - C_{st} is the start time of the consumer's stay
 - C_{et} is the end time of the consumer's stay
 - C_{loc} is the spatial location of the consumer $\langle x, y \rangle$
 - C_{Rel} is the reliability score of the consumer

Definition 3: Incentive. The energy service provider gains an incentive reward R by sharing energy. Rewards are provided by the IoT coordinator at the edge. The rewards come in the form of stored credits to providers. We assume the system has a limit for the amount of requested energy. The collected credits may be used later by the provider to increase the limit of requested energy as a consumer.

In Chapter 3, we demonstrate that the quantity of provided energy influences the incentive's value to promote energy provision. As a result, we will compute the reward (rwd) of an energy request based on the amount of requested energy using the below equation.

$$rwd = C_{re}/100 \quad (4.1)$$

where C_{re} is the amount of requested energy, and 100 is the energy capacity of a full charge.

Definition 4: Reliability-based energy requests composition problem. We assume that, in a microcell, there exists an energy service $EaaS$ and a set of n energy requests $ERs = \{ER_1, ER_2, \dots, ER_n\}$ as shown in Fig. 4.2. The $EaaS$ is posted by a provider P . ER are initiated by consumers C . $EaaS$ and ER are described using Definitions 1 and 2, respectively. We use the service paradigm to reformulate the energy-sharing problem as a service composition problem. The composition of energy requests considers the spatio-temporal features of the service and requests. Composing energy requests for a provider's $EaaS$ requires the composition of energy requests $ER_i \in ER$ where $[C_st_i, C_et_i] \subset [P_st, P_et]$, $\sum C_re_i \leq P_ec$, the provider reward $= \sum rwd_j$ is the maximum they can get and the reliability of the composition $= \sum C_Rel_i$ is the highest in reliability.

We use the following assumptions to formulate the problem:

- IoT devices are equipped with wireless energy transmitters and receivers.
- The composition considers the case of a *single* provider and *multiple* consumers.
- The provider has a fixed energy size during the composition.
- The provider and consumers may have different time windows, but consumers' time windows must fall within the provider time window $T_c \in T_p$.
- The provider has a fixed location $\langle x, y \rangle$ for the duration of the energy service.
- The provider transfers energy within their range (15 feet using Energous) to one consumer at a time.
- There is no energy waste while sharing.
- All consumers have a history of energy requests
- Consumers' requests include information about the reward they are willing to pay.
- A trust framework has been implemented to preserve the integrity and privacy of the participating IoT devices [21].

4.4 Reliability Model

We define the reliability model to determine an energy request's reliability score (Rel). The reliability model considers the behavior of consumers and their history in requesting energy. We assume that the history of consumers will be collected in a phase prior to deploying our proposed framework. The below attributes are used to compute the reliability score.

- **Consumption History:** Consumer's history of energy requests may indicate their reliability. The number of successful requests can be used in computing the reliability. A successful request is one that receives all that was requested. Therefore, the computation of a consumer's reliability based on their history ($Rel_History$) is:

$$Rel_History = SER/Total_ER \quad (4.2)$$

where SER is the consumer's number of successful requests, and $Total_ER$ is the consumer's total number of submitted requests.

- **Energy Disruption:** As previously mentioned, disruption of energy transfer may occur voluntarily or involuntarily. We refer to the disconnection in energy transfer without reconnecting as the voluntary disruption (see Fig. 4.3). Similarly, we refer to the intermittent disconnection in energy transfer within one energy request as the involuntary disruption (see Fig. 4.4). We define the reliability computation of both types of energy disruption as follows:

- **Voluntary Reliability:** Voluntary disruption refers to a conscious decision by the consumer to discontinue the energy transfer. We identify voluntary disruption as having a disconnection in the energy request without reconnecting (see Fig. 4.3). A disconnection's duration has to be greater than a predefined threshold (δ) and less than a predefined

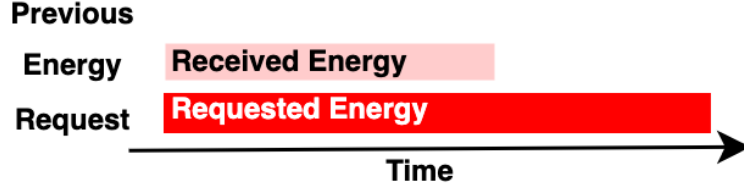


FIGURE 4.3. Example of voluntary disruption

threshold (ϵ) to be considered a voluntary disruption. Therefore, the voluntary disconnection duration can be identified as follows :

$$\delta \leq \text{Voluntary duration} \leq \epsilon$$

We compute the voluntary reliability if a voluntary disruption is identified in a previous energy request. We measure the voluntary reliability ($Rel_Voluntary$) based on *energy* and *time*. The reliability of energy (Rel_Energy) for a given consumer is:

$$Rel_Energy = \left(\sum_{i=1}^n \left(\frac{Received_Energy_i}{Requested_Energy_i} \right) \right) / n \quad (4.3)$$

where n is the number of the previous ER for a given consumer. The previous equation measures the average energy ratio of all the consumer's previous ER.

The reliability of time (Rel_Time) for a given consumer is:

$$Rel_Time = \left(\sum_{i=1}^n \left(\frac{Stay_Time_i}{Required_Time_i} \right) \right) / n \quad (4.4)$$

where n is the number of the previous ER for a given consumer. The previous equation measures the average time ratio of all the consumer's previous ER.

The equation of voluntary reliability ($Rel_Voluntary$) is:

$$Rel_Voluntary = (Rel_Energy + Rel_Time) / 2 \quad (4.5)$$

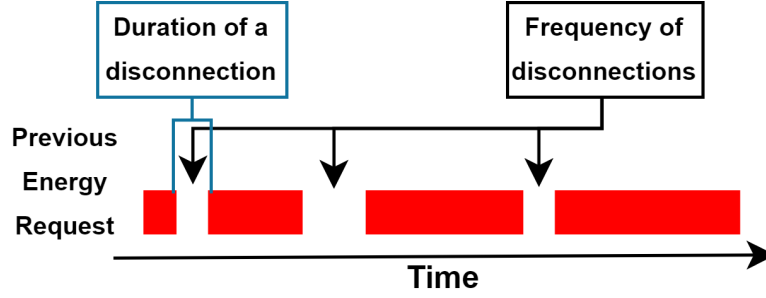


FIGURE 4.4. Example of involuntary disruption

- **Involuntary Reliability:** Involuntary disruption refers to an intermittent disconnection in energy transfer within a single ER (see Fig. 4.4). We identify involuntary disruption as having at least one disconnection between two connections of the same ER. A disconnection's duration has to be greater than a predefined threshold (β) and less than a predefined threshold (γ) to be considered an involuntary disruption. Moreover, we consider a single disconnection at the end of the energy transfer to be an involuntary disruption if its duration is less than (α). Therefore, the involuntary disconnection duration can be identified as follows:

$$\beta \leq \text{Involuntary duration} \leq \gamma$$

$$\text{or } (\text{Involuntary duration} \in \text{end of } T$$

$$\text{Involuntary duration} \leq \alpha)$$

We compute the involuntary reliability if an involuntary disruption is identified in a previous ER. The involuntary reliability for an ER ($Rel_Involuntary$) is computed based on the *frequency of disconnections* and the *duration of disconnections* in the request. ER is more reliable when the disconnections are less frequent and for smaller periods. For every previous ER ($Prev_ER$), we compute the reliability based on the frequency of disconnections (Rel_f) as the following:

$$Rel_{f_i} = \begin{cases} 1, & \text{if } f_i = 0 \\ 0.7, & \text{if } (f_i = 1 \wedge \beta < d_i < \gamma \wedge d_{i_et} \neq Prev_ER_{i_et}) \vee \\ & (f_i = 1 \wedge d_i < \alpha \wedge d_{i_et} = Prev_ER_{i_et}) \\ 1/f_i, & \text{if } (f_i \geq 1 \wedge \beta < d_i < \gamma) \end{cases} \quad (4.6)$$

where f_i is the frequency of disconnections within a single previous ER ($Prev_ER_i$), d_i is the duration of disconnections within a single ER, and $Prev_ER_{i_et}$ is the end time of the ER. We compute a consumer's reliability based on the frequency of disconnections as the average of the reliability of frequency (Rel_f) of all their previous requests ($Prev_ER$). Therefore, the equation to compute the reliability of a consumer based on the frequency of disconnections (Rel_freq) is:

$$Rel_freq = \left(\sum_{i=1}^n Rel_{f_i} \right) / n \quad (4.7)$$

where n is the number of the previous requests for a given consumer.

The duration of disconnections in ER may result in a loss in the provider's reward. For every previous ER, we compute the reliability based on the duration of disconnections (Rel_d) as:

$$Rel_{d_i} = 1 - \frac{\sum_{j=1}^k d_{i_j}}{Prev_ER_{i_et} - Prev_ER_{i_st}} \quad (4.8)$$

where k is the number of disconnections in a consumer's previous ER, d_{i_j} is the duration of disconnection within a single ER, $Prev_ER_{i_st}$ and $Prev_ER_{i_et}$ are the start and end times of the ER, respectively.

We define the reliability of the consumer, based on the duration of disconnections, as the average of the reliability of the duration of all their previous requests ($Prev_ER$). Therefore, the equation to compute the reliability of a consumer based on the duration of disconnections (Rel_dur)

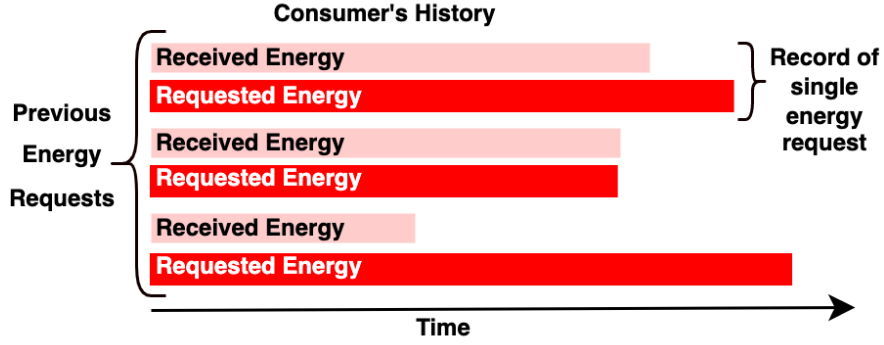


FIGURE 4.5. Example of random behavior

is the following:

$$Rel_dur = \left(\sum_{i=1}^n Rel_d_i \right) / n \quad (4.9)$$

where n is the number of the previous ER for a given consumer.

As previously stated, the involuntary reliability of a consumer will be computed based on the reliability of the frequency (Rel_freq) and the reliability of the duration of disconnections (Rel_dur). Therefore, we compute the involuntary reliability of a consumer ($Rel_{Involuntary}$) as the following:

$$Rel_Involuntary = (Rel_freq + Rel_dur) / 2 \quad (4.10)$$

- Random Behavior:** Random behavior measures the level of fluctuations regarding the consumer's commitment to receive the requested energy (see Fig. 4.5). We use the variation rate to estimate the random behavior of a consumer. Consumers with a high variation rate of commitment will be identified with random behavior. Highly random behavior is an indication of an unreliable consumer. We measure the variation rate of a consumer's behavior based on *energy* and *time*. The equation of the Energy variation rate (E_vr) for a given

consumer is:

$$E_{vr} = \sqrt{\frac{\sum_{i=1}^n \left(\frac{Received_Energy_i}{Requested_Energy_i} - \overline{\left(\frac{Received_Energy}{Requested_Energy} \right)} \right)^2}{n}} \quad (4.11)$$

where n is the number of the previous ER for a given consumer. The previous equation measures, for a given consumer, the variation between the energy ratio of their previous ER and the mean of all their previous ER.

The equation of the Time variation rate (T_{vr}) for a given consumer is:

$$T_{vr} = \sqrt{\frac{\sum_{i=1}^n \left(\frac{Stay_Time_i}{Required_Time_i} - \overline{\left(\frac{Stay_Time}{Required_Time} \right)} \right)^2}{n}} \quad (4.12)$$

where n is the number of previous ER for a given consumer. The previous equation measures, for a given consumer, the variation between the time ratio of their previous ER and the mean of all their previous ER.

As previously mentioned, the random behavior of a consumer is measured based on energy and time. Therefore, we compute the reliability of a consumer based on their random behavior (Rel_Random) as:

$$Rel_Random = (2 - (E_{vr} + T_{vr}))/2 \quad (4.13)$$

where E_{vr} is the energy variation rate computed by Eq. 4.11 and T_{vr} is the time variation rate computed by Eq. 4.12.

Consumer Reliability: The *reliability* score of a consumer is computed as the summation of all aforementioned reliability attributes. Therefore, the equation of computing the reliability score of a consumer (C_{Rel}) is:

$$C_{Rel} = \sum_{i \in n} w_i \times Rel_i \quad (4.14)$$

where $n = \{History, Voluntary, Involuntary, Random\}$ and w is the weight of each reliability attribute. We use $w = 0.25$ to normalize the impact of each factor on the overall reliability score. Since our dataset is synthesized with disconnections, as

explained in Section 4.6, it is unrealistic to use this dataset to compute the weight of each attribute.

Service Composition Reliability Score: The reliability score of a service composition will be the summation of the selected consumers' reliability scores. Therefore, the equation for computing the reliability score of a service composition is as follows:

$$\text{Service Composition Reliability} = \sum_{j=1}^m C_{Relj} \quad (4.15)$$

where m is the number of selected energy requests that would receive the energy from the provider, and C_{Relj} is the reliability score of the consumer of a selected energy request.

Actual Reward: The actual reward of an ER refers to the reward of a request considering its consumer's reliability score. For example, if an ER's consumer reliability score is 80%, then this indicates that the consumer will probably commit to 80% of the requested energy. Therefore, we compute the actual reward for an ER (Act_Rwd) with respect to its consumer's reliability as the following:

$$Act_Rwd = rwd * C_{Rel} \quad (4.16)$$

where rwd is the reward of the ER computed using Eq. 4.1 and C_{Rel} is the reliability score of the consumer of an ER computed using Eq. 4.14.

4.5 Composition Framework for Energy Requests

The framework of composing energy service requests will involve two phases: Selecting energy service requests and composing energy service requests (see Fig. 4.6). The first phase includes the selection of the energy requests (ERs) that can be composed,

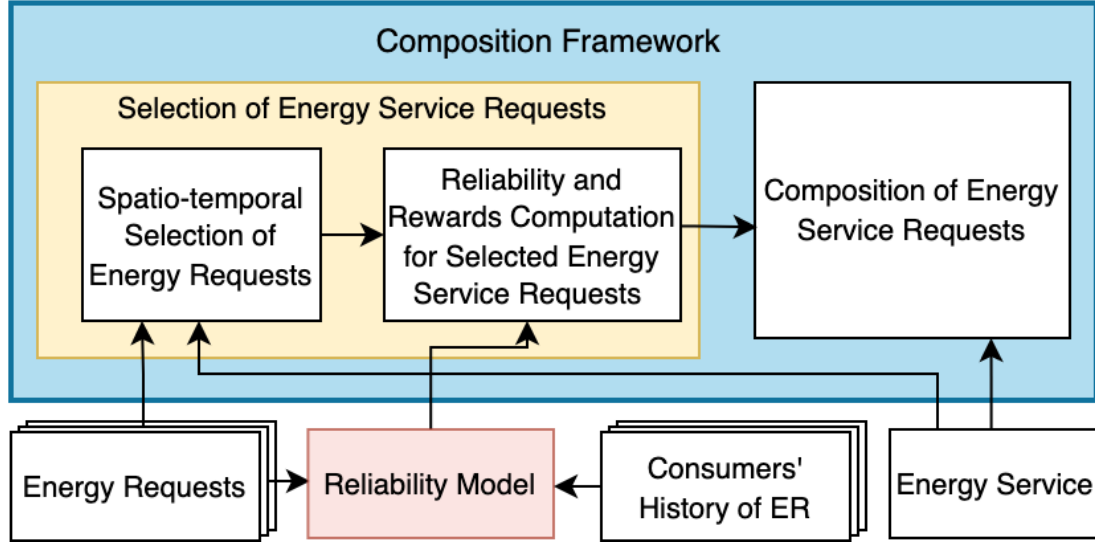


FIGURE 4.6. The composition framework for ER services

within space and time, with the energy service (EaaS). Then, we calculate the reliability score for each of the selected ERs using the reliability model. Additionally, the reward of each ER is computed in the same phase using Eq. 4.16. The second phase is the composition of the most reliable ERs that maximize the provider's reward. Our composition is designed for the case of a *single* provider and *multiple* consumers. The composition is inspired by Priority-based scheduling algorithms [95].

4.5.1 Selection of Energy Service Requests

The selection phase selects energy requests (ERs) that are composable with the provider's energy service (EaaS). The selection phase has two stages: The first stage involves the spatio-temporal selection of energy requests (see Fig. 4.6). The second stage includes the reliability score calculation and the rewards computation for the selected ERs. The first stage selects neighbor ERs that the provider may serve. The selection is achieved using the spatial and temporal features of EaaS. The spatial classification is required to transfer the wireless energy between consumers and providers. The temporal selection of an ER requires its duration to fall within the time window of the energy provider. In the first stage, the final step is to check if the EaaS may provide the requested energy. The second stage computes the reliability score for each selected

Algorithm 4.1 Reliability-based Composition of ER Services**Input:** $EaaS$, All_{ER} , Reliability Model (RM)**Output:** ER_{comp} , $compRel$, $compRwd$ **Phase 1: Selection of Energy Service Requests**

```

1: for  $ER_i$  in  $All_{ER}$  do
2:   if  $[C_{st}, C_{et}] \subseteq [P_{st}, P_{et}]$  then
3:      $d = \text{Compute\_Distance}(C_{loc.x}, C_{loc.y}, P_{loc.x}, P_{loc.y})$ 
4:     if  $d \leq \text{Max Energy Distance}$  then
5:       if  $C_{i.re} \leq P_{ec}$  then
6:          $C_{i.Rel} = \text{Compute\_Reliability}(ER_i, RM)$ 
7:          $ER_{i.Rwd} = C_{i.re} / P_{ec}$ 
8:          $ER_{i.Act\_Rwd} = C_{i.Rel} * ER_{i.Rwd}$ 
9:          $SelectedERs.insert(ER_i, C_{i.re}, ER_{i.Act\_Rwd})$ 

```

Phase 2: Composition of Energy Service Requests

```

10:  $SelectedERs_{sorted} = \text{sort}(SelectedERs,$ 
     $C_{st} : \text{ascending}, ER_{i.Act\_Rwd} : \text{descending})$ 
11:  $compRel = 0; compRwd = 0;$ 
12:  $Provider_{ST} = P_{st}; Provider_{EC} = P_{ec};$ 
13: for  $ER_i$  in  $SelectedERs_{sorted}$  do
14:   if  $C_{i.st} \geq Provider_{st}$  then
15:      $compRel += C_{i.Rel}$ 
16:      $compRwd += ER_{i.Act\_Rwd}$ 
17:      $Provider_{st} = C_{i.et}$ 
18:      $Provider_{ec} -= C_{i.re}$ 
19:      $ER_{comp}.insert(ER_i)$ 
20: return  $ER_{comp}, compRel, compRwd$ 

```

ER. We compute the reliability score using the model discussed in Section 4.4. We also use Eq. 4.16 to compute the reward in the second stage based on the amount of requested energy.

The selection phase of the energy requests is detailed in Phase 1 in Algorithm 4.1 (Lines 1-9). For every ER, the algorithm checks if the ER's time interval falls in the energy service's time window (Lines 1-2). Then, the distance d between the ER and the energy service is computed in Line 3. Line 4 checks if the computed distance d is less than the maximum required distance to transfer energy (15 feet using Energous). Line 5 tests if the available energy service is enough to provide the requested energy, while Line 6 computes the reliability score of the consumer using the previously defined reliability model (Section 4.4). Line 7 computes the reward value of the ER using

Eq. 4.1. Line 8 computes the actual reward using Eq. 4.16. Line 9 inserts the ER information, the consumer reliability score, and the ER's actual reward value in the set of selected ERs.

4.5.2 Reliability-based Composition of Energy Requests

Phase 2 of Algorithm 4.1 represents the composition of the energy requests (Lines 10-20). This phase aims to compose the most reliable ERs that ensure the maximum reward for the provider. Line 10 sorts the selected ERs in ascending order based on their start time, then in descending order based on their actual reward. Lines 13-15 check, for each ER , if the available provided energy can deliver the requested energy, then add up the consumer's reliability score as part of the composition's reliability using Eq. 4.15. In addition, Line 16 adds up the ER's actual reward $ER_{i.Act_Rwd}$ as part of the provider's reward. Line 17 updates the start time of the provider so that ERs do not overlap, since the provider can only transfer energy to one consumer at a time. Line 18 updates the remaining energy of the provider by subtracting the amount of requested energy C_{re} from the provider's available energy P_{ec} . Line 19 inserts the selected ER into the set of composed ERs.

4.5.3 Adaptive Reliability-based Composition of Energy Requests

We propose an *adaptive* reliability-based composition as an enhancement to Algorithm 4.1. In the adaptive composition, ERs will be modified and divided based on their voluntary reliability. Recall from Subsection 4.4 that voluntary reliability measures users' commitment to fully receive their requested energy. Therefore, our approach will downsize the amount of ER based on the consumer's voluntary reliability score. Algorithm 4.2 describes the composition in detail and is implemented across two phases: a selection phase and a composition phase. Lines 1-12 describe the first phase of selecting ERs that are composable with the provider's energy service. The selection phase has two stages: the first stage is similar to Subsection 4.5.1 (Lines 1-5), while the

Algorithm 4.2 Adaptive Reliability-based Composition**Input:** $EaaS, All_{ER}, Reliability\ Model(RM)$ **Output:** $ER_{composition}, compReliability, compReward$ **Phase 1: Selection of Energy Service Requests**

```

1: for  $ER_i$  in  $All_{ER}$  do
2:   if  $[C_{st}, C_{et}] \subseteq [P_{st}, P_{et}]$  then  $d =$ 
      $Compute\_Distance(C_{loc.x}, C_{loc.y}, P_{loc.x}, P_{loc.y})$ 
3:   if  $d \leq Max\ Energy\ Distance$  then
4:     if  $C_{i.re} \leq P_{ec}$  then  $C_{i.Rel.Voluntary} = Vol\_Rel(ER_i, RM)$   $C_{i.re} =$ 
        $C_{i.re} * C_{i.Rel.Voluntary}$   $C_{et} = C_{st} + (C_{et} - C_{st}) * C_{i.Rel.Voluntary}$   $C_{i.Rel} =$ 
        $Compute\_Reliability(ER_i, RM)$   $ER_{i.Reward} = C_{i.re} / P_{ec}$   $ER_{i.Act\_Reward} =$ 
        $C_{i.Rel} * ER_{i.Rwd}$   $SelectedERs.insert(ER_i, C_{i.re}, ER_{i.Act\_Rwd})$ 

```

Phase 2: Composition of Energy Service Requests**Similar to Phase 2 of algorithm 1 return** $ER_{comp}, compReli, compRwd$

second stage includes computing the reliability score, downsizing ERs, and computing the rewards for the selected ERs (Lines 6-11). Line 5 tests if the available energy service is enough to provide the requested energy. Line 6 computes the voluntary reliability score of the ER using Eq. 4.5. Line 7 downsizes the amount of requested energy based on the voluntary reliability score. Similarly, Line 8 adjusts the ER's end time based on the voluntary reliability score. Line 9 computes the reliability score of the consumer using the previously defined reliability model (see Subsection 4.4). Line 10 computes the reward value of the ER using Eq. 4.1. Line 11 computes the actual reward using Eq. 4.16. Line 12 inserts the ER's information, the consumer reliability score, and the ER's actual reward value in the selected ERs. The composition phase follows the same description as Phase 2 in Subsection 4.5.2.

4.6 Experiments and Discussions

We compared the proposed composition approaches, the Reliability-Based composition (RB) and the Adaptive Reliability-Based Composition (ARB), with two different approaches, a greedy approach (Greedy) and a Brute Force approach (BF). The greedy approach is inspired by the first come first served algorithm [95], where the ERs will

be sorted and composed based on their start time without considering their reliability. We considered the greedy approach as the baseline. In the Brute Force approach, we retrieved all the possible compositions of the energy service requests. Then, we selected the composition that has the maximum total reward.

We evaluated the effectiveness and the computation cost of the composition approaches. The effectiveness of each composition approach was measured in terms of its reliability score, reward value, and energy utilization. The experiments only tested the performance of the approaches from a *single* provider perspective.

4.6.1 Dataset Description

To the best of our knowledge, there is no existing dataset for wireless energy sharing among IoT devices. Therefore, we built an energy services environment close to reality, based on a dataset released by IBM for a coffee shop chain with three branches in New York City². The dataset consisted of one-month transaction records (49,895 records) of customers' purchases in each coffee shop. Each coffee shop consisted of, on average, 560 transnational records per day and a total record of 16,500 transactions. We considered the coffee shop customers as IoT users in the prospected environment. We assumed energy services were provided from wearables or the spare energy of IoT users' batteries. Similarly, energy requests were sent by IoT users. Additionally, we assumed that energy services and requests were sent to an IoT coordinator at an edge server—for example, a WiFi router in a coffee shop. In our experiments, we used the consumer ID, transaction date, time, location, and coffee shop ID from each record in the dataset to define the spatio-temporal features of the energy services and requests—for example, the start and end times of an energy service or request. We augmented the dataset by randomly generating the rest of the QoS parameters based on a uniform distribution. For instance, the transactions' time and location were used as the energy requests' starting time and location. The ending time of each energy request was generated randomly. We used a random uniform distribution to generate

²<https://ibm.co/2O7IvxJ>

TABLE 4.1. Experiments variables

Variables	Value
Total Energy Requests for coffee shop 1 in April	16,830
Energy Services	8,000
Duration of Services	30 - 120 minutes
Duration of Energy Requests	5 - 30 minutes
Provided Energy	50 - 100 %
Requested Energy	1 - 100 %
Battery Level	1 - 60%

the amount of provided energy for each service and the amount of requested energy for each request. Moreover, we generated energy disruptions for each request. We ran a total of 8,000 energy service advertisements. We used our proposed approaches for each energy service to compose energy requests. Table 4.1 summarizes the experimental variables.

4.6.2 Evaluation of the Composition Framework

We ran four experiments to determine the efficiency of our proposed approaches. The experiments evaluated the composition approaches in terms of their reliability score, reward value, energy utilization, and computation cost. We ran the composition algorithm and the other techniques in different settings by changing the time interval length of services. We gradually increased the time duration of energy services from 30 minutes to 120 minutes. We repeated the experiment 2,000 times at each point where the duration of energy services increased. We considered the average value for each technique at each point.

The first experiment compared the average reliability score of the proposed composition algorithms RB and ARB against BF and Greedy. As previously stated, the reliability score presents the consumers' commitment to receiving the requested energy. Therefore, a high-reliability score of a composition ensures a better reward for the provider. The composition reliability score was computed using Eq. 4.15. Fig. 4.7 presents the average reliability score for each algorithm. The x-axis in Figs. 4.7-4.9

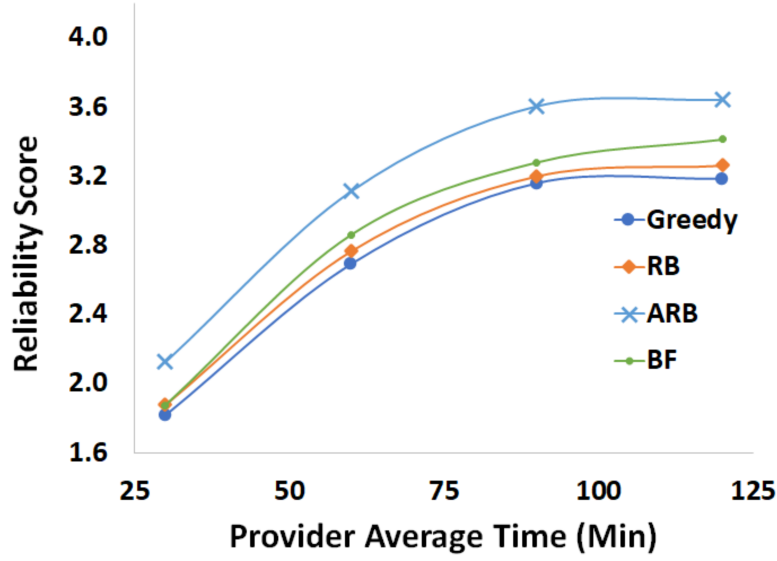


FIGURE 4.7. The average of the reliability score

represents the provider's average staying time. In Fig. 4.7, the reliability score increased when the provider staying time increased for all the composition algorithms. For instance, when the provider staying time is 30 minutes, all algorithms provided a lower reliability score compared to their scores when the provider staying time was 120 minutes. This observation can be explained by the provider staying time to share their energy. The longer the staying time of the provider, the more requests they could fulfill. Therefore, the aggregated reliability score of these requests increased. The proposed algorithm RB performed slightly better than Greedy in terms of reliability as, unlike the Greedy approach, it considers the reliability of each ER. In addition, the BF approach gave slightly better results than RB as it looks for all possible combinations of compositions and selects the composition with the highest reliability score. However, the BF approach had a higher computation cost than RB, as shown in Fig. 4.10. The proposed algorithm ARB gave the best results by downsizing the amount of requested energy in each energy request based on its reliability score. Downsizing the requests allows the ARB algorithm to include more requests and, aggregate additional reliability scores.

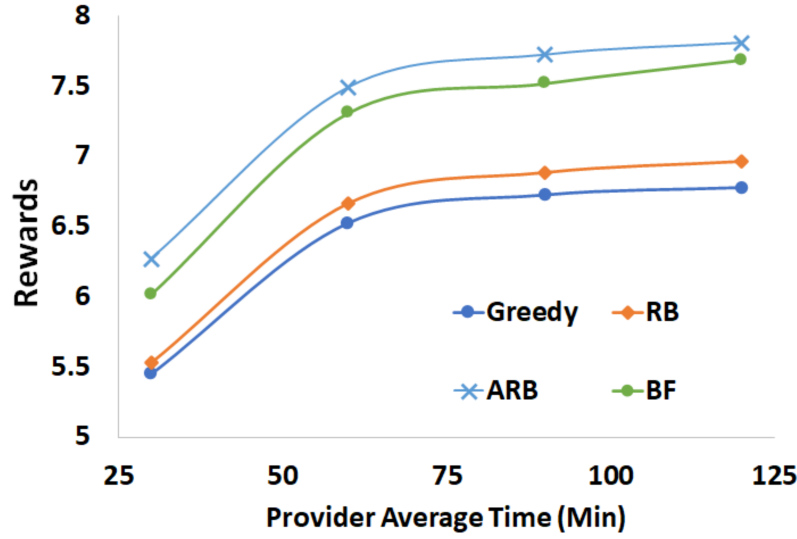


FIGURE 4.8. The average of total rewards

The second experiment compared the average of the total reward of each approach. Recall that rewards incentivize the provider to share their energy. Therefore, a high reward encourages providers to participate more in energy sharing. Similar to the previous experiment, in Fig. 4.8, the reward value increased when the provider's staying time increased for all the composition algorithms. For instance, when the provider staying time was 120 minutes, all algorithms provided more reward value compared to when the provider staying time was 30 minutes. Similarly, this observation can be justified by the provider's staying time to share their energy. The longer the provider stayed, the more requests they could fulfill, and therefore, the reward value increased. By observing Fig. 4.8, it is clear that the proposed approach RB resulted in higher rewards than the Greedy approach. The BF approach gave better results compared to RB and Greedy. However, as aforementioned, BF required more computations. The proposed algorithm ARB gave the best results among all approaches. As mentioned, the ARB algorithm downsizes the amount of requested energy in each energy request based on their reliability score. Downsizing the requests allowed the ARB algorithm to include more requests and aggregate additional rewards.

The third experiment compared the average of the providers' remaining energy after each composition (see Fig. 4.9). The low remaining energy of a provider's energy

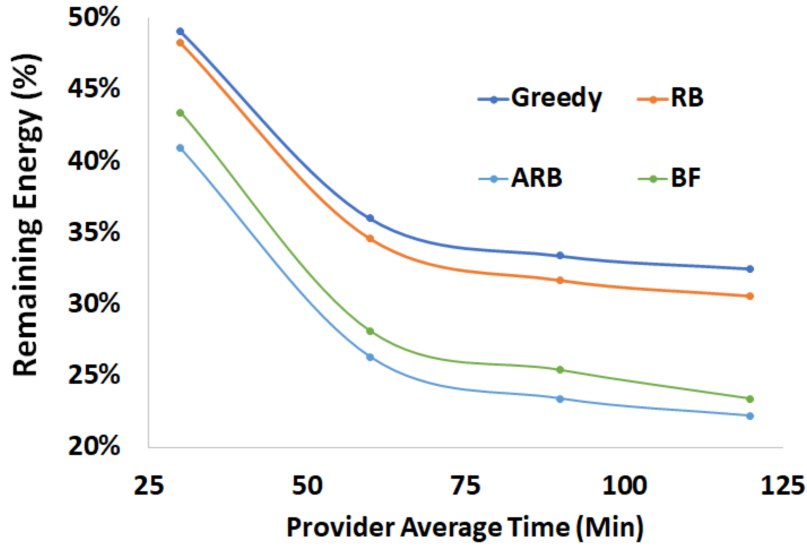


FIGURE 4.9. The average of providers' remaining energy

service indicates good energy utilization, which promotes a green IoT environment. Therefore, the experiment aimed to compare the energy utilization of each approach by comparing their remaining energy. In Fig. 4.9, the remaining energy decreased when the provider staying time increased for all the composition algorithms. For instance, when the provider stayed 120 minutes, all algorithms provided a low remaining energy percentage compared to their scores when the provider's staying time was 30 minutes. This observation can be explained by the provider's staying time to share their energy. The longer the provider stayed, the more requests they could fulfill. Thus, the provider's remaining energy will be less. The figure also shows that the proposed approach RB performed better than the Greedy approach in utilizing the provider's energy service. As aforementioned, the RB approach composes ERs while considering the reliability score of each ER. Considering the reliability enable the RB approach to ensure the commitment of consumers to receive energy, which results in better energy utilization. Comparing BF to RB and Greedy, the BF method provided better results. However, BF needed further computations, as previously mentioned. The proposed algorithm ARB gave the best results among all approaches. As mentioned, the ARB algorithm downsizes each energy request based on its reliability score. Downsizing the

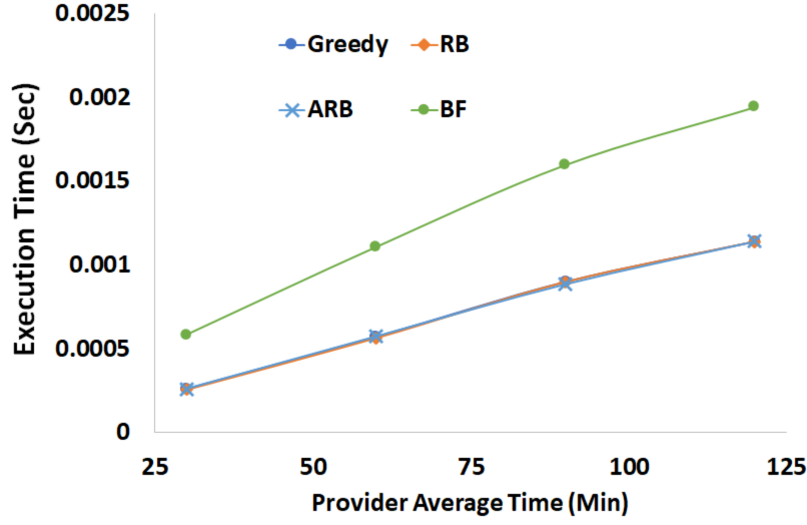


FIGURE 4.10. The average execution time

requests allowed the ARB algorithm to include more requests, increasing providers' energy utilization.

The fourth experiment evaluated the computation cost of all the composition approaches. Fig. 4.10 shows the execution time of each approach. The figure shows that the BF approach has the highest computation cost because the BF approach looks for all the possible combinations of compositions and selects the composition with the highest reliability score. The ARB approach has a lower computation cost than the BF approach. Additionally, The ARB approach offers the best results in terms of the reliability score, total reward, and energy utilization, as previously discussed and presented in Figs. 4.7-4.9.

4.7 Conclusion

We proposed an energy requests composition framework that evaluates consumers' reliability while incentivizing providers to share energy. A new reliability model was proposed to capture the users' consumption patterns. A reliability-based and incentive-driven composition of energy service requests was proposed. The approach selects

the most reliable energy service requests that maximize the provider's reward to overcome the provision's resistance. We additionally proposed an adaptive approach where energy requests will be downsized based on the consumers' reliability score. Experimental results show that the adaptive composition (ARB) outperforms all the evaluated approaches. The efficiency of the proposed approaches was tested against Brute Force (BF) and Greedy approaches.

CHAPTER 5

Energy Service Demonstration

In this chapter, we presented several demonstrations and frameworks. We also conducted preliminary experiments to provide : (1) a proof of concept for deploying an energy services ecosystem for Internet of Things (IoT) devices, where users share spare or harvested energy, and (2) comprehensive real-world datasets to evaluate the effectiveness of the proposed models and composition frameworks. These demonstrations and experiments can be grouped into two main components: a wireless energy transfer setup and an energy harvesting estimation model.

The platform collects real energy transfer datasets and demonstrates the transfer of wireless energy between smartphones to share energy. We also propose an energy estimation framework to predict the actual received energy. Preliminary experiments are conducted to evaluate the effectiveness of our framework.

The energy harvesting estimation model may be used to predict the amount of energy a provider may harvest. The predicted amount may be used in forecasting the provider service availability. To design an energy harvesting model, we designed an activity profiling framework to estimate harvested kinetic energy. The framework is the first step to building the energy harvesting model. However, our proposed activity profile framework requires users to consistently wear their activity trackers (i.e., wearables) and report their daily activities. Attaining consistent use and adoption of wearables is a challenge [97]. Therefore, we also study the impact of nudging strategies on individuals' adherence to wearing activity trackers.

5.1 Peer-to-peer Sharing of Wireless Energy Services

This section demonstrates a peer-to-peer wireless energy setup. The demo involves using coils and a mobile app to exchange energy services. Additionally, the app monitors and collects data about wireless energy transfer. We also propose an energy estimation framework to predict the actual received energy. Determining the actual received energy will inform existing energy service composition frameworks to allocate services efficiently. In the following two subsections, we present the details of the wireless energy service demo and the energy estimation framework.

5.1.1 Wireless Energy Services Demonstration

Recent studies have made substantial progress in addressing several challenges in crowdsourcing energy services, including service composition and fairness [24][13]. However, existing works rely on simulation analysis of the proposed techniques using synthetic datasets. In other words, there is no demonstration of the existing literature on *real devices using wireless power transfer over a distance*. Hence, we demonstrate peer-to-peer sharing of wireless energy services among smartphones over a distance.

Our demo leverages a service-based technique to share energy services, the state-of-the-art power transfer technology over a distance, and an extension of the proposed app by [5]. Similar to the previously mentioned app, our app enables consumers and providers to share energy conveniently and to monitor the sharing process for further analysis. In addition, unlike the previous app, our demo allows users to (1) share energy by requesting an amount of energy or requesting to be charged for a specific period and (2) transfer energy wirelessly over a distance. Moreover, in this demo, we conduct preliminary experiments to verify the feasibility and stability of the platform with wireless charging over a distance. We also compare and analyze the performance of the wireless charging technology with reverse and cable charging technologies.

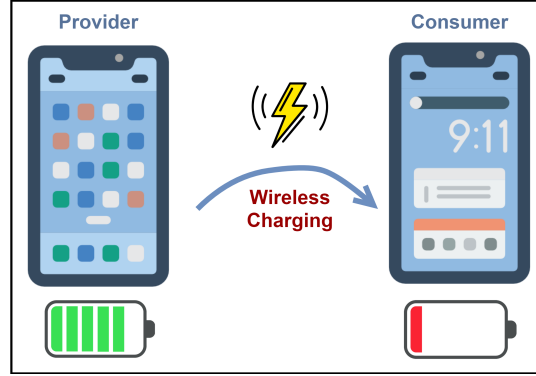


FIGURE 5.1. Wireless energy charging scenario

5.1.1.1 System Overview

We consider a scenario where people congregate in confined areas (e.g., restaurants, coffee shops, and movie theaters). In the confined areas, we assume that IoT users will share their spare energy or request energy with nearby devices (see Fig. 5.1). IoT users will use our mobile application to request energy as *energy consumers* or to accept energy requests as *energy providers*. In our app, IoT users send and receive requests directly using Bluetooth. Moreover, our mobile application extends the app proposed by [5] because of its limitations in synchronization, variety of request types, and using built-in reverse charging. In contrast, our application expends the previous application with the following features: (1) our app allows users to send and receive energy based on size. For instance, a consumer may ask a provider to charge 1,000 mAh. (2) our app also offers the option of requesting energy based on a period of time. For instance, a consumer may ask a provider to charge them for the next 10 minutes. Finally, (3) our application records the battery status of providers and consumers every time interval t , e.g., every 10 seconds, with the actual time of each record. This enables us to synchronize the monitoring process by recording the data at the exact time, e.g., second. The recorded data may be used for further analysis, e.g., studying the charging rate or optimizing the energy loss.

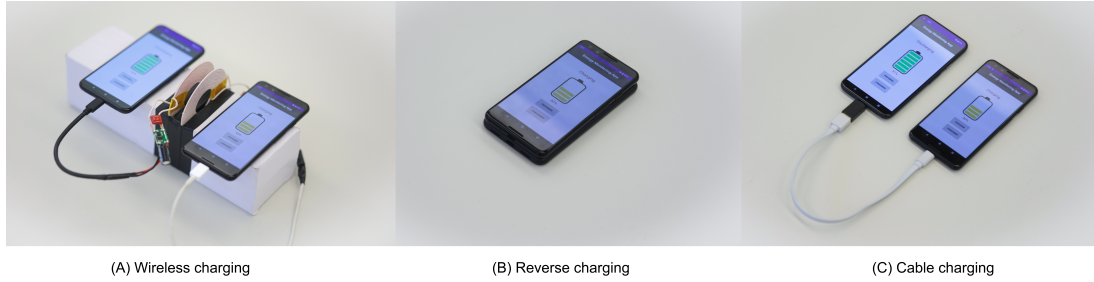


FIGURE 5.2. Used charging technologies

5.1.1.2 Demo Scenario

Our application runs in the following scenario. Once a consumer has submitted a request, our platform will automatically submit a request to the closest provider and await the provider's response. If the provider accepts, the energy sharing and monitoring process will start. Then the process will end once the consumer receives their request (e.g., an amount of energy or being charged for a specific period). Simultaneously, the platform synchronizes the monitoring process between the provider and consumer and captures the data of the entire energy transfer process. Once the sharing process ends, the app uploads the collected data to the edge service management system (e.g., a router associated with the confined area). The current version of our energy-sharing platform supports a one-to-one energy transfer mode and has demonstrated reliable performance in monitoring and demonstrating the wireless energy-sharing process over a distance.

5.1.1.3 Demo Setup

Our demo will exhibit our platform through a real case of wireless charging over a distance between two smartphones. In this case, we use near-field wireless charging (i.e., inductive coupling wireless technique). The inductive coupling device consists of two coils, and the electrical energy is delivered based on the magnetic field (see Fig. 5.2(A)). We use a Google Pixel 5 smartphone as the provider and a Google Pixel 3 as the consumer. We connect the consumer and provider to the wireless charging coils,

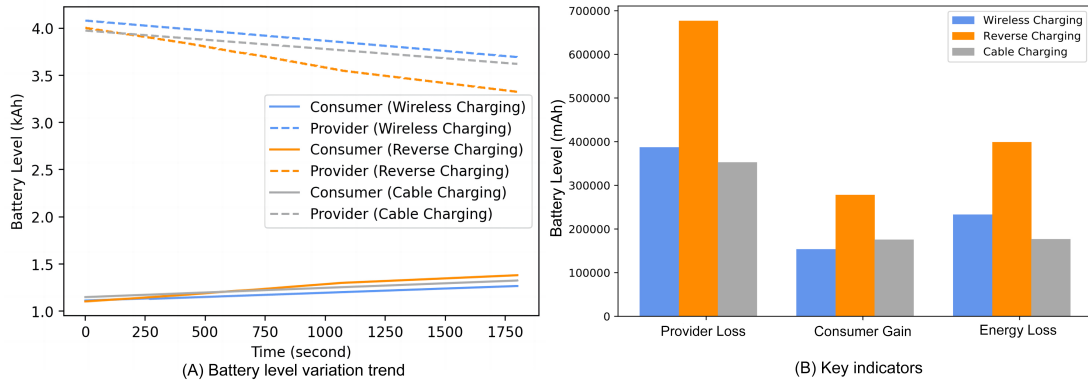


FIGURE 5.3. Comparison among different charging technologies

then run a case similar to the previously mentioned scenario in Section 5.1.1.2. A real-time video of the entire process of using the devices and app to request, receive, and monitor the energy-sharing process is published at this link: youtu.be/d-bdFGk6z4A

5.1.1.4 Preliminary Experiments

We conducted preliminary experiments to demonstrate our platform's feasibility and monitor different charging processes. We used the same setup mentioned in Section 5.1.1.3. We ran two experiments on the collected dataset from the charging process to analyze and demonstrate the behavior of each charging technology, i.e., wireless charging, reverse charging, and cable charging (see Fig. 5.2). The distance between the two coils was set to 2 centimeters for wireless charging settings. The first set of experiments aimed to compare differences between charging technologies regarding battery level variation trend, provider loss, consumer gain, and energy loss. The start battery of the consumer was 40%; the provider battery was 100%. We ran the experiment using a time-based request to receive energy. The requested time was 30 minutes. Additionally, we set the recording interval to 1 second. The experiment showed that all approaches behaved similarly (see Fig. 5.3). However, the provider with reverse charging had the highest energy loss. It is worth noting that energy loss involves the energy consumed to share energy and to run the device, e.g., run the operating system, system applications, etc.

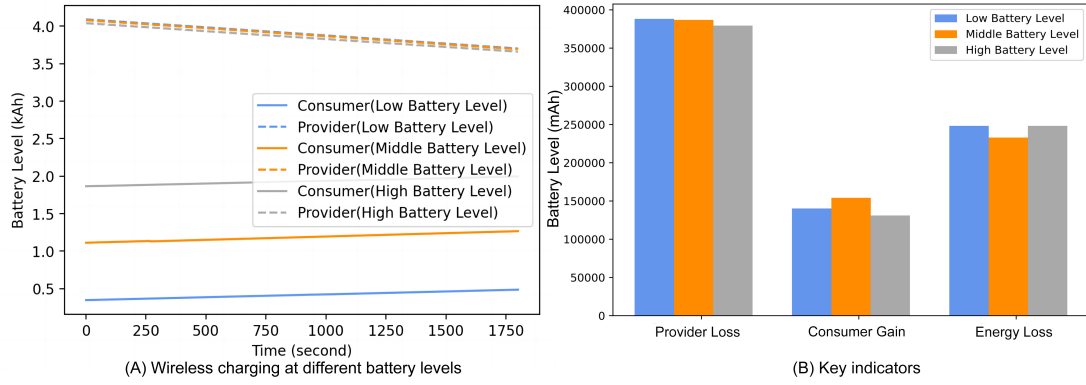


FIGURE 5.4. Comparison of wireless charging with different battery levels for consumer

The second set of experiments aimed to test wireless charging with different battery levels for the consumer at each experiment. The consumer's start battery level was tested on 10%, 40%, and 70% and labeled as low, middle, and high battery levels. The experiment showed that varying battery levels have less impact on the charging process (see Fig. 5.4).

5.1.2 Wireless Energy Estimation Prediction

Several energy service composition frameworks have been proposed to manage allocating services to requests [14][49]. These frameworks assume that the exact amount of advertised energy fulfills energy requests. However, in reality, consumers receive less energy than expected [6]. Several factors contribute to *energy loss* during wireless energy transfer, including the type of technology employed, the distance between devices, and the consumption behavior of IoT devices [48][98].

Estimating the *actual received energy* is crucial in assessing energy sharing efficiency. Capturing the wireless energy-sharing loss is essential for successfully deploying efficient energy service composition techniques. Energy loss may inform the service composition to optimize the allocation of energy services. For example, choosing a smaller energy service could be better than selecting a larger service due to distance.

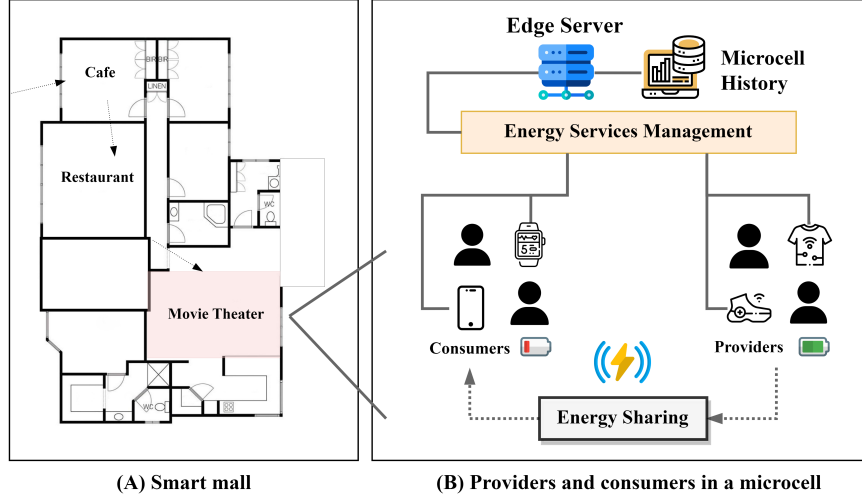


FIGURE 5.5. The energy services environment

Furthermore, considering actual energy in the composition ensures consumers' expectations are met [23].

We propose to monitor wireless charging efficiency by building a wireless-transfer energy estimation framework. Our framework is used to *estimate the actual amount of received energy* by a consumer. Our framework uses two machine learning models, i.e., XGBoost and Neural Network, to estimate the actual received energy. Both models consider two features in estimating the received energy: (1) the distance between the devices, and (2) the duration of the energy transfer. We consider these features because different distances and durations impact the wireless transfer of energy [6]. We employed a real-world wireless energy-sharing dataset to demonstrate the efficacy of our framework. To the best of our knowledge, existing research has yet to consider the actual received energy in the wireless energy delivery [13][14]. This work represents one of the first attempts to quantify actual received energy in the context of wireless energy-sharing services.

5.1.2.1 Motivating Scenario

We describe a scenario in a confined area (i.e., a microcell) where people typically gather (see Fig. 5.5(A)). Each microcell may have several IoT devices that act as

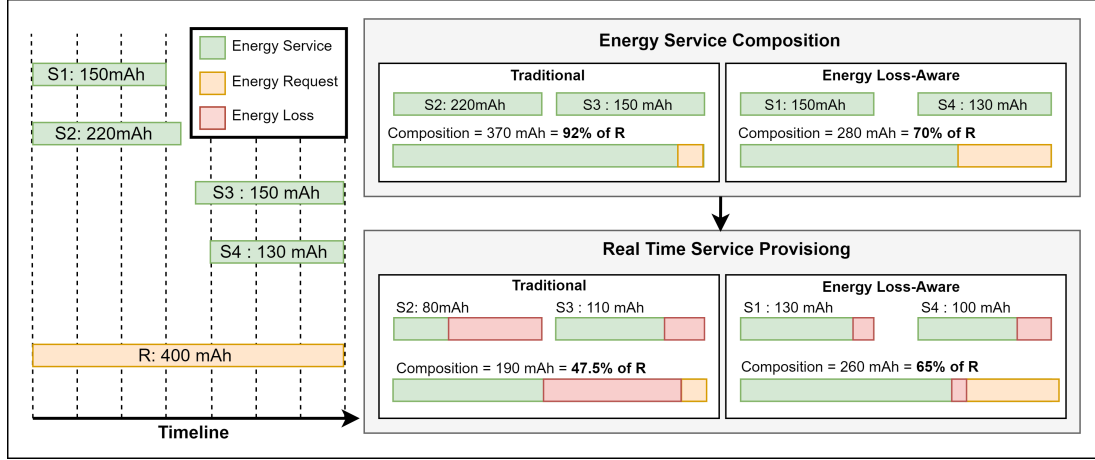


FIGURE 5.6. Example of the motivating scenario

energy consumers or providers (see Fig. 5.5(B)). Consumers and providers may submit their requests or service advertisements to a local *edge*, e.g., a router in a microcell. Assume that a *device owner*, e.g., a *smartphone*, requires energy to execute critical tasks, such as making a call. In this scenario, the consumer is assumed to be able to receive service advertisements from multiple energy providers through the edge. We assume that consumers can receive energy from one provider at a time. Several resource-allocation frameworks have been proposed to select the optimal set of energy services for a consumer [23][25]. However, none of the existing frameworks considers the energy loss during wireless energy delivery. In reality, consumers often receive less energy than expected [6]. In this respect, estimating the actual received energy while exchanging services is essential. Indeed, services with high energy loss do not deliver the expected energy. As a result, selecting services with high energy loss will discourage consumers from participating in this ecosystem [14].

Figure 5.6 shows an example of the received energy services with different energy amounts. A traditional energy service composition allocates services with a higher capacity. For instance, using traditional composition, the first ($S1$) and third ($S3$) services are selected with the expectation of fulfilling 92% of the required energy. However, in real-time charging, energy loss occurs based on several factors, such as

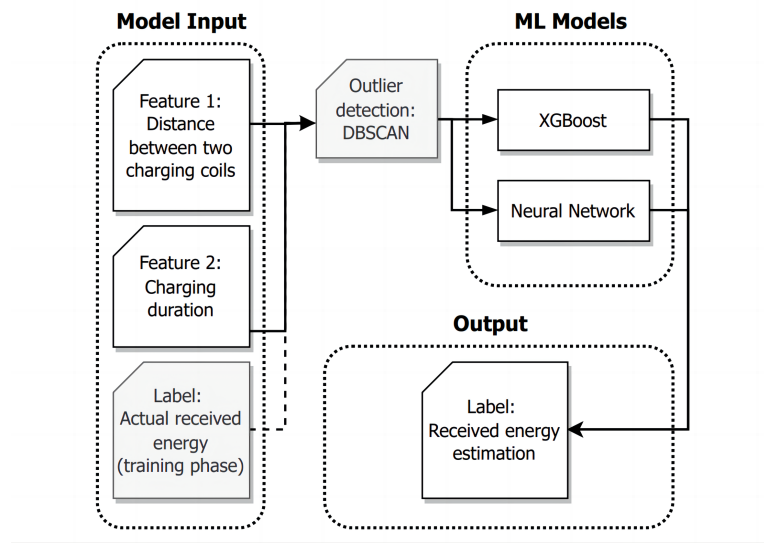


FIGURE 5.7. Wireless energy estimation framework

distance. The energy composition fulfills 47.5% of the required energy due to the energy loss of both services. Conversely, informing the composition framework with the predicted actual received energy will enhance the selection of services. For instance, given the actual received energy of shared services, the energy-loss-aware composition will guarantee to fulfill 65% of the required energy. In this case, the energy loss plays a key role in the development of composition algorithms [14]. Therefore, estimating the actual received energy before selecting and composing energy services ensures the efficient composition of the services.

5.1.2.2 Wireless Energy Estimation Framework

Our wireless energy estimation framework is presented in Fig. 5.7. Specifically, our framework consists of four steps:

- (1) **Wireless Energy Transfer Dataset Collection:** We collect a real wireless energy transfer dataset to train our machine-learning models. Data is collected using the app developed in Section 5.1.1. In our experiments, we used a Google Pixel 5 smartphone as the provider and a Google Pixel 3 as the consumer. We connected the consumer and provider to wireless charging coils

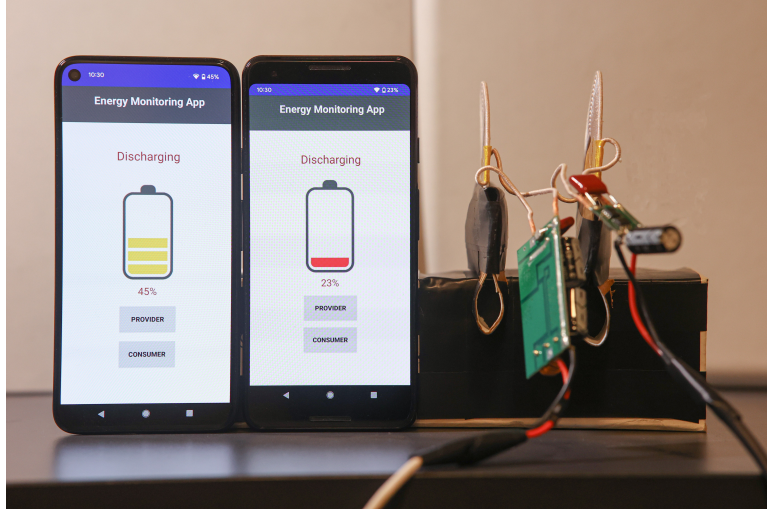


FIGURE 5.8. Hardware experiments setup for data collection

(see Fig. 5.8). Moreover, all experiments were conducted in the laboratory at temperatures around 20 degrees Celsius to obtain accurate results. Furthermore, we used the app to request energy by time. We set the charging period to 30 minutes and the monitoring interval mt to 1 minute. In order to conduct wireless charging at various distances between two coils, we created a sliding device (see Fig. 5.8). Users can use our device to share energy wirelessly over different distances. Our experiments collected datasets over different distances, i.e., 1 cm, 1.5 cm, and 2 cm. We then repeated the experiments over each distance five times. Therefore, we have 450 data points (i.e., $5 \times 30 \times 3$). To sum up, the collected dataset consists of five attributes: (1) distance between the coils, (2) charging duration, and (3) battery levels of the consumer and the provider every mt time interval in mAh and percentage. In our experiments, mt was 1 minute.

- (2) **Feature Selection:** In our experiments, we considered the actual energy received by the consumer. Therefore, we selected the following attributes to train our models: (1) the charging duration, (2) the consumer's battery level at each time, (3) and the distance between charging coils. The gathered dataset is visualized in Fig. 5.9(A). In training our models, we used the actual received energy as the label, while the other two variables were the selected

features. As a result, we transformed the objective of predicting the received energy as a regression machine learning problem.

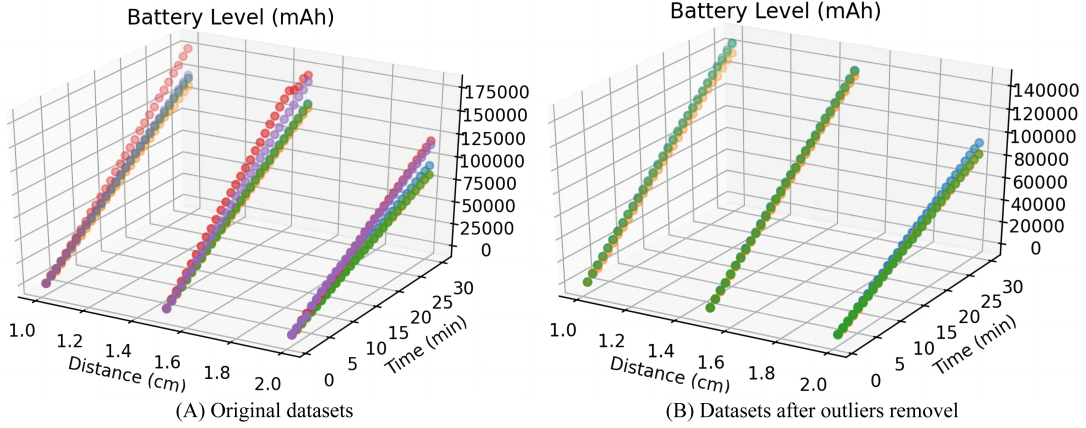


FIGURE 5.9. Training datasets visualization

- (3) **Outlier Detection:** The collected dataset had some abnormal charging battery levels that resulted in some of the data points being far away from the majority of the data points (see Fig. 5.9(A)). As we used the Root Mean Square Error (RMSE) as our evaluation metric and due to RMSE sensitivity to outliers [99], we use an outlier detection technique, namely, Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [100], in order to detect outliers. In contrast to the statistical method that can detect anomalous points that are above and below a certain threshold (extremes), DBSCAN is able to discover data that occur infrequently (i.e., detect outlier points with a lower density) [101]. Moreover, there is a low likelihood of errors in the wireless charging process, which may result in abnormal values in the consumer battery level. Consequently, those anomalous data points have a lower density than normal data points. Therefore, we used DBSCAN to detect the outlier battery values. As aforementioned, a charging error may cause an anomalous rise in battery level. Hence, focusing on the increase in the consumer battery level after the charging process reduces the problem to two dimensions: the distance between two charging coils and the increase in battery level. Fig. 5.9(B) depicts the training dataset excluding the outliers, resulting

in 270 data points. Furthermore, before feeding the data into our model, all of the target values, i.e., the battery level of the consumer, are divided by 1,000 to reduce computational complexity. Hence, our framework transforms the unit of battery level from milliampere-hours (mAh) to ampere-hours (Ah).

- (4) **Energy Estimation Models Training:** We used a conventional machine-learning method, Extreme Gradient Boost (XGBoost), as our baseline [102]. The XGBoost algorithm is a boosting-based method for integrated learning. Our XGBoost model selects the optimal hyperparameters using the Grid-Search method in scikit-learn [103]. Additionally, we used neural networks as our second model. Since the data distribution is not particularly complex, we constructed a simple multi-layer perception with three hidden layers as our neural network model. As we have two input features, i.e., the distance between charging coils and the charging duration, we set the number of neurons in the input layer to two. Furthermore, as this is a regression machine learning problem, the number of neurons in the output layer was set to one. The result of our neural network model is a numerical value representing the estimated increase in the consumer battery level.

5.1.2.3 Evaluation

Since our energy estimation is a regression problem, we calculate the absolute gap between the predicted and actual battery level increase [99]. We use a commonly used evaluation metric for regression problems, i.e., the Root Mean Square Error (RMSE) (see Eq. 5.1). The numerator represents the summation of squares of the difference between the prediction and ground truth of each data point, while the denominator denotes the number of input data.

$$RMSE = \sqrt{\frac{\sum (\hat{y}_i - y_i)^2}{n}} \quad (5.1)$$

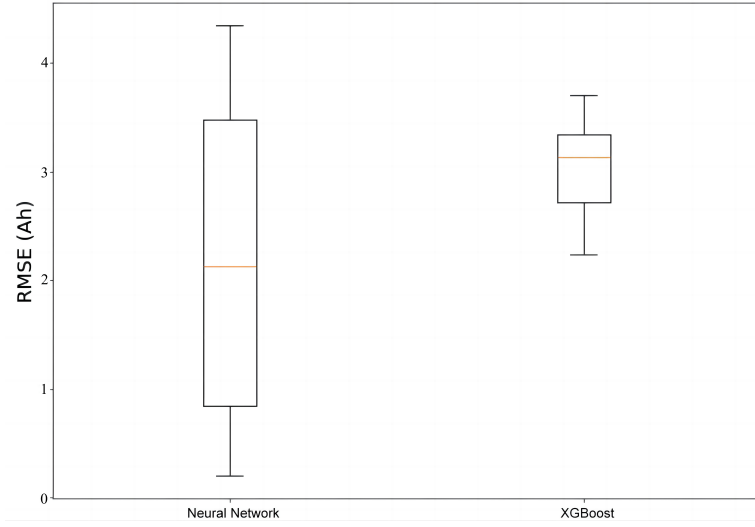


FIGURE 5.10. Box-plot of Root Mean Square Error (RMSE)

The RMSE of XGBoost and neural network models are computed by ten-fold cross-validation. A comparison of both models' performances is represented in Fig. 5.10. The neural network model effectively reduces the RMSE score compared to the XGBoost model.

5.2 Activity-based Profiling for Energy Harvesting Estimation

Enabling a sustainable energy services ecosystem will offer ubiquitous access to energy [13]. In this environment, IoT devices can be charged automatically without human intervention. To achieve that, the system must know *when*, *where*, and *how much* energy an IoT device may request/offer as a service from/to nearby IoT devices [14]. This section focuses on estimating the energy services a provider may offer, specifically on *estimating how much energy a provider may harvest to offer as a service*.

Studies have proposed to estimate the energy from daily activities by measuring the output from kinetic energy harvesters [104]. Harvested energy may be estimated by capturing the device wearer's kinetic movement and body temperature. Moreover, other studies used kinetic energy harvesting data to recognize daily activities [105][106].

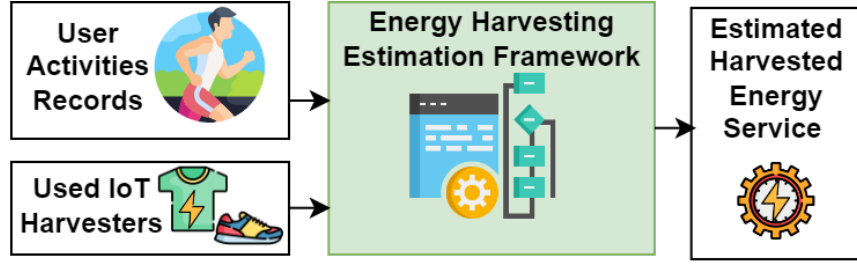


FIGURE 5.11. Energy Harvesting Estimation System

In principle, it is possible to estimate the amount of harvested energy based on the activity and the worn harvester's type [105]. Such activity may be collected by wearables such as smartwatches. In this thesis, we envision a novel framework to estimate IoT users' harvested energy, consisting of an activity-based profiling component and an energy-harvesting estimation model (see Fig. 5.11) [14]. The activity-based profiling component will classify users based on their daily activities to associate them with the corresponding energy harvesting estimation model. The model will use the activities profiles and the type of worn harvester to estimate the harvested energy. As energy harvesting technologies are still developing, *we focus on activity-based profiling*.

Several studies have used fitness data to model its wearer behavior for fitness purposes [107]. Hence, we propose to *use fitness data to recognize users' activities for energy harvesting purposes*. Even though people may do the same activity, e.g., walking, they may differ in activity intensity level, which may harvest different amounts of energy [105]. Therefore, *we propose an activity-based profiling framework to identify macro profiles that represent the major activity behavior of IoT users*. These profiles will be used to build an energy harvesting model to estimate the harvested energy based on the users' activity and the used energy harvester.

The proposed activity profile framework requires users to consistently wear their activity trackers (i.e., wearables) and report their daily activities. However, attaining consistent use and adoption of wearables is a challenge [97]. Indeed, people often tend to forget or stop wearing their devices over time [108]. The inconsistency in using the wearables is a common challenge that hinders the complete understanding of users'

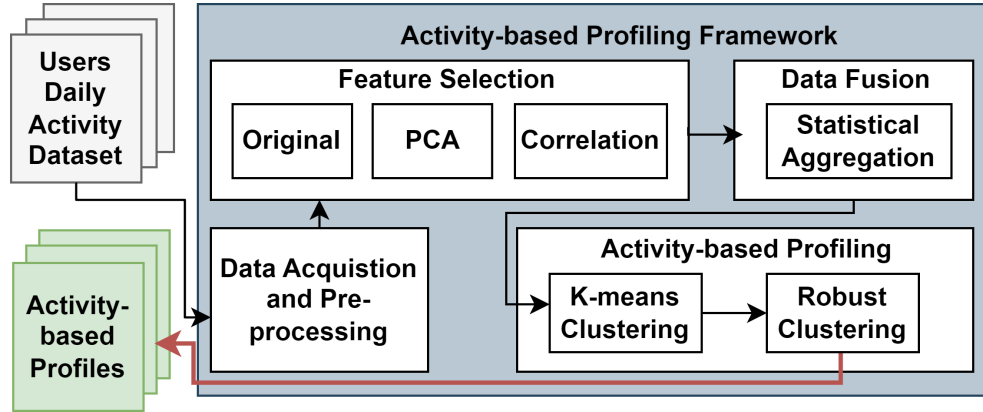


FIGURE 5.12. Activity-Based Profiling System Architecture

behavior [109]. A crucial question is how to change behaviors to increase and sustain regular activity tracking. *Nudging* is a key mechanism to increase and motivate regular human engagement [110]. Therefore, in this section, we also study the impact of nudging strategies on individuals' adherence to wearing activity trackers.

In the following subsections, we discuss the details of our activity-based profiling framework and our study on the impact of nudging on activity tracking.

5.2.1 Activity-based Profiling Framework

The activity-based profiling framework clusters users according to their daily activity records. The framework consists of four phases: Data Acquisition and Pre-processing, Feature Selection, Data Fusion, and Activity-based Profiling (see Fig. 5.12). In what follows, we discuss each phase:

- (1) **Data Acquisition and Pre-processing:** Users' health-related information and daily activity data are obtained during this phase from their fitness trackers. We used a public Fitbit dataset of 940 records of 33 users' daily activities over two months in 2016¹. The dataset does not have bio-related data for all users, so only activity-related data is used. Also, three features are removed

¹[kaggle.com/arashnic/fitbit](https://www.kaggle.com/arashnic/fitbit)

from the dataset: ‘ActivityDate’, ‘TrackerDistance’, and ‘LoggedActivity-Distance’ because they are exact duplicates of other attributes.

- (2) **Feature Selection:** Feature selection identifies the most representative features of an IoT user’s daily activity. The following three approaches were applied separately to the original cleaned dataset, resulting in three processed datasets:
 - (a) **Original module:** This module contains all the features of the pre-processed dataset.
 - (b) **PCA module:** We apply the Principal Component Analysis (PCA) technique on the original dataset to transform all the attributes into only three components.
 - (c) **Correlation module:** This module uses a feature-based heat map to select features based on their correlation, removing highly positively or negatively correlated features.
- (3) **Data Fusion** Clustering users into groups does not apply to our fitness dataset as it has multiple activity records for each user. Therefore, data fusion is needed to aggregate the records of each user with minimal information loss. Specifically, we used the statistical values (maximum, minimum, range, standard deviation, mean, and median) to capture user characteristics. These statistics are computed for each user in the three processed datasets from step (2).
- (4) **Activity-based Profiling** This phase includes applying *K-means* clustering to users’ aggregated records, followed by using the resulting clusters in a robust clustering approach.
 - (a) **K-means Clustering** K-means clustering is applied to each dataset obtained earlier. We used the *elbow* method to determine the optimal number of clusters for each dataset. As a result, each dataset produced four clusters.
 - (b) **Robust Clustering** As aforementioned, the clustering outcome was four clusters for each dataset, but with varying cluster sizes (number of users).

TABLE 5.1. Evaluation Metrics of clustering algorithms

Metrics \ Feature Selection	K-means Clustering			Robust Clustering		
	Original	PCA	Correlation	Original	PCA	Correlation
Silhouette Score	0.47574	0.51564	0.48323	0.73409	0.68170	0.72274
Davies Bouldin Index	0.55996	0.49664	0.54936	0.38857	0.42842	0.40737
Calinski Habrasz Index	72.97150	75.68820	75.72814	81.62642	78.73628	79.16408

To address this, we define *robust clustering* by identifying common users across the same clusters in all three sets. Users clustered together in all three K-means sets are defined as one robust cluster. Users with different clustering neighbors in the original three K-means sets are defined as separate robust clusters. This produces seven robust clusters/profiles, each representing an activity-based macro profile for a group of IoT users with common behavior, i.e., activity-based behavior. A harvested-energy estimation machine-learning model may be used to associate each profile with an estimated energy amount based on its characteristics, using a user's daily records, profile, and type of harvester worn. The energy harvesting estimation model will be investigated in the future.

5.2.1.1 Evaluation Metrics

The performance of our robust clustering algorithm is evaluated using three metrics: Silhouette Score (SS), Davies-Bouldin Index (DBI), and Calinski-Habrasz Index (CHI)[111]. We compare the performance of the robust clustering algorithm with the K-means clustering applied to each feature selection module (see step 4). Each set of features is used with robust clustering to calculate the metrics. The results of the metrics are shown in Table 5.1. The SS ranges from -1 to 1, with a score closer to 1 indicating denser and more distinct clusters. The robust clustering approach outperforms K-means clustering. The original module achieved the highest SS . In contrast, a lower DBI represents a better separation among clusters. The robust clustering with all modules has a DBI than k-means clustering, with the original module having the lowest index. A higher CHI indicates better clustering performance, and the robust

clustering approach with all modules outperforms K-means clustering, with the original module achieving the highest index.

5.2.2 The Nudging Effect on Tracking Activity

The previously proposed activity profile framework requires users to consistently wear their activity trackers (i.e., wearables) and report their daily activities. However, attaining consistent use and adoption of wearables is a challenge [97]. Nudging is a key mechanism to increase and motivate regular human engagement [110]. Therefore, we aim to study the impact of nudging strategies on individuals' adherence to wearing activity trackers. Even though several studies have explored the use of nudging theory to change human behavior [110]. Most of these studies focused on changing instant behavior in the short term. In addition, the activity of wearing a tracker requires daily motivation and remembrance. Hence, it may require different nudging strategies. Most of these studies would also try a different nudging strategy for each group. This suggests that people will respond to nudging strategies similarly [110][112]. However, individuals react differently based on their characteristics and preferences. Therefore, this study aims to explore the effect of several nudging strategies on each individual over time. For instance, all participants were studied using the same nudging strategies. Furthermore, our study used different email reminders as a nudging tool. The emails reminded participants to wear and use their gadgets once every month. Then we assessed the change in their device-wearing behavior regarding the number of days they wore them.

5.2.2.1 Study Method

Overall, we had recruited 28 participants to wear the Fitbit smartwatch and report health data for around six months (gender: female = 13, male = 15; age: $M = 21.71$, $SD = 3$; BMI: $M = 23.46$, $SD = 3.95$; education: High diploma = 10, Bachelors = 11, Masters = 3, PhD = 1). Our data was collected through the Fitbit smartwatch remote back-end database. Fitbit smartwatch tracks user health data such as average breathing

rate, steps count, and consumption of calories to the remote server in real-time. At the beginning of this study, we provided all participants with a Fitbit smartwatch and requested them to wear it daily. We computed their active rate as the percentage of the days they wore the devices over a month. A device was considered worn if the number of steps is more than one. Based on their first-month reporting active rates (baseline period), 28 participants were evenly divided into the experimental and control groups. Starting from the second month, we sent short emails containing a nudging message to the experimental group at the end of each month. In total, there are five different nudging strategies that we have used in our emails: 1) Neutral - which contains the basic monthly activity data without emotional bias; 2) Positive - motivational message, which explains the positive impact of their consistency in wearing the devices on their health and our research; 3) Negative - a message that informs the participants that a third of them have not worn their devices the past 30 days; 4) Rank - a message that tells each participant their rank among all participants in the previous month and how consistent the top three participants were; and 5) Monetary - a message that reminds participants that a monetary award would be obtained at the end of the month for their participation.

5.2.2.2 Results

The first experiment studies the impact of each nudging strategy by comparing the average active rates between the experiment and the control groups each month (see Fig. 5.13). Note that the experiment group's participants who did not open their email in a month were dropped from that month. The preliminary results show that the experimental group had a higher activity rate than the control group, except when the message was negative. According to the results, all nudging methods increased the active rates of participants by 5% to 20% except for the negative message, which slightly reduced the participation by 3% (see Fig. 5.14). Additionally, the results show that the positive message had the highest impact. Another observation is that, generally, the active rates of the participants of both groups had declined over time. This was due to the natural habit of humans losing motivation as time passes [108].

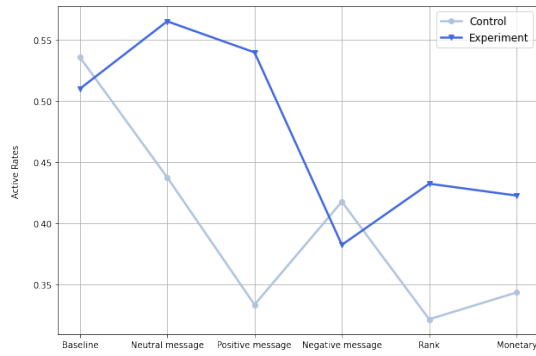


FIGURE 5.13. Active rates of both groups each month

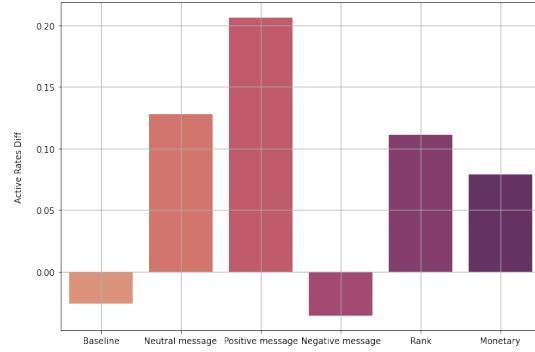


FIGURE 5.14. Average active rates difference between experiment and control groups each month

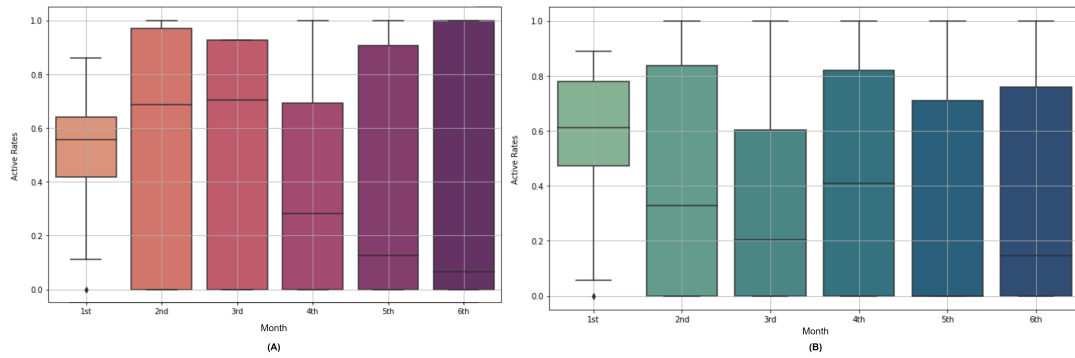


FIGURE 5.15. Active rates distribution in each month for (A) Experiment group (B) Control group

We analyzed the participants' active rates distribution to ascertain the previous experiment. The previous experiment only considered the average active rate in each month. We visualized the distribution of both groups as shown in Fig. 5.15. Both figures generally show a wide distribution of active rates across the months except for the first baseline month. This distribution indicates that: (1) Nudging techniques impact people differently, e.g., some will be more motivated by the positive messages while others will be motivated by ranking. This observation shows us that individuals' behavior varies with nudging strategies. A customized nudging mechanism is needed to identify the proper nudging for each individual based on their previous behavior. (2) People's behavior in consistently wearing their Fitbit is different; this is reflected in the active rates distribution of the control group (Fig. 5.15(B)). Lastly, the distribution of

the active rates in the first month is condensed. This may be justified by participants' excitement at the start of the project [108].

5.3 Conclusion

This chapter introduced our wireless energy-sharing setup for IoT devices. This setup includes a platform that enables energy transfer between smartphones. We collected real data on this wireless energy transfer. An energy prediction model was also developed, which assists in forecasting the availability of energy services. Furthermore, we proposed an activity-based profiling framework, which we anticipate employing in the future to create an energy harvesting model. This framework necessitates consistent reporting from IoT users. Acknowledging the challenge of maintaining regular wearable device reporting, we conducted a study to understand the impact of nudging on the regular use of these devices.

Energy Services in Drone Swarm Delivery

This chapter addresses the efficient composition of energy services within dynamic environments represented by drone swarms. The focus on drone swarms is purposeful; their operational dynamics present unique challenges for energy management, making them an ideal testbed for our research objectives. Here, we introduce a novel framework designed to enable dynamic in-flight energy sharing, optimizing the swarm's overall mission efficacy. This exploration is pivotal for demonstrating the practical application of our theoretical models and advancing the understanding of energy efficiency in autonomous IoT systems. In this respect, we aim to demonstrate the adaptability of our energy composition strategies and their potential to enhance service reliability in complex, energy-constrained IoT environments.

6.1 Introduction

Combining unmanned aerial vehicle technologies with the service paradigm has given rise to the concept of Drone-as-a-Service (DaaS) [113]. A drone may sense the surrounding environment and carry payloads. The DaaS concept wraps the capabilities of drones and offers them as consumable services in different domains such as traffic monitoring [114], firefighting [115], and delivery [116]. In exceptional circumstances such as pandemics, drones help create more resilient supply chains and socially distanced delivery services [117]. Indeed, drones may deliver goods autonomously, quickly, and contactless to customers. While single drone delivery provides a multitude of opportunities, there are instances where a *swarm* of drones may be needed to fulfill the requirements of a delivery request. Hence, the Swarm-based DaaS (SDaaS)

augments the DaaS concept to describe services delivered by a swarm of drones instead of single drones [8].

A swarm refers to a set of drones that act as a single entity to achieve a specific goal. In delivery, a requested package may exceed a single drone payload capacity due to strict federal aviation regulations that approve only small drones for delivery in the city (payload < 2.5 kg) [118]. In this case, a divided package could be carried by multiple drones in a swarm. A swarm also effectively distributes the energy burden among the drones by distributing the payload, allowing the swarm to travel farther destinations. However, these advantages come with the cost of multiple constraints, including the limited arrival time of all the drones in the swarm to the destination. In addition, drones in a swarm would usually carry different payloads, which causes different rates of energy consumption within a swarm.

The limited battery capacity of drones presents a major challenge in swarm-based drone delivery. Several solutions have been proposed to address this limitation. Intuitively, battery replacement could be the first solution to adopt. However, it requires human involvement to replace the drones' batteries. Unfortunately, most automatic battery replacement approaches exhibit inadequate accuracy in outdoor environments [119]. Carrying additional batteries could be another solution. However, the payload capacity should be utilized for the carried package, and adding an extra payload means extra energy consumption. As a solution for the drones' battery-limited capacity, solar charging depends highly on the weather conditions that may not always be practical to adopt [120]. Presumably, a swarm could stop at intermediate recharging stations to recharge the batteries [8]. However, sequential charging in the case of limited recharging pads would extend the charging and waiting times, which affects the overall delivery time of a package. Hence, *we propose to leverage an in-flight energy-sharing between the drones in the swarm to optimize the package delivery time.* We introduce the concept of *support drones* that fly with a swarm to in-flight charge other drones.

In-flight charging refers to the technology that enables recharging drones' batteries on the fly. For example, the GET Air solution provides an in-flight wireless power

charger to give about 25 minutes worth of endurance [121]. The drone only needs to hover over a recharging power station for six minutes. A new approach allows drones to deliver power wirelessly while in-flight [122]. In this case, the drone transmits signals that can be harvested and rectified to DC voltage. Inspired by this emerging technology, we propose to leverage in-flight charging to define *energy-sharing* services between delivery drones and recharging support drones. Energy-sharing services shall allow drones in a swarm to travel to farther destinations, reducing the number of stops needed to recharge. *Energy sharing* is the process of sharing energy among IoT devices [25]. An IoT device that shares energy is called an *energy provider*. In contrast, an IoT device that requires energy is called an *energy consumer*. In a swarm-based drone delivery services context, we view a drone that transmits signals as the provider. The consumer is the drone that rectifies the signals to DC voltage.

Swarm-based drone delivery services lend themselves quite naturally to being modeled using the service paradigm because they map to the key ingredients of the services concept, i.e., functional and non-functional attributes [8]. The function of an SDaaS is the swarm-based delivery of packages from a source to a destination. The non-functional properties include energy consumption, delivery time, etc. We also leverage the service paradigm to model energy-sharing services, termed as *Energy-as-a-Service* (*EaaS*) [23]. The function of EaaS is to share energy between a provider and a consumer. The non-functional properties include the amount of energy, sharing duration, etc. We propose a nested two-level service composition framework that encompasses SDaaS with EaaS.

We summarize our main contributions in this chapter as follows:

- A novel in-flight energy-sharing model in swarm-based drone delivery services.
- A two-level composition framework of Swarm-based Drone-as-a-Service and Energy-as-a-Service.
- A Constraint-aware SDaaS using a modified A* heuristic algorithm to compose the optimal delivery services.

Combining unmanned aerial vehicle technologies with the service paradigm has given rise to the concept of *Drone services* [113]. It is defined as using drones to provide services [113]. A drone may sense the surrounding environment and carry payloads. DaaS is used in such domains as traffic monitoring [123], agriculture [124], and delivery [113]. Drone *delivery services* are the new technology that promises impactful and innovative solutions to the ever-expanding online shopping [8]. Drones provide a cost-effective and environmentally friendly alternative compared to terrestrial delivery services like trucks [8]. Examples of terrestrial services include FedEx and UPS. In addition, drones enabled mitigating the effect of pandemics such as COVID-19 by creating more resilient supply chains and socially distanced delivery services [117]. In that regard, the use of drones allows for the autonomous delivery of goods in a *quicker, cheaper, and safer*, i.e., contactless manner. Major retailers are already integrating drones within their delivery options after governments worldwide relaxed regulations in response to the COVID-19 pandemic. For instance, Google's Wing has expanded its drone delivery services by doubling its deployment rates in Australia [125].

While single drone delivery provides a multitude of opportunities, there are instances where a *swarm* of drones may be required to fulfill the requirements of a delivery request. These include instances where a package is too heavy to be delivered by any one single drone or when multiple packages need to be delivered together. Federal aviation regulations dictate that payloads per drone not exceed 2.5 kg [126][127]. Hence, swarm-based drone services are an effective alternative and solution to address these limitations [8]. *The focus of this research is on the use of drone swarms for delivering packages as a service.* More formally, a swarm is defined as a set of drones that move together within proximity and arrive at any stop within a time window, thus acting as a single entity to achieve a specific goal. Swarms of drones have been used in various applications, including military [128], airborne communication networks [129], sky shows [130], and delivery [8].

The *limited battery capacity* of drones presents a major challenge in swarm-based drone delivery services. Several solutions have been proposed to address this limitation. Intuitively, a battery replacement would be the obvious solution. However, most automatic battery replacement approaches exhibit inadequate accuracy in outdoor environments [119]. Carrying additional batteries may be another solution. However, given the drones' small size, this would severely limit their delivery load. Solar energy is another option. However, its dependence on the weather conditions and slow charging process would make it hardly practical [131]. Moreover, the size of the solar panels would add to the payload [131]. Another suggested solution is the use of laser beams fitted on rooftops. This would not be applicable in a delivery scenario, as the beam range will vary when the drones move. Furthermore, the line-of-sight would be blocked by the urban environment [131]. A more realistic option is using *recharging pads* at delivery/recharging stations [8]. A major challenge is the limited number of recharging stations, thus limiting the parallel recharging of drones. This may result in extra waiting times until all swarm drones are recharged, as they act as an atomic unit for delivery purposes. Consequently, delivery times would likely be negatively impacted. *We propose using in-flight recharging drones to speed up swarm-based drone package delivery services.* We introduce the concept of *support drones* that fly with a swarm to charge other drones in-flight.

In-flight charging refers to the technology that enables recharging drones' batteries on the fly. For example, the GET Air solution provides an in-flight wireless power charger that can give a maximum output power of 12 kW [121]. The drone only needs to hover over a recharging power station for six minutes. A new approach allows drones to deliver power wirelessly in-flight [122]. In this case, the drone transmits signals that can be harvested and rectified to DC voltage. We focus on *energy-sharing services*, using in-flight recharging as an overlay to expedite swarm-based drone service delivery. Energy-sharing services enable drone swarms to reach more distant destinations, thereby minimizing the number of stops required for recharging. *Energy sharing* refers to the general process of sharing energy among IoT devices [23]. An IoT device (in our case, a drone) that shares energy is called an *energy provider*. In contrast, an

IoT device that requires energy is called an *energy consumer*. In a swarm-based drone delivery services context, we view a drone that offers energy as the provider and the drone that receives the energy as the consumer.

Swarm-based drone delivery services lend themselves quite naturally to being modeled using the service paradigm because they map to the key ingredients of the services' concept, i.e., functional and non-functional attributes [8]. The *function* of a swarm-based drone service is the delivery of packages from a source to a destination by a swarm. The *non-functional* properties include energy consumption, delivery time, etc. Likewise, we leverage the service paradigm to model energy-sharing services, termed as *Energy-as-a-Service (EaaS)* [23][25]. The function of EaaS is to share energy between a provider and a consumer. The non-functional properties include the amount of energy, sharing duration, etc. We propose a *nested two-level service composition* framework that consists of composing swarm-based drone services while achieving an Energy-as-a-Service composition (EaaS). In this respect, the first level composition of energy services acts as a non-functional QoS for the second level of composition of swarm-based services. Therefore, finding an optimal plan for swarm-based drone services for delivering goods from a source to a destination is a two-step service composition: 1. finding the best energy service composition between the two connected nodes, and 2. finding the best path using other criteria (e.g., time, cost, etc.) between source and destination. The main contributions of this chapter are as follows:

- A novel in-flight energy-sharing model in swarm-based drone delivery services.
- A Constraint-aware swarm-based drone services composition using a modified A* heuristic algorithm to compose the optimal delivery services.
- A two-level composition framework where the first level is the composition of energy services and the second level is the composition of swarm-based services.

6.2 Motivating Scenario

Consider the scenario of delivering medical supplies using a swarm of drones in a city. We consider a skyway network where the nodes represent buildings' rooftops. Each rooftop may be a delivery destination, a recharging node with a number of stations, or both (Fig. 6.1). Suppose a swarm delivers multiple packages from a supplier to a hospital. The requirement is that all packages must arrive as soon as possible and together within a predefined timeframe [132]. Such requirements stem from the inherent interdependencies of the supplied packages. For example, certain medical drugs need to be mixed at the hospital, but compounds would need to be delivered separately and within a timeframe [133]. Hence, they need to arrive at the same time. Fig. 6.1 represents a skyway network. We assume that the packages requested are of different weights. An optimal composition of swarm-based drone services results in the optimal path from source to destination with the fastest delivery time operating under constraints. The constraints include different intrinsic and extrinsic constraints. Intrinsic constraints include the different energy consumption rates in the swarm due to different payloads and battery capacities. A drone carrying a heavier payload consumes more energy than a drone with a lighter payload [8]. The extrinsic constraints include the limited recharging pads at a station, wind conditions, and the time-constrained arrival of packages. We assume that swarms are *static*: 1. the number of drones in a swarm is constant, and 2. drones in a swarm stay together spatially until they reach their destination [134]. We also assume that the optimal swarm *formation is preset, i.e., has already been decided and selected*, based on the averaged wind conditions and will not change until packages are delivered [118]. We adopt the five formations proposed in [118], including Column, Front, Echelon, Vee, and Diamond. These formations are inspired by bird flocks and military airforce for energy conservation purposes and protection [118]. An optimal formation consumes the least amount of energy under certain wind conditions [118]. We assume each skyway segment may have a different wind speed and direction.

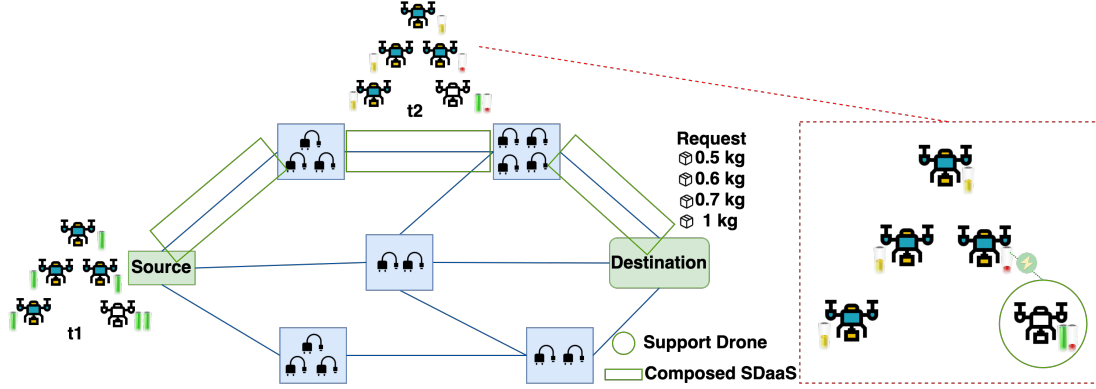


FIGURE 6.1. Swarm-based delivery services with in-flight energy-sharing

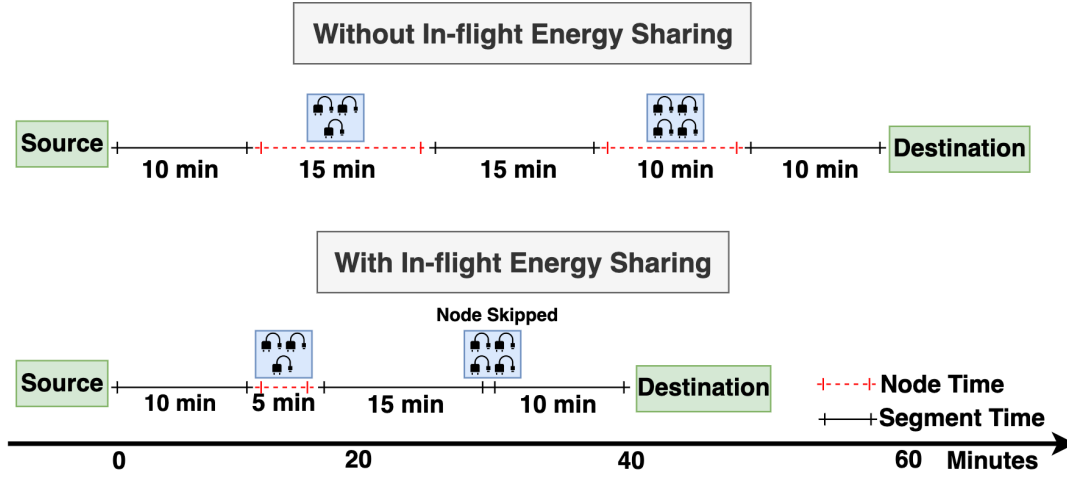


FIGURE 6.2. Effect of in-flight recharging on swarm-based drone services composition

We aim to minimize the swarm-based delivery time by *reducing the number of stops and recharging time at intermediate nodes in the skyway network* (Fig. 6.2). The recharging time of a swarm at an intermediate node includes time for recharging and waiting. A drone must wait to be recharged when no recharging station is available. We assume that all the recharging pads are similar, and hence the charging rate is the same. In this context, given a preset swarm formation, we propose leveraging the concept of in-flight charging to reduce the number of stops and recharging time at intermediate nodes. We aim to adopt the concept of Energy-as-a-Service and implement it within a swarm. As shown in Fig. 6.1, a swarm has a designated *support drone* whose

sole job is to recharge other drones in the swarm. We assume that the support drone carries a larger battery to share energy with other drones. In that respect, the support drone is considered an Energy-as-a-Service (EaaS) provider. The rest of the drones are EaaS consumers. The energy-sharing is assumed to occur during the swarm flight in a segment. Since there are multiple EaaS consumers and a limited flight time within a segment, there is a need for an EaaS provider (i.e., support drone) to select an optimal set of energy requests emanating from EaaS consumers over a period of time. We assume that the provider can only share energy with a single consumer. Hence, the provider should select the best consumer candidate at a time.

In most types of formations (e.g., Vee formation), each drone would consume a different amount of energy that is dependent on its position in the formation due to wind drag forces, and upwash/downwash forces generated by front drones [118]. The in-flight energy-sharing occurs at less than 1.2 meters distance between the provider and consumer [122]. We assume that the position of the support drone should be neighboring the currently selected consumer. Therefore, the swarm may need to reorder itself to facilitate energy-sharing. Additionally, the reordering problem is compounded by the different energy consumption rates of the drones according to their positions. In that respect, *we propose to study the position of a support drone in a swarm* to examine the effect of reordering the swarm on energy consumption.

6.3 Related Work

We identify three areas of research related to our work: (1) *swarm-based drone services*, (2) *energy-sharing and wireless energy transfer*, and (3) *service selection and composition*. As the latter two are presented in chapter 2, we describe state of the art in Swarm-based Drone Services and highlight the particular gap our research aims to address.

6.3.1 Swarm-based Drone Services

The use of drones for delivery is receiving increasing investment and interest from industry and research. Drones are used to deliver medicine [135], mail [136], and pizza [136]. The benefit of using drones in delivery over traditional methods is time sensitivity [135]. Drones are fast and can deliver packages to areas that are hard to reach with vehicle-based services [137]. A natural progression of using single drones in different applications, including delivery, is the use of multiple drones cooperatively to achieve a common goal.

Tackling tasks as a collective unit adds to the benefits of a single drone. *Drone swarms* could cut down the time needed in target search [138], surveillance missions [123], site mapping [139], and agriculture applications [140]. In addition, it could achieve tasks that a single drone can not achieve. This includes fascinating sky shows [130] and the delivery of multiple packages to the same destination at the same time [8]. Multiple works in swarm-based deliveries refer to coordinating multiple individual-operated drones as a drone swarm [141][142]. However, *we refer to a swarm as a set of drones that act as a single entity and are bounded by a space window*.

The large-scale adoption of drones in deliveries comes with technical challenges to overcome. Power is considered the bottleneck of the drone industry when traveling long distances to deliver packages [135]. Solutions proposed include creating new types of batteries that live longer than the traditional Li-Po batteries [143]. Solar-powered drones are also developed to extend the battery life [144]. CyPhy proposed a micro-filament solution to have an infinite flying time by connecting the drone with a micro-filament wire to batteries on the ground [145]. Battery replacements throughout travel and the use of wireless charging stations midway are also proposed [113]. However, these solutions suffer from a lack of mobility, weather condition reliance, human-power dependability, and added recharging times. In drone swarms, an alternative solution is proposed that utilizes *drone flight formations* to enhance the energy consumption during delivery missions [118].

The study of flight formations has been widely explored in military and bird migrations. A swarm in the military takes different formations to allow for better protection, view, and aiming [146]. Birds take different formations for energy conservation purposes [147]. As a bird flaps, a rotating vortex of air rolls off at each of its wingtips. This causes the air directly behind the bird to be pushed downwards (downwash). In contrast, the air to the side of the bird gets pushed upward (upwash). The upwash forces give a free lift to the birds at the back, causing them to consume less energy on flapping. Hence, we see birds flying in Vee formations [147]. The heart rate of birds flying in a vee formation was lower, indicating less energy consumption [148]. This is especially important when birds travel for long distances during migration.

In swarm-based drone applications, the study of formation flying is still in its infancy. A swarm of UAVs flies in different geometrical flight formations for 3D scene reconstruction purposes [149]. For energy conservation purposes, using a Vee formation in a fixed-wing UAV swarm has proven to consume less energy and behave similarly to birds [150]. In addition, the effect of different formations on the energy consumption of quadcopters was studied under different wind conditions [118]. Flying in a formation consumes less energy for different wind conditions due to reduced drag and upwash/downwash forces. The study also presented that drones consume different amounts of energy based on their *position* in a formation [118]. However, to the best of our knowledge, the effect of reordering the drones in a formation was not explored previously. We propose to optimize the distribution of energy consumption between the drones in the swarm by *reordering the drones within a formation*. In addition, we propose to overcome the power limitation in swarm-based deliveries using *in-flight energy-sharing* within a swarm.

The challenges in service computing can be grouped into four areas: service design, service composition, crowdsourcing-based reputation, and IoT [18]. The work proposed in this chapter lies mainly under the two umbrellas of service composition and IoT. Drones play a role in IoT as they depend on sensors and embedded software to communicate to achieve their goal [151]. In drone services specifically, a model

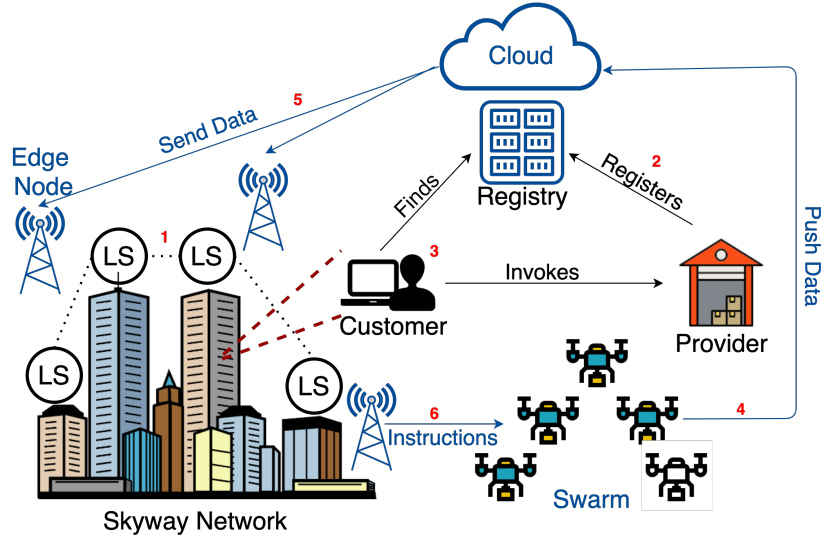


FIGURE 6.3. Swarm-based drone services system architecture

for single drone-based services was proposed [113]. The single drone-based services model was defined by its functional and non-functional (QoS) properties, including delivery time and cost. Each skyway segment is abstracted as a drone service that may be served by a single drone. Their work only focuses on the composition of single drone services. This concept was augmented to swarm-based drone service to compose the optimal path of a drones swarm from a source to a destination under various constraints [8]. The concept of wireless energy-sharing services between IoT devices has been previously defined as Energy-as-a-Service (*EaaS*) [23]. The existing work focused on composing services in a confined area to fulfill the consumers' queries [23][49]. To the best of our knowledge, there is no previous work on composing swarm-based delivery services based on swarm formations, energy-sharing, and reordering. Hence, this work is the first attempt to model energy and time-constrained delivery environment for swarm-based drone deliveries using *in-flight energy-sharing* and formation reordering using a service-oriented approach.

6.4 System Architecture and Composition Model

We present a high-level system architecture for swarm-based delivery services. As shown in Fig. 6.3, the architecture is premised on the delivery of multiple packages to a single destination in a line-of-sight skyway network with landing spots (LS) at each node (rooftops): (1) The architecture is an adopted service-oriented architecture (SOA) with consumers, providers (warehouses), and registries. The *provider* registers their swarm-based delivery services in the registry for advertisement; (2) *Consumers* would view, select, and invoke the desired services from the registry; (3) When a customer invokes a swarm-based drone service, the delivery management system determines the set of drones needed, including the support drones and the path to be taken. The delivery management system shares the computations among the swarm, cloud, and edge nodes. The simple computations, including analysis of sensor data and battery information, will be computed at the *swarm* level. More sophisticated and expensive computations will be performed at the *edge* level. These computations include swarm-based drone services and EaaS compositions. The edge nodes are distributed in strategic locations to aid delivery allowing for lower latency and faster communication between the swarm and the edge. Computations requiring lots of data will be done on the *cloud* level. Such computations include navigation throughout the skyway network. As the swarm moves, the drones communicate their data, i.e., battery states and locations, to the cloud for storage purposes; (4) The cloud frequently advertises this data to the edge nodes to help in decision-making; (5) The swarm receives multiple instructions, such as picking up the packages, the path to the destination, recharging, etc. These instructions are communicated with the swarm from the cloud through the edge server; (6) On arrival of the swarm to the destination, the customer is notified.

As explained earlier, we abstract each swarm traveling in a skyway segment as a swarm-based drone service. We aim to select and compose the optimal set of swarm-based drone services from a source point to a destination point, given the highly constrained environment due to different carried payloads, limited charging spots, and constrained delivery time windows. We then aim to further optimize the swarm-based

drone service composition by incorporating Energy-as-a-Service (EaaS). This study considers the environment deterministic, i.e., the surrounding conditions do not change after compositions are decided. We formally define a Swarm-based Drone Service. Then we define a swarm-based drone service customer request. Later, we formally define an Energy-as-a-Service (EaaS).

Definition 1: Swarm-based Drone Service (SDS). It is defined as a tuple of $\langle SDS_id, S, F \rangle$, where:

- SDS_id is a unique identifier.
- S is the swarm traveling in a swarm-based drone service. S consists of:
 - D , which is the set of delivery drones and support drones forming S , a tuple of D is presented as $\langle D_{d1}, D_{d2}, \dots, D_{dn} \rangle$ and $\langle D_{s1}, D_{s2}, \dots, D_{sm} \rangle$.
 - The battery levels of every d in D $\langle b_1, b_2, \dots, b_n \rangle$.
 - The payloads every d in D is carrying $\langle p_1, p_2, \dots, p_n \rangle$.
- F describes the delivery function of a swarm on a skyway segment between two nodes, A and B. F consists of the travel time tt , charging time ct , and waiting time wt when recharging pads are not enough to serve D simultaneously in node B. In many instances, the number of delivery drones D would be larger than the number of available recharging pads at a node. Therefore, the drones will be unable to recharge simultaneously and will have to wait for each other.

Definition 2: Swarm-based Drone Service Request (R). A request is a tuple of $\langle \alpha, \beta, P \rangle$. α is the source node, β is the destination node, and P are the weights of the packages requested, where P is $\langle p_1, p_2, \dots, p_n \rangle$.

Definition 3: In-flight Energy-as-a-Service (EaaS). We adopt the definition of $EaaS$ in [23]. An $EaaS$ is defined as a tuple of $\langle id, F, Q \rangle$, where:

- $EaaS_id$ is a unique service identifier.
- F is the function of sharing energy by a support drone.

- Q is a tuple $\langle ae, loc, st, et \rangle$ where each attribute donates a QoS property of $EaaS$ as follows: ae is the amount of available energy that a support drone can share, loc is the location of a support drone $\langle x, y \rangle$, st is the start time of a support drone's $EaaS$, and et is the end time of a support drone's $EaaS$.

Definition 4: In-flight Energy Service Request (ER). We adopt the definition of ER in Chapter 3. An ER request is defined as a set of $\langle id, F, QR \rangle$, where:

- ER_{id} is a unique energy service request identifier.
- F is the function of receiving energy by a delivery drone.
- QR is a tuple $\langle re, loc, st, et \rangle$ where each attribute donates a requirement property of ER as follows: re is the amount of requested energy by a delivery drone, st , and et are the start and end times of the delivery drone's receiving duration, and loc is the location of the delivery drone.

Table 6.1 serves as a reference for the meanings of the main symbols used in this chapter.

TABLE 6.1. Summary of notations

Symbol	Meaning
SDS	Swarm-based Drone Service
S	Swarm
D_d	Delivery drone
D_s	Support drone
dt	Total delivery time of the packages
tt	Travel time within a segment
nt	Node time (time spent at a node to recharge the swarm)
ct	Charging time (time spent by a drone recharging, depends on the charging pad rate and required energy)
wt	Waiting time (time spent by drones waiting for other drones to finish recharging due to the limited number of pads)
$EaaS$	Energy-as-a-Service
ER	Energy request
st, et	Start and End times of a provisioned energy request

6.4.1 Problem Formulation

Given a swarm-based drone service request from a consumer, the problem is formulated as composing the best *SDS* services from the source node α to the destination node β using in-flight energy-sharing. Several constraints inform the composition. First, the payload and battery limitations of the drones are intrinsic constraints. Second, the availability of the charging pads and the wind conditions are extrinsic constraints. Third, in some instances, the rate of in-flight recharging might not be sufficient to cover the energy loss due to formation reordering and re-positioning the drones within a formation. As described earlier, drones consume different amounts of energy based on their positions within a formation. Moreover, there are a set of challenges when addressing *SDS* composition with in-flight energy-sharing. First, the selection of which delivery drone to charge, when to charge, and for how long to charge affects the planning of the journey. Second, the composition of the next service, in addition to the aforementioned constraints, depends on the gain provided by in-flight energy-sharing. Third, deciding the number of support drones D_s to carry out the mission is an extra challenge that depends on various variables like the size of the swarm, the carried payloads, and the wind conditions.

Given all the aforementioned constraints and challenges, our goal is to compose two-level services optimally. First, composing *SDS* services from the source to the destination with the shortest delivery time ($\min(dt)$) and second, composing *EaaS* services within every *SDS* service. In the second level *EaaS* composition, the goal is to allocate the energy needing delivery drones to the support drones for an overall energy gain that will reduce the charging time at the next node. The *EaaS* provisioning is constrained by the start times st and end times et of energy requests ER by the delivery drones D_d and the st and et of the support drones *EaaS* service.

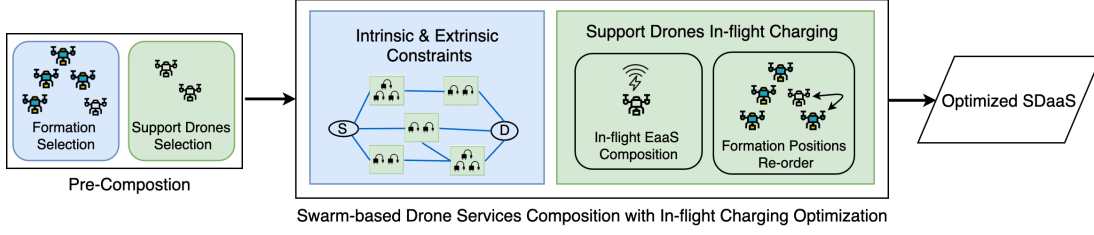


FIGURE 6.4. Swarm-based drone service composition framework

6.5 Swarm-based Drone Services Composition Framework

The composition framework consists of two main modules, namely, swarm-based drone services *Pre-composition*, and swarm-based drone services *Composition with in-flight charging*. The second module is a nested two-level swarm-based drone service *SDS* and *EaaS* compositions. The result of these modules is an optimized composition of swarm-based drone services *SDS*. Fig. 6.4 depicts the steps involved in the proposed *SDS* composition framework.

6.5.1 Swarm-based Drone Service Pre-composition

Before composing the set of optimal services from a source to a destination, we need to select the optimal swarm formation and support drones. In this section, we discuss how to select the optimal formation as proposed by [118], select the optimal number of support drones, and where to position the support drones in a formation.

6.5.1.1 Formation Selection

The selection of the swarm formation is essential for an energy-optimized composition of *SDS* services [118]. Drones in different swarm formations consume different amounts of energy for two reasons. First, a drone in a formation experiences upwash and downwash forces from other drones in the swarm. These forces cause drones to consume different amounts of energy based on their position in the formation. Second,

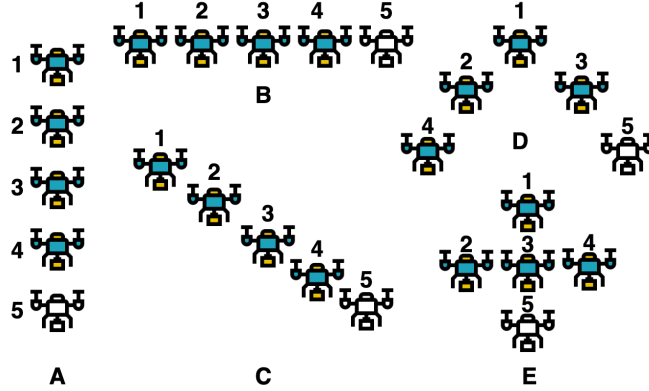


FIGURE 6.5. Different swarm formations A: Column, B: Front, C: Echelon, D: Vee, E: Diamond

a drone in a formation experiences diverse drag forces based on its position and the external wind direction and speed. Hence, it is crucial to select an optimal formation to minimize the energy consumption of the *SDS*. For example, with a front wind, drones in a Vee formation consume the least energy compared to other formations. On the other hand, with a side wind, a diamond formation consumes less energy [118]. In this study, we adopt a *fixed formation* for the swarm-based drone services [118]. once an optimal formation is selected, we assume it will not change throughout the journey to the destination. Five formations are identified, including Column, Front, Echelon, Vee, and Diamond. Fig. 6.5 shows the different formations. We adopt the study done in [118] to compute how much each drone consumes energy in different formations under different wind conditions.

6.5.1.2 Support Drones Selection

Redundancy is the duplication of critical components of a system to either increase the reliability of a system or to improve actual performance [152]. In this study, we adopt the concept of redundancy to improve the performance in terms of energy consumption and delivery time by having *support drones*. We consider the inclusion of support drones to be equivalent to redundancy. This is because we assume the support drones have similar specifications to the delivery drones. The only difference is the payload carried from extra batteries and energy transmitters. This is analogous to disk drives,

which can be redundant for efficiency rather than just reliability [153]. In this step, we study the optimal number of support drones for a particular delivery request using the redundancy theory [152]. We then study the options for positioning the support drones in the formation before the *SDS* services composition.

There are different types of redundancies based on the needs of a system [152]. These include $N + 1$, $N + 2$, $2N$, and $2(N + 1)$ redundancies, where N is the number of components in the system. In our case, N is the number of delivery drones in the swarm before adding the support drones. Each redundant component, i.e., support drone, added to the system decreases the probability of failures. In our scenario, a failure (i.e., unsuccessful delivery) occurs when a swarm cannot reach the destination due to the battery limits of drones. In this study, we assume that the drones in the swarm are of the same size and specifications. For example, we use the DJI Phantom 3 drone model for all the experiments in this chapter. We also assume that a drone's max payload is 1.4 kg. The battery in the DJI Phantom 3 weighs 365 g. Hence, in the case of this drone model, we assume a support drone can carry three additional batteries increasing its energy capacity. We also assume that all the drones are initially fully charged. We consider the following attributes that affect the delivery to compute the probability of failure and select the optimal number of redundant components, i.e., support drones, for a certain delivery request:

- The size of the swarm without the support drones S_s . This is based on the number of packages in an *SDS* request as explained in Section 6.4.
- The payload ratio for each drone D_{pr} . The payload directly affects the energy consumption of a drone. Hence, we compute the average payload ratio for all the drones in the swarm using the following equation:

$$D_{pr} = \frac{\sum_i^n \frac{p_i}{\text{max payload capacity}}}{\text{no. of drones}} \quad (6.1)$$

where p_i is the payload drone i is carrying, the *max payload capacity* is 5 kg, and the *no. of drones* is the total number of drones in the swarm before adding any support drones.

- Distance between the source and the destination $d(\alpha, \beta)$. We use Dijkstra's shortest path to compute the distance.
- Energy capacity of a support drone E_c . This is 3+1=4 times the delivery drone, including the original battery, as explained earlier.
- The averaged wind conditions W . As described earlier, wind severely affects the energy consumption of a swarm. Hence, the wind factor needs to be considered in the support-drones selection.

We use the aforementioned delivery-affecting factors to compute how many redundant components, i.e., support drones, are needed. We do this by computing the probability of failure and matching the probability with the equivalent redundancy values in Table 6.2. We compute the probability of failure using the following equation:

$$P = \prod_{i=1}^4 w_i * \tilde{p}_i \quad (6.2)$$

where $p_1 = d_{pr}$ is the drone payload ratio, $p_2 = d(\alpha, \beta)$ is the distance between the source and the destination, $p_3 = E_{cp}$ is the energy capacity of the support drones, and $p_4 = W$ is the averaged wind conditions. The weights w_i in the equation are computed based on how much each factor *consumes energy per unit time independently*. For example, let us assume that a swarm consumes 3% of its total power to travel for 1 minute at a speed of 30 km/hr; hence, $w_2 = 0.03$. The distance between source and destination $d(\alpha, \beta)$ is 5 km. Therefore, $P_{i=2} = 0.03 * ((5/30) * 60) = 0.3$. The same is done to compute the rest of the weights w_i . Instead of taking the true value of each factor, we normalize them between 0 and 1 to unify the different ranges of the affecting factors $\tilde{p}_i$. Once the probability of failure is computed, the number of redundant components (support drones) is decided according to Table 6.2.

6.5.1.3 Support Drones Positioning

Once the number of support drones in a swarm is decided, we need to study the optimal position to place the support drones in the swarm. The positioning is important for a

TABLE 6.2. Redundancy values for different failure probabilities

Failure Probability P	Redundancy Number
0-19	N+1
20-39	N+2
40-59	N+3
60-79	N+4
80-100	2N

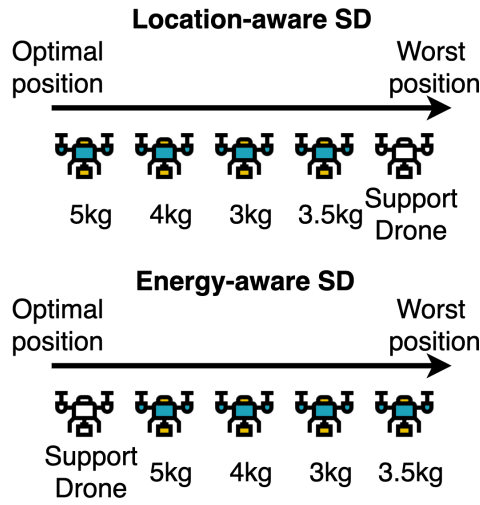


FIGURE 6.6. Support drones positioning

couple of reasons. First, a support drone can charge other drones up to 1.2 meters away [122]. Second, the position of the support drones in the swarm affects their energy consumption due to upwash/downwash and drag forces in different formations. The drones in different formations consume energy differently based on their positions in a formation and the wind direction. For example, a drone at the front of a Vee formation consumes the most energy if the wind comes from the front [118]. In this chapter, we propose to study the effect of two positioning settings. Fig. 6.6 describes the two settings. Note, in this step, *we are only selecting the initial position of a support drone*. Later, in Section 6.5.2.2, we will reorder them as needed.

- **Location-aware Support Drone:** Looking at the full swarm, i.e., the delivery drones and the support drones, we position the delivery drone carrying the maximum payload at the optimal position in the formation. In descending order, we position the drones carrying the heavier payloads in the optimal

positions and the lightest payloads in the worst positions. Last, we position the support drones in the worst positions. This setting would ensure that the delivery drones consume the least energy due to their positioning. However, it comes with the cost of non-optimal support drone positioning. The support drones are aware of their location and give up their prime position for optimal positioning of the delivery drones.

- **Energy-aware Support Drone:** Looking at the whole swarm, i.e., the delivery drones and the support drones, we position the support drones at the optimal position in the formation. Then in descending order, as the first setting, we position the delivery drones from optimal to worst positions based on their carried payloads. This approach would ensure that the support drones consume the least energy due to their positions and would have more energy to share with the delivery drones. However, the delivery drones would consume more energy in this approach. The support drones know their energy and are in the least energy-consuming positions.

As seen from the two settings, we need to compromise between the optimal positioning of the delivery drones and the support drones. Hence, in Section 6.6, we run experiments to test both approaches and conclude which setting to adopt.

6.5.2 Swarm-based Drone Services Composition with In-flight Energy Sharing

After pre-composition, we select and compose the optimal swarm-based drone services from the source to the destination using a modified A* algorithm. In this study, optimality is a single objective of composing *SDS* services with the least delivery time. We assume the swarm is *static*, i.e., the members are decided at the source node, and no additions or reduction of members occurs throughout the path to the destination [8]. Given a connected network, we compose the best available path from the source to the destination. The composition of the optimal services mainly depends on two

components, as shown in Fig. 6.4. The first is the intrinsic and extrinsic constraints surrounding the delivery environment. The second is the in-flight energy-sharing process that will affect the composition of the services.

A swarm-based delivery environment is highly affected by multiple constraints. The intrinsic constraints include the different energy consumption rates between the drones due to *payload* and *position in a formation*. The extrinsic constraints include the *wind conditions* that highly affect the energy consumption of a swarm and the different number of *recharging stations* available at every node to serve the swarm. The proposed swarm-based drone services composition framework addresses the aforementioned constraints and optimizes the selection of the services through in-flight energy-sharing.

Algorithm 6.1 describes the *SDS* services sequential composition with the in-flight energy-sharing process. The swarm computes the shortest path in terms of distance only between the source and destination [154], at the source node. If *all* the drones $D = D_d + D_s$ in the swarm S are able to reach the destination without the need to recharge at intermediate nodes, then the swarm will select this path and update the travel time tt (Lines 5-7). This capability is computed by considering the amount of consumed energy E_c due to payload, formations, distance, and wind conditions. Otherwise, the swarm estimates the effect of in-flight energy-sharing on the swarm to reach the destination directly (Line 8). The in-flight energy-sharing process consists of two main steps. Namely, *composing EaaS services* and *reordering* the drones in the formation when required to facilitate the energy-sharing process. These two steps are described in Section 6.5.2.1 and Section 6.5.2.2, respectively.

The proposed nested composition is used if the swarm cannot reach the destination directly, i.e., without stops at intermediate nodes, even with energy-sharing (Line 11). The path is composed *sequentially* based on our previously proposed approach [8] along with in-flight energy-sharing at each line segment in the composed path. The swarm selects the nearest reachable node with the least travel time tt and node time

Algorithm 6.1 Swarm-based Drone Services Composition with In-flight Energy Sharing Algorithm

Swarm S and SDS Request R

Input: S, R

Total delivery time from source to destination

Output: dt

```

1:  $dt = 0$ 
2: while  $S$  is not at destination do
3:   distance to destination = Dijkstra(current, destination)
4:   compute  $E_c$  for every  $D_d$  and  $D_s$  in  $S$  based on  $R$  package weights,  $S$  formation, wind condition, and distance to destination
5:   if all  $D$  in  $S$  can reach destination without intermediate nodes then
6:      $S$  travels to destination
7:      $dt += tt$ 
8:   else if all  $D$  in  $S$  can reach the destination without intermediate nodes using in-flight energy-sharing then
9:     In-flight_Energy_Sharing_Composition(battery[D], tt)
10:     $dt += tt$ 
11:   else
12:     find nearest neighbor nodes
13:     EaaS_Composition()
14:     select best neighboring node,  $\min(tt + ct)$ 
15:      $S$  travels to a neighboring node
16:     $dt += tt + ct + wt$ 
return  $dt$ 

```

nt (Lines 12-14). The node time consists of the charging time and the drone's waiting times to charge sequentially $nt = ct + wt$.

The charging time ct is the required time to charge the drone to 100%. As the energy consumption differs among the swarm members, we take the longest charging time ct to be the wt of parallel recharging drones. The waiting time wt highly depends on the number of available recharging pads. When the number of pads is less than the number of drones, an optimization problem occurs to find the best set of drones to charge together to minimize the node time nt . Since the number of pads is finite, we use a brute force approach to compute the node time nt . For example, if there are five drones with charging times $ct < 60, 50, 40, 30, 20 >$ minutes. Then the node time will be computed as follows:

- 1 pad available: node time = $60 + 50 + 40 + 30 + 20 = 200$ minutes.

- 3 pads available: pad 1: $60 = 60$, pad 2: $50 + 20 = 70$, pad 3: $40 + 30 = 70$, node time = $\max(60, 70, 70) = 70$ minutes.
- 5 pads or more available: node time = $\max(60, 50, 40, 30, 20) = 60$ minutes.

The node time nt is greatly affected by the in-flight energy-sharing process before it reaches the node. The drones will reach the node with higher battery capacity than they would without the energy-sharing process. Once the swarm is fully recharged, it tries to find if the destination is directly reachable and repeats the above steps until it reaches the destination. The total delivery time dt comprises the total travel time tt and the total node times nt (Line 16).

6.5.2.1 In-flight Energy Sharing Composition

The in-flight energy-sharing composition will be invoked in the path composition. The support drones will share energy with the neighboring drones in the swarm. In the case of multiple support drones, we assume each support drone will charge a subset of the delivery drones. We also assume that a support drone can share its energy with one delivery drone at a time [23][49]. The energy-sharing time window is the travel time between the current node and the next destination node, i.e., a line segment in the composed SDS services path. The delivery drones will trigger an energy-sharing request. Each drone has a different energy consumption model depending on several factors. The factors include the distance traveled, carried payload, wind conditions, and the drone's position within a formation. If any of these factors change during the composition, the energy consumption for each drone is recomputed. An in-flight energy-sharing request is launched once the drone's battery capacity decreases to a predefined threshold γ . The in-flight energy-sharing composition aims to select and compose the energy requests ER to maximize support drones' energy utilization. Maximizing the energy utilization will reduce the required time to wait and charge at charging pads in an intermediate node. We propose to compose ERs using two approaches and study their impact on travel time. The first approach, i.e., Priority-based (PB) energy-sharing, is inspired by the first come first served scheduling algorithm [95]. In the PB approach,

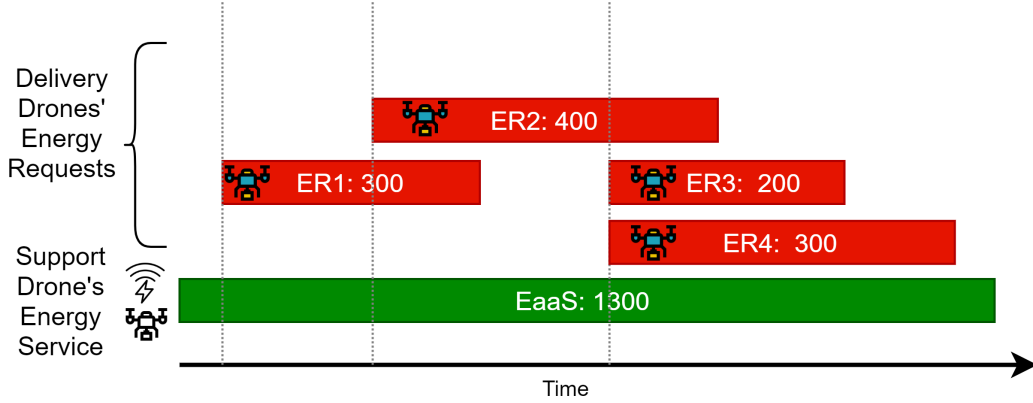


FIGURE 6.7. Drones' energy-sharing requests and service

the support drone charges drones based on the start time of their request. Moreover, the support drone fully charges each selected drone based on their requests. The second approach, i.e., Fairness-based (FB) energy-sharing, charges each drone partially in a round-robin manner.

Priority-Based (PB) Energy Sharing

This approach is a modified version of the first come first served scheduling algorithm [95]. Thus, the priority in selecting ERs will be defined by the requests' starting times st . In this approach, selected ERs will be charged fully according to their requested amount. Each delivery drone requesting energy defines the starting time st and the requested amount re for their request. If the support drone has enough spare energy to provide and there are multiple overlapping requests, the earlier request will be selected. If multiple requests start simultaneously, the request with the highest requested energy will be selected. For instance, using this approach on the given requests in Fig. 6.7, ER1 is selected since it is the earliest request.

Algorithm 6.2 presents the PB energy-sharing composition in detail. The algorithm takes as an input the delivery drones' battery $battery[D]$ and the segment travel time tt . The algorithm returns the updated delivery drones' battery $updated_battery[D]$ as an output. Line 1 retrieves the available energy requests (ERs) by checking the battery of the delivery drones. If the drone's battery capacity decreases during the flight on the

Algorithm 6.2 In-flight PB Energy Sharing Composition**Input:** $battery[D], tt$ **Output:** $updated_battery[D]$

```

1:  $ER = \mathbf{Generate\_ER}(battery[D])$ 
2:  $SortedER = \mathbf{sort}(ER, st : ascending, re : descending)$ 
3:  $st = tt.st$ 
4:  $et = tt.et$ 
5: for  $er_i \in SortedER$  do
6:   if  $er_i.st \geq SD.st$  and  $er_i.re \leq SD.ae$  then
7:      $st = er_i.et$ 
8:      $SD.ae = SD.ae - er_i.re$ 
9:      $ER\_composition.add(er_i)$ 
10:     $battery[D] = battery[D] + er_i.re$ 
11:     $SortedER.remove(er_i)$ 
12:     $battery[D] = \mathbf{Reoder}(\mathbf{Fixed/Flexible})$ 
13:     $newER = \mathbf{Generate\_newER}(battery[D])$ 
14:    if  $newER \neq \emptyset$  then
15:       $SortedER = SortedER \cup newER$ 
16:       $SortedER = \mathbf{sort}(SortedER, st : ascending, re : ascending)$ 
17:  $updated\_battery[D] = battery[D]$  return  $updated\_battery[D]$ 

```

line segment, then an energy request (er) is launched. The er 's start time will be when the drone's battery reaches less than a predefined threshold γ . Line 2 sorts the set of ERs in ascending order based on their start time, then in descending order based on the requests' size. In lines 3-19, for every er , the algorithm checks if: (1) the start time of the er falls in the period of the support drone service, and (2) the support drone has enough energy to fulfill the required energy, then the er will be selected. Once an er is selected, the support drone time and capacity will be updated (Lines 7-8). In addition, the er information will be added to the composition (Line 9). Line 10 updates the drone's battery of the selected er . Line 11 removes the er from the set of ERs . Moreover, a formation reorder function will be called (Line 12). The reorder function checks if a re-positioning of the drones is required to deliver the requested energy. The position of the drones affects their energy consumption, which may trigger new ERs that should be considered in the current in-flight energy-sharing composition. Therefore, reordering our energy-sharing composition approach will check if new ERs are

submitted by the delivery drones (Lines 13-17). The formation reordering approach is discussed in detail in Section 6.5.2.2.

Fairness-Based (FB) Energy Sharing

In this approach, a fixed amount of energy will be provided to each drone in a round-robin fashion based on the drone's id until the end time of the segment. The amount of energy may vary from small to large amount. Algorithm 6.3 describes the Fairness-based in-flight energy-sharing composition. The amount of provided energy is fixed based on a user-defined value λ (Line 3). Line 4 computes the charging time based on the charging rate of the support drone. The algorithm then goes through drones in order of ID to provide a fixed amount of energy (Lines 5-16). Note that the energy will be provided by the support drone in rotation between the delivery drones as long as: (1) the support drone battery is greater than a threshold δ . The threshold δ is used to keep enough spare energy for the support drone's energy consumption; and (2) there is enough time to charge a delivery drone. Similar to the Priority-Based energy-sharing approach, once a drone is selected (Lines 5-11), the battery of all drones will be updated (Lines 9-11), and a formation reorder function will be called (Line 12). The reorder function checks if a re-positioning of the drones is required to deliver the requested energy. As previously mentioned, the positions of the drones affect their energy consumption, which needs to be updated in the current in-flight energy-sharing composition. The formation reordering approach is discussed in detail in Section 6.5.2.2.

6.5.2.2 Formation Re-ordering

Once the in-flight energy-sharing process is initiated, the swarm may need to reorder itself to facilitate the process. As described earlier, the support drones could share their energy to up to 1.2 meters [122]. Hence, the support drones should neighbor the energy-requesting delivery drone. Therefore, the drones in the swarm swap positions to facilitate the process. For simplicity, we assume there is no overhead time or energy during reordering. However, the energy consumption rate of the drones will

Algorithm 6.3 In-flight FB Energy Sharing Composition

Input: $battery[D], tt$
Output: $updated_battery[D]$

```

1:  $ct = tt.st$ 
2:  $et = tt.et$ 
3:  $amountE = \lambda$ 
4:  $chargingt = amountE / chargingRate$ 
5: while  $ct < et$  and  $SD.ae > \delta$  do
6:   for  $d_i \in battery[D]$  do
7:     if  $ct < et$  and  $SD.ae > \delta$  then
8:        $ct = ct + chargingt$ 
9:        $SD.ae = SD.ae - amountE$ 
10:      if  $battery[D]$  is not full then
11:         $battery[D] = battery[D] + amountE$ 
12:         $battery[D] = \text{Reorder(Fixed/Flexible)}$ 
13:  $updated\_battery[D] = battery[D]$  return  $updated\_battery[D]$ 

```

change due to their position change in the formation. We recompute the new energy consumption model whenever reordering occurs. In this chapter, we investigate a fixed approach for swarm reordering.

In a fixed approach, the support drones do not change their positions. Once the energy-sharing is initiated and the distance between the support drone and the energy-requesting delivery drone is more than 1.2 m, the delivery drones swap their places to neighbor the support drones. Otherwise, the support drone directly shares its energy without reordering. In this method, we ensure that the energy consumed by the support drone does not change. If the support drone is placed in the optimal position initially, it will consume the least energy due to its position. On the other hand, the delivery drones swapped may be placed in a non-optimal position during the energy-sharing process. Once the energy-sharing process ends, the drones reorder themselves again to their initial positions.

6.6 Experiments and Results

We evaluated the proposed swarm-based drone services composition framework with in-flight energy-sharing. We assessed the proposed framework against three composition algorithms, baseline [8], Dijkstra’s [154], and Floyd-Warshall’s [155]. We first conducted a set of experiments to evaluate the *effectiveness* and *feasibility* of the system under the different controlling attributes, including the initial positioning and energy-sharing methods. We assumed the amount of energy shared per round for the Fairness-based energy-sharing method is 2,240 mA. We first compared the number of fulfilled requests under the different energy-sharing methods. Second, we compared the delivery times of the proposed requests under different settings. Finally, we compared the runtime efficiency of the composition methods.

6.6.1 Dataset and Experiments Setup

To the best of our knowledge, there is no publicly available data on drone trajectories in skyway networks. Therefore, the dataset used for the experiments is an urban road network dataset from London city to mimic a skyway arrangement. The nodes are the intersections in the city, and the edges represent the distances between the nodes [156]. We took a sub-network of connected nodes with the size of 195 nodes to mimic the skyway network for our experiments. We then synthesized 10,000 requests with random source and destination nodes. We also synthesize different wind speeds and directions for the skyway segments. We only considered wind speeds under 13.8 m/s, in which drones are safe to fly [118]. We have incorporated a real drone trajectory dataset in the energy model calculations. The dataset, however, had limitations of being only for single drones. Hence, it did not reflect the formations’ effect on energy consumption due to drag and upwash/downwash forces. Therefore, we used the payload and flight range effect on the energy consumption from the dataset. We then augmented the real dataset with the computational fluid dynamics study proposed in [118] to learn the behavior in terms of swarm formation and the wind effect on

TABLE 6.3. Experimental Variables

Variable	Value
No. of nodes in the network subset	276
No. of nodes in the largest connected network	195
No. of generated requests	10000
Max. weight of a package	1.4 Kg
Drone model	DJI Phantom 3
Speed of the drone	30 km/h
Battery capacity	4480 mAh
Voltage	15.2 V
Rate of in-flight energy-sharing	5.88 mAh/min

energy consumption. The maximum weight of a package was assumed to be 1.4 kg, following the payload capacity of the DJI Phantom 3 used in the experiments. The real-drone dataset uses the CrazyFlie 2.1 drone, which is a small drone with a 240 mAh and 3.7 V battery, compared to the DJI Phantom 3 with a 4,480 mAh and 15.2 V battery. Therefore, we multiplied the consumption rate by the increasing factor, i.e., 4.1. Table 6.3 summarizes the experimental variables. The variables were used to compute the energy consumption.

We compared the proposed composition algorithm against a baseline approach [8], Dijkstra’s algorithm [154], and Floyd-Warshall’s algorithm [155]. The baseline approach considers all aspects of composition without in-flight energy-sharing. Formations, intrinsic constraints, and extrinsic constraints are all considered in the baseline approach. Hence, we evaluated across the two levels. The blue parts of Fig. 6.4 represent the baseline approach. For Dijkstra’s and Floyd-Warshall’s methods, we considered the cost of every edge to be the travel time tt of the edge and the node time nt (charging + waiting times). The proposed framework was evaluated under the two different positioning settings of the support drones. The first setting positioned the support drones at the worst positions (location-aware). The second setting positioned the support drones in the best positions (energy-aware). In addition, the proposed framework is evaluated using the different energy-sharing methods proposed, i.e., Priority-based (PB) and Fairness-based (FB) sharing.

6.6.2 Effectiveness

The first experiment compared the number of successful requests between the proposed settings. A successful request is a request that was able to be served by the swarm without getting stuck in intermediate nodes due to a lack of power. We compared the four different combinations of settings against the Baseline, Dijkstra's, and Floyd-Warshall's algorithms. This included the initial positioning and energy-sharing approaches. As shown in Fig. 6.8, the trend shows that *positioning* the support drones in the optimal position (energy-aware) generally results in a lower number of successful requests (18,050 successful requests out of 20,000 compared to 19,271 in location-aware setting). This is because positioning the support drones in the optimal position meant the rest of the delivery drones would be shifted to the worst positions. The number of delivery drones is usually higher than the support drones, and hence more energy consumption occurs during travel. Although the support drone consumed the least energy in this setting and could share more of its excess energy with the delivery drones, the number of energy requests generated by the delivery drones was more than what the support drones could serve at a time. Hence, more failures occurred. Last, we can note that the PB energy-sharing approach performed better in all settings. All requests were successfully delivered compared to the Fairness-based energy-sharing. This finding is explained in the second experiment. Compared to the baseline, Dijkstra's, and Floyd-Warshall, our proposed energy-sharing approach performed much better. In baseline, only 8,839 requests were successful out of 10,000, which was less than any of the four combinations. This is because the swarm in the baseline approach depended solely on the delivery drones' batteries and, in many cases, were stuck at a node due to the battery limitations. Floyd-Warshall's performed worse than the baseline but slightly better than Dijkstra's (6,691 successful out of 10,000). Dijkstra's performed worse regarding successful requests (6,646 successful out of 10,000) because our modified A* sequential composition algorithm could detour its path if the swarm was stuck at a node. In contrast, Dijkstra's and Floyd-Warshall's path composition was solely based

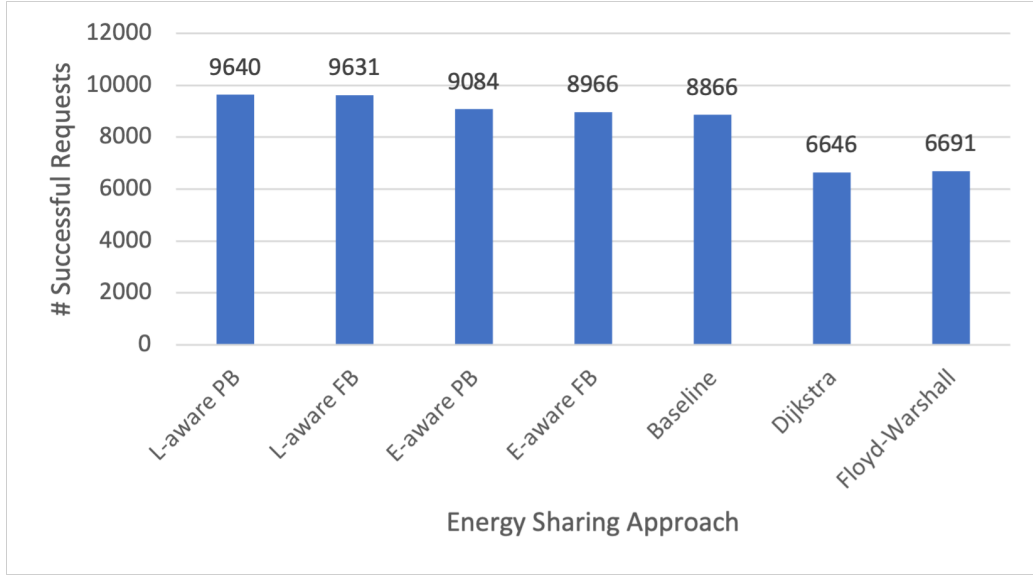


FIGURE 6.8. Number of successful requests using in-flight energy-sharing

on the cost of each segment only. Hence, the composed path may have had segments that a swarm could not travel in due to its battery limitations.

In the second experiment, we focused on studying the behavior of the two proposed energy-sharing methods, i.e., Priority-based (PB) and Fairness-based (FB). Fig. 6.9(A) and Fig. 6.9(B) present the delivery times of the successful requests in PB and FB sharing under the different positioning settings. The x-axis presents the distance in kilometers, and the y-axis is the delivery time in hours. The successful requests are grouped by the distance between the source and the destination at 0.5 km intervals. The average delivery time of requests in every interval is presented on the graphs. The figures show that PB sharing is steadily performing the best in different settings. This is because the energy-sharing requests by the delivery drone are generated dynamically after every served request (see Algorithm 6.2), ensuring the proactive, dynamic generation of energy requests. Hence, most needy drones are served first, ensuring better energy distribution. Better distribution of energy means less time spent at intermediate nodes to charge. It may also mean the ability to skip nodes if the swarm can reach the destination directly without stopping. Hence, a better delivery time. The delivery drones are served equally at every i interval with FB. This means that there is

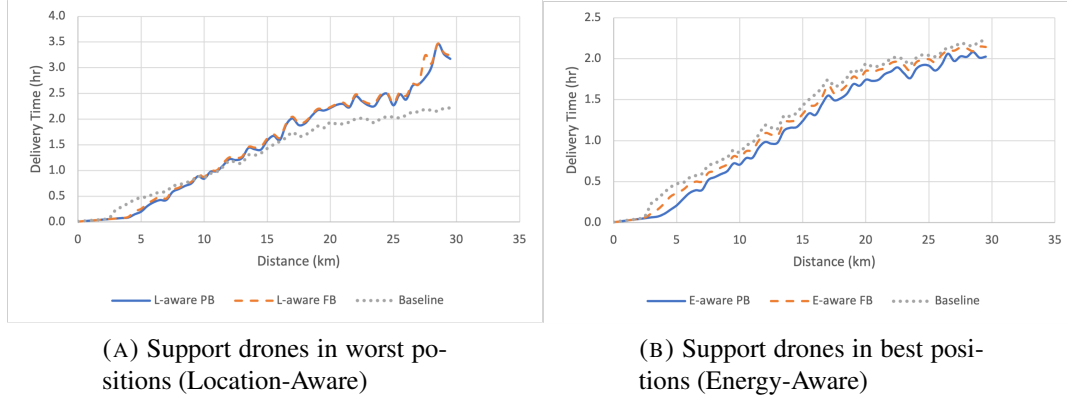


FIGURE 6.9. Delivery Times of Various Energy Sharing Methods

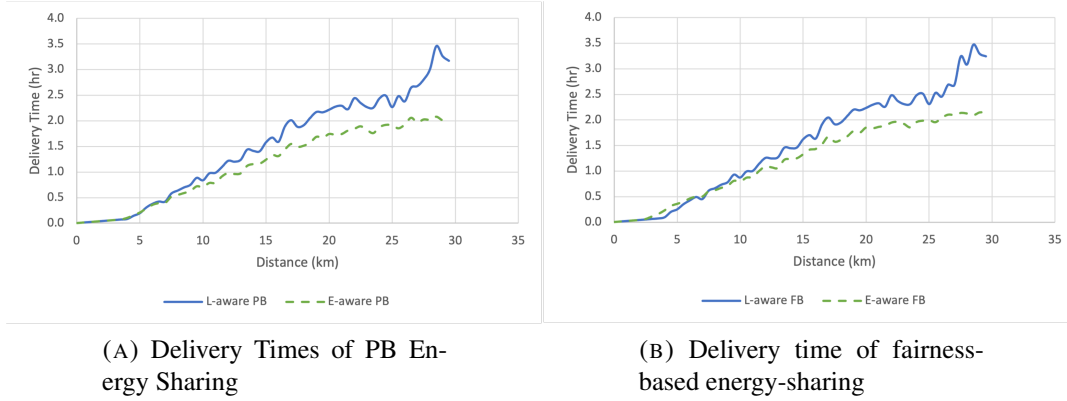


FIGURE 6.10. Delivery times under different positioning settings

no priority in charging. Most needy delivery drones may be served later or might not get the chance to be charged if the skyway segment is short. As explained earlier, the energy-sharing process is constrained by the travel time of an edge. The better performance of PB sharing is more visible in the energy-aware setting where the support drones are positioned in the optimal positions, and the delivery drones are losing more energy due to their positions. This means that delivery drones generate more energy requests in this positioning approach, and the difference between the energy-sharing methods is more evident.

In the third experiment, we studied the initial positioning settings and their effect on the delivery times. As shown in Fig. 6.9(A), positioning the support drones in the worst position (location-aware) results in worse delivery times than the baseline at

longer distances. However, it had the most number of successful requests (Fig. 6.8). All the drones charged fully when the swarm landed on a recharging station. If the support drones were in the worst positions, they consumed the most energy (in addition to given energy in the sharing process). Since the support drones had bigger battery capacities than delivery drones, the time taken to recharge fully was longer when landing as the rest of the swarm was waiting for it. This is especially significant at longer distances, as shown in Fig. 6.9(A). This behavior is reflected in Fig. 6.11(A), which represents the nodes' time with the location-aware setting. The time spent at the node was longer than the baseline due to the charging of the support drones, which were placed in the worst position consuming the most amount of energy. With longer distances, the support drone consumed more energy and shared more energy, hence, spent more time charging. Whereas in an energy-aware setting, where the support drone was positioned optimally, it took less time to recharge at a recharging station. The delivery times were, therefore, better than the location-aware setting. This difference is visible in Fig. 6.10(A) and 6.10(B) with both PB and FB energy-sharing. Hence, we conclude that, based on the goal of the application, the choice of the setting in positioning is determined. If the goal is to serve as many requests as possible, the support drones would be placed in the worst positions. Otherwise, if the goal is to serve requests as quickly as possible, the support drones would be placed in the best positions. For example, in emergency cases, the delivery time would be the most critical objective. If the delivery is successful, then the energy-aware setting should be adopted. Otherwise, the location-aware setting would be adopted to ensure the success of the delivery. Moreover, the in-flight energy-sharing composition outperformed the baseline under the best positioning setting (Fig. 6.9(B)). This reflects the usefulness of implementing in-flight energy-sharing to overcome a swarm's battery limitation and increase its flight range. Figs. 6.11(A) and 6.11(B) represent the node times of the delivery requests without the travel times. The figures prove that the improvement in delivery times is due to the reduction of node times because of in-flight recharging. The behavior of the lines is similar to Fig. 6.9(A) and Fig. 6.9(B) but with shorter times, demonstrating the reduction in node times mainly.

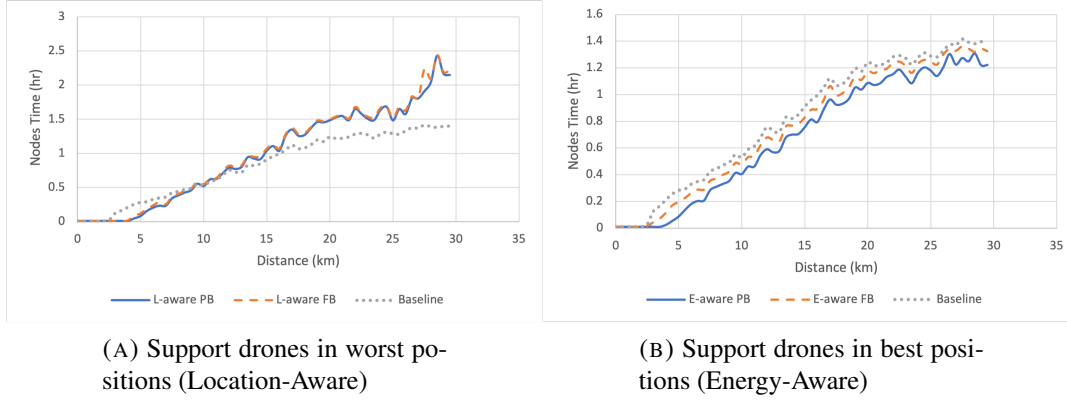


FIGURE 6.11. Nodes time of various energy-sharing methods

6.6.3 Runtime Efficiency

We measured the execution times of the proposed methods. Fig. 6.12(A) and Fig. 6.12(B) depict the execution times of different positioning and energy-sharing approaches. The graphs show that the positioning did not affect the execution times. However, in both positioning settings, the execution time of the PB energy-sharing method outperformed the FB energy-sharing methods. This is because PB sharing only gives energy on request. The FB is pre-emptive and keeps charging the drone iteratively by switching between them, increasing the algorithm's execution time. In addition, with FB energy-sharing, the shared amount per round is parameterized. This means the user inputs the amount of shared energy per round. The smaller the amount is, the more rounds there are, which would make the execution time longer. We assume the support drones share 2,240 mA per round for this experiment. If the amount is smaller, less time would be needed to share the energy and, therefore, more rounds in the FB execution. The baseline's execution time is less than any proposed method because it has no in-flight energy-sharing processes.

6.7 Conclusion

This chapter proposed a novel Swarm-based Drone-as-a-Service framework for delivery with in-flight energy-sharing. We formulated the composition problem as a single

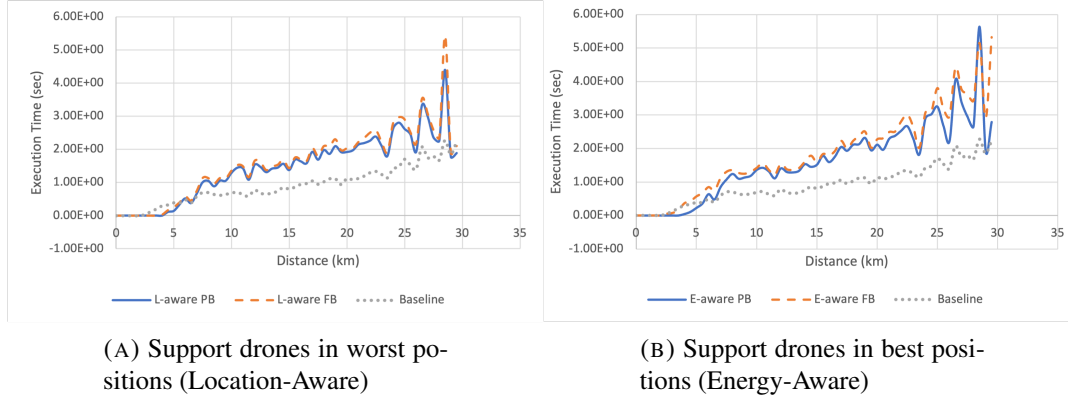


FIGURE 6.12. Execution time of various energy-sharing methods

objective to optimize a consumer's request delivery time. We then introduced the use of support drones that share their extra available energy with the delivery drones. We proposed a redundancy-based method to estimate the number of support drones and reduce the failure probability. Two settings are presented to initially position the support drones in a flight formation. Two novel energy-sharing methods are proposed, which are based on FCFS and Round Robin approaches. A formation reordering method was presented to facilitate the energy-sharing process. A modified A* algorithm was implemented to compose the best and optimal services from the source to the destination. Experimental results showed the efficiency of our proposed approaches compared to Dijkstra's, Floyd-Warshall's, and baseline approaches. In future work, we will expand this problem to a multi-objective problem to include the cost and overall energy utilization.

Flow-based Energy Service Composition

7.1 Introduction

Energy providers' and consumers' *dynamic* movement may impact the *balance* between energy services and requests. Energy service providers and consumers may have different preferences regarding location, time, and energy. Therefore, the available energy may be *wasted* by being imprudently allocated to requests. For instance, a provider may elect to offer a large amount of energy to a small request—the match between a large service and a small request wastes unused energy. Typically, in crowdsourced environments, the number of service providers is less than the number of consumers [54]. Accordingly, it is crucial to *efficiently* allocate the available scarce energy services [14]. For example, provisioning an existing equal-in-size request efficiently utilizes a large service. An additional challenge is the likelihood of a single energy service's inadequacy to fulfill the requirement of a consumer due to the limited resources of wearables [14]. In such cases, multiple services may be composed to accommodate a single energy request. Consequently, *services composition* is key to efficiently leveraging the available energy in crowdsourced ecosystems.

Existing energy service composition frameworks mainly focus on accommodating a *single* request [14][24] or catering a *single* service as presented in Chapter 3 and Chapter 4. These techniques do not consider leveraging *multiple* services to fulfill *multiple* requests simultaneously [24][14]. This may impede unlocking the full potential of crowdsourcing energy services. This chapter focuses on *efficiently matching and composing multiple* energy services to accommodate *multiple* requests simultaneously.

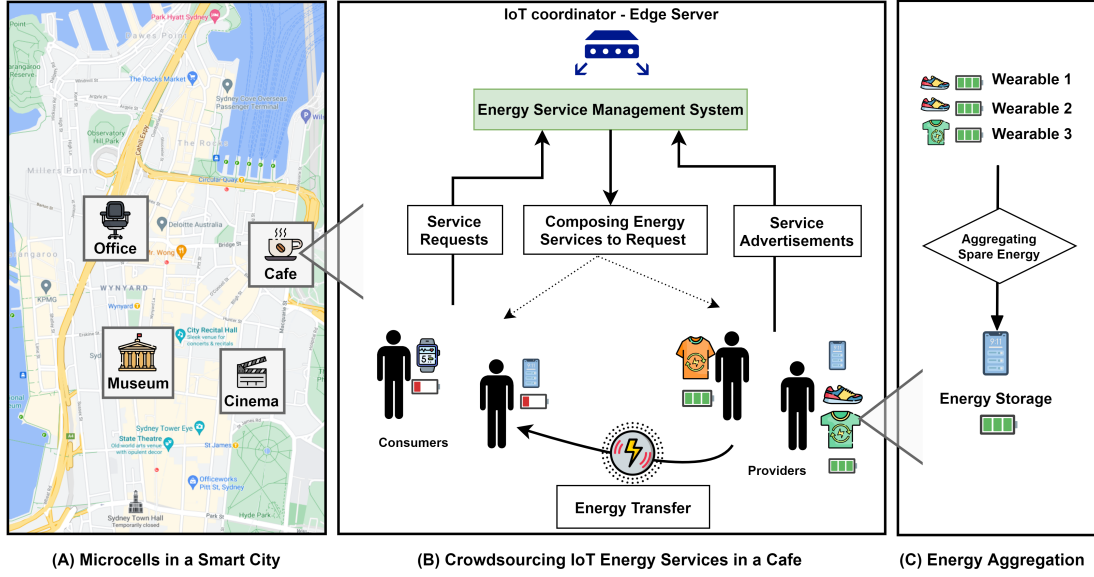


FIGURE 7.1. Crowdsourcing IoT energy services scenario

We propose a novel spatio-temporal framework to compose *multiple* energy services and fulfill the requirements of *multiple* energy requests within a specified time interval in a confined area. The framework aims to *maximize the utilization of the available energy*. Indeed, in a crowdsourced IoT environment, fully utilizing the available energy services is not guaranteed due to the incongruity of the spatio-temporal preferences of energy providers and consumers. Therefore, optimizing the available energy services among the existing requests is crucial. The framework consists of two levels of composition. The first level allows providers with multiple wearables to aggregate and store their services in a single device (see Fig. 7.1(C)). This device is used as a proxy to offer energy to other IoT users securely [157]. The second level is the composition approach to match *multiple* energy services to *multiple* energy requests. We propose *EnergyFlowComp*, a composition approach that utilizes the unique *fragmented* nature of energy to share *partial* service with multiple requests to maximize the utilization of energy [23]. Unlike other services, which are atomic (i.e., either fully served or not served at all), energy services may be divided and shared partially. For instance, a large energy service may be split to fulfill two smaller requests. Additionally, we propose *PartialFlowComp*, which considers the *partial-temporal overlap* between services and requests in provisioning. The main contributions of this chapter are:

- A novel Aggregated energy service model.
- A two-level composition of *multiple* energy services to fulfill *multiple* requests.
- *EnergyFlowComp* a QoS-based *network-flow* approach to match energy services to requests.
- *PartialFlowComp* a heuristic-based approach to efficiently allocate energy services to requests.
- A set of extensive experiments to show the efficiency and effectiveness of the proposed framework.

The chapter's remainder is organized as follows. Section 7.2 presents the motivation scenario. Section 7.3 provides an overview of the system model and defines the problem. Section 7.4 presents the details of the proposed energy composition framework. Section 7.5 evaluates the approach and presents the results. Section 7.6 concludes the chapter and highlights future work.

7.2 Motivating Scenario

We consider the following motivating scenario to highlight the significance of our work. Let us consider a mall as the general geographical area for crowdsourcing energy services. We divide this area into microcells, where each cell is a confined space (e.g., coffee shop, restaurant, movie theater, etc.) (see Fig. 7.1(A)). Each microcell contains IoT devices that provide or receive energy (see Fig. 7.1(B)). The devices are assumed to have wireless energy transmitters and receivers such as Energous¹. All energy requests and advertisements are submitted and processed at the *edge* (e.g., a router associated with a microcell). The IoT *coordinator* at the edge manages and matches the energy services and requests to utilize the available energy efficiently. Additionally, the IoT coordinator is responsible for providing rewards to encourage providers to offer services.

¹<https://www.energous.com/>

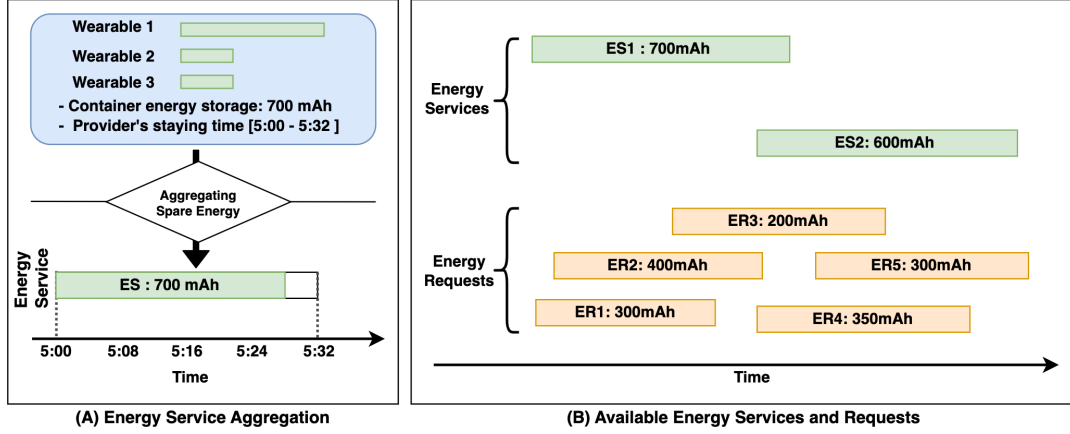


FIGURE 7.2. Crowdsourcing wireless energy example: (A) Phase 1: The proxy aggregates its wearables' spare energy as an energy service (B) Phase 2: The IoT coordinator composes services to requests.

We assume *IoT users* are willing participants to provide energy to nearby IoT devices as energy providers. Each provider has at least one wearable that harvests energy (e.g., smart shoe) or has spare energy (e.g., smartphone). Energy providers use one of their devices as a *proxy* to store the spare energy (see Fig. 7.1(C)). The proxy is assumed to have computing and energy storage capabilities, i.e., a smartphone. The proxy uses its extra energy storage (i.e., battery) as a container to aggregate and store the spare energy from the provider's wearables. The proxy uses aggregated energy to define its energy service. We assume the proxy's container is big enough to store all the wearables' spare energy.

The process of crowdsourcing energy services consists of two phases: in the first phase, each provider aggregates their wearables' spare energy into a single energy service. For example, provider 1 would collect and aggregate 700 mAh of energy in its energy storage from its three wearables, e.g., two smart shoes and a smart shirt. (see Fig. 7.2(A)). In the second phase, the IoT coordinator receives energy services advertisements from the providers and energy requests from the consumers (see Fig. 7.2(B)). We assume that consumers and providers have different requirements in terms of energy, time availability, and location. The IoT coordinator, then, *coordinates and allocates* the received energy services to requests. *We focus on optimizing the two phases of the crowdsourcing energy services to maximize the utilization of the available energy.*

Fig. 7.2 represents a detailed scenario of seven IoT users staying at a coffee shop. Two users are assumed to be willing to provide their energy services, ES1 and ES2. The other five IoT users request energy from their neighboring IoT devices ER1, ER2, ER3, ER4, and ER5. The advertisement of services and requests includes various information, e.g., start time, end time, location of the IoT device, and provided or requested energy amount. Fig. 7.2(B) illustrates the available services and requests by their timelines. It also shows the amount of available energy. It is challenging to efficiently leverage the available energy and fulfill multiple requests' requirements. Traditional scheduling strategies, such as FCFS and priority-based scheduler, may not be a good fit to provision crowdsourced IoT energy [24]. These strategies fulfill the requirements of each energy request independently and sequentially based on their arrival time. As a result, services may satisfy the requirements of an energy request without being fully utilized, which affects the energy allocation efficiency. For example, assigning ES1 (700 mAh) to ER1 (300 mAh) will use 42.9% from ES1. Alternatively, assigning ES1 to ER2 (400 mAh) will utilize 57.1% from ES1.

The limited provided energy and time constraints of services and requests represent critical challenges for the efficient and maximum provisioning of IoT energy services. We reformulate our service provisioning problem as *a time-constrained optimization problem, i.e., maximizing the allocated energy provided by the available services with respect to their time constraints*. Our proposed approach *EnergyFlowComp* transfers the spatio-temporal services and requests to a Bipartite graph [158]. The approach then matches the services to requests using a modified Maximum Flow algorithm [158]. Furthermore, our approach employs the divisibility nature of energy services to maximize energy utilization [23]. In addition, we propose *PartialFlowComp*, an extension of *EnergyFlowComp* that considers the *partial-temporal overlap* between services and requests in provisioning. For example, as ES1 partially overlaps with ER4, it may offer a partial service to part of ER4.

7.3 System Model and Problem Formulation

We formally present the definitions of IoT energy services and requests. We then introduce the concept of energy utilization among energy services based on their allocated energy. This work considers a provisioning framework for stationary services and requests to focus only on the temporal constraints in allocating multiple energy services to multiple requests in a microcell within a predefined time interval. The aim is to ensure maximizing the energy provision over a predefined time.

7.3.1 Energy Service Model

We define an energy provider as an IoT user willing to share their devices' spare energy with nearby IoT devices. We use energy services to abstract the energy-sharing process. An energy service is the abstraction of the wireless energy delivery from an IoT device to another [25]. In the previous energy service model, energy provisioning was from a single energy harvester [23]. We extend the existing energy service model by formally representing the aggregated energy from a set of wearables owned by the IoT user [159]. The goal is to define the aggregated energy amount, time interval automatically, and spatial location to provide the energy service. We formally define the energy service as follows:

Definition 1: Energy service (ES). An ES is defined as a tuple of $\langle EID, PID, W, F, Q \rangle$, where:

- EID is a unique identifier for energy service.
- PID is a unique identifier for the provider.
- W is a set of wearables $\{w_1, \dots, w_n\}$ owned by the provider.
- F is the function of sharing energy wirelessly.
- Q denotes a QoS property of the energy service [25].

Definition 2. Quality attributes of an energy service (QoS). QoS is a set of parameters that allows energy consumers to distinguish among energy services [23], namely:

- ae is the aggregated energy from the provider's wearables they may share.
- l is the spatial location of the provider.
- r is the range for a successful wireless energy transfer.
- $[st, et]$ represents the start and end times of the provider's aggregated energy service.

7.3.2 Energy Request Model

An energy consumer in the crowdsourced IoT environment is an IoT user who requests energy for their devices from nearby IoT devices. An energy request is defined as the abstraction of the energy consumer requirement according to their spatio-temporal preferences [25]. We formally define an energy service request as follows:

Definition 3: Energy Service Request (ER). An ER is defined as a tuple of $\langle RID, CID, re, st, et, l \rangle$, where:

- RID is a unique identifier of energy request.
- CID is a unique identifier of the consumer.
- re is the amount of requested energy by the consumer.
- $[st, et]$ is the duration of the consumer's request.
- l is the spatial location of the consumer.

7.3.3 Energy Utilization and Problem Formulation

The goal of crowdsourcing energy services is to utilize the available energy by efficiently allocating it to the existing energy requests. Typically, the available energy services in a crowdsourced IoT environment are limited [24]. Therefore, allocating the available services efficiently is essential to maintain a self-sustained ecosystem [14]. We formally define energy utilization as follows:

Definition 4: Energy Utilization. The energy utilization of an ES may be defined as the ratio of the allocated energy to the advertised amount of energy. We measure the

Energy Utilization (EU) of a composition using the following:

$$EU(Comp) = \frac{\sum_{i=1}^n ES_i.all}{\sum_{i=1}^n ES_i.ae} \quad \forall ES_i \in \mathcal{ES}, \quad (7.1)$$

if $ES_i.[st, et] \subset T$ and ES_i is serviceable

where $ES_i.all$ is the service's allocated energy to requests, $ES_i.ae$ is the service's available energy, and n is the number of available energy services $\mathcal{ES}$. Note that a service will be considered if (1) its time duration $[st, et]$ falls within the time window of the composition T , and (2) it is serviceable, meaning there are energy requests that can utilize that service.

Definition 5: Energy Services Composition problem. We assume that in a micro-cell there exists a set of n energy services $\mathcal{ES} = \{ES_1, ES_2, \dots, ES_n\}$ and a set of m energy requests $\mathcal{ER} = \{ER_1, ER_2, \dots, ER_n\}$ as shown in Fig. 7.2(B). The $\mathcal{ES}$ will be advertised by providers P . Each ES_i is described using the aforementioned Definitions 1 and 2. The energy requests are submitted by consumers C . Each energy request ER_j is described using the aforementioned Definition 3. We formulate the problem of allocating $\mathcal{ES}$ to $\mathcal{ER}$ as a time-constrained resource matching problem as follows:

- Maximize $EU(Comp)$,

Subject to :

- $\sum re_j \leq ae_i$ for each $ES_i \in \mathcal{ES}$,
- $ER_j.[st, et] \subset ES_i.[st, et]$ for each $ES_i \in \mathcal{ES}$.

The objective function optimizes the available energy services to the existing requests according to their spatio-temporal features.

We use the following assumptions to formulate the problem:

- All IoT energy services and requests are deterministic and stationary, i.e., there is prior knowledge about service availability, QoS values, energy requests, time constraints, and demands [25].
- The IoT coordinator (at the edge) is responsible for *batching* the energy requests and energy services from all consumers and providers in a microcell *over a predefined period*.
- There is no energy loss in sharing, i.e., as the technology of wireless power transfer matures, we anticipate that the devices will be able to share more energy, and the energy loss of sharing will become minimal [24].
- The composition considers the case of multiple providers and multiple consumers.
- Wearables have a fixed energy size during the composition.
- The proxy's energy container is always sufficient to accommodate the wearables' energy.
- The proxy will not use the aggregated energy.
- Providers and consumers may have different time windows.
- A trust framework has been implemented to preserve the integrity and privacy of the participating IoT devices [21].

7.4 Wireless Energy Sharing Framework

In this section, we present the composition framework of crowdsourced energy services. The composition aims to maximize the utilization of the available energy services within a time window by fulfilling nearby requests with respect to their spatio-temporal and energy-related constraints. The framework takes as input the energy services and the energy requests. We assume that consumers define their energy requirements and request duration based on predefined consumption models [23]. In our framework, all the available energy requests and services within a predefined time

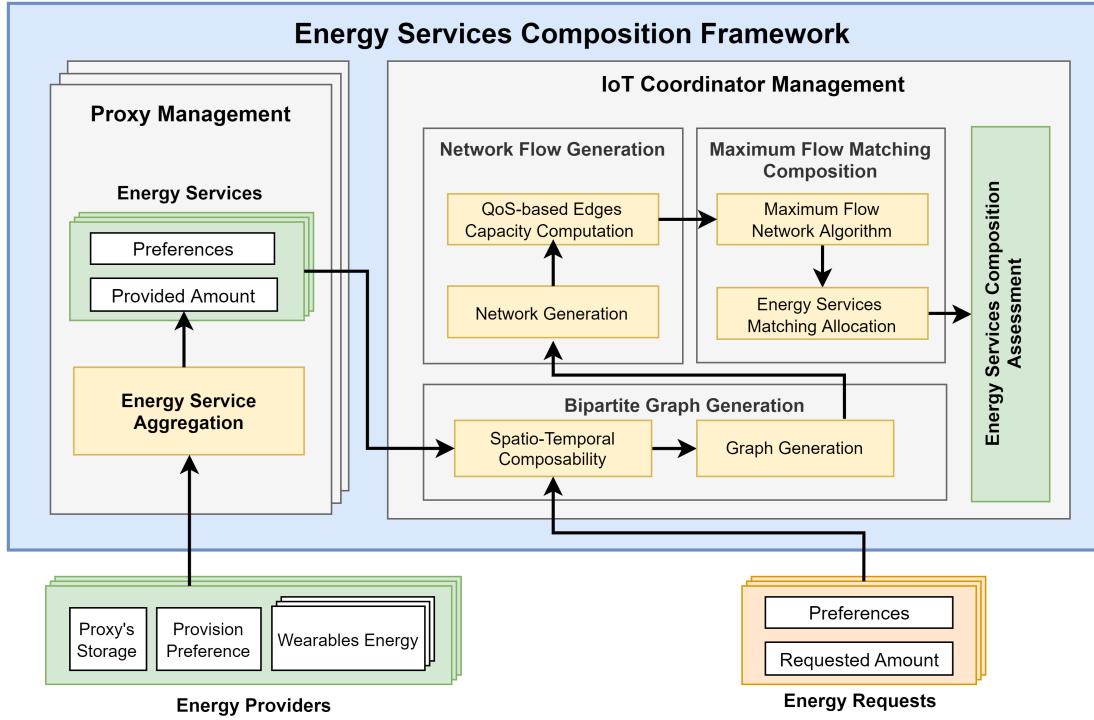


FIGURE 7.3. Wireless energy services framework

frame are batched. This holistic view allows the IoT coordinator to efficiently allocate the available energy among the existing requests with respect to their time constraints. The Energy services composition framework consists of The Proxy Management Phase and the IoT Coordinator Management Phase (see Fig. 7.3). The first phase aggregates the wearables' spare energy of a provider to define their energy service. The second phase matches the available services to requests.

7.4.1 Phase 1: Proxy Management

The first phase takes as an input the amount of available energy by the provider's wearables, the proxy storage size, and the provider provisioning preferences (e.g., start time and duration of provisioning). The first phase in the energy-sharing process involves aggregating the wearables energy to define an energy service (see Fig. 7.2(A)). The aggregation is achieved by transferring the energy from the wearables to the proxy energy storage. We assume the energy storage is enough to contain all the aggregated

Algorithm 7.1 Energy Services Aggregation**Input:** $\mathcal{E}$, $time$ **Output:** ES

//Phase 1: Proxy Management

```

1: for  $e_i$  in  $\mathcal{E}$  do
2:    $energy\_amount \leftarrow energy\_amount + aggregate(e_i)$ 
3:  $time\_preference \leftarrow Define\_Time\_Interval(time)$ 
4:  $location \leftarrow Detect\_Location()$ 
5:  $ES \leftarrow Define\_Energy\_Service(energy\_amount, time\_preference, location)$ 
6: return  $ES$ 

```

energy. Algorithm 7.1 represents the phase in detail. Algorithm 7.1 takes as input the amount of harvested energy by wearables ($\mathcal{E}$) and the temporal preferences of the IoT devices owner ($time$). The output of the aggregation algorithm is an energy service (ES). Lines 1-2 aggregate the wearables' energy amounts by adding them up. Lines 3-4 define the spatio-temporal features of the service from the owner's preferences. Line 5 defines the energy service (ES) using Definitions 1 and 2.

7.4.2 Phase 2: IoT Coordinator Management

The IoT coordinator management phase aims to maximize energy utilization by allocating services to the corresponding requests. This phase takes as input the advertised aggregated energy services and energy requests. Four steps implement this phase in the following order: Bipartite Graph Generation, Network Flow Generation, and Maximum Flow Matching composition and the composition assessment. In the following subsections, we discuss these steps in detail, respectively.

7.4.2.1 Bipartite Graph Generation

The Bipartite Graph Generation step aims to present the spatio-temporal composability between the energy services and requests. This step transfers a given set of energy services $\mathcal{ES}$ and requests $\mathcal{ER}$ to a Bipartite Graph (BG) (see Fig. 7.4). BG is a directed graph represented by a set of (U, V, DE) where U is a set of nodes that represent the available services, V is a set of nodes that represent the available requests, and DE is a set of directed edges from U to V . For example, if two nodes (u_i, v_j) are connected

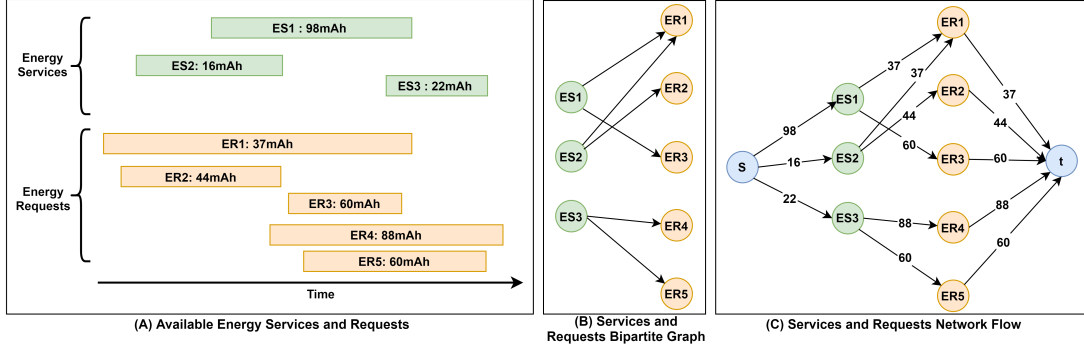


FIGURE 7.4. Bipartite graph generation example

Algorithm 7.2 Services Bipartite Graph Generation**Input:** $\mathcal{ES}, \mathcal{ER}, T$ **Output:** BG

```

1:  $\mathcal{ES} = \text{Temporal\_filter}(\mathcal{ES}, T)$ 
2:  $\mathcal{ER} = \text{Temporal\_filter}(\mathcal{ER}, T)$ 
3:  $BG = \{\}$ 
4: for  $ES_i$  in  $\mathcal{ES}$  do
5:    $BG.\text{Add\_node}(ES_i.EID, ES_i, \text{type} : U)$ 
6: for  $ER_j$  in  $\mathcal{ER}$  do
7:    $BG.\text{Add\_node}(ER_j.RID, ER_j, \text{type} : V)$ 
   // Spatio-Temporal Composability
8: for  $ES_i$  in  $\mathcal{ES}$  do
9:   for  $ER_j$  in  $\mathcal{ER}$  do
10:    if  $\text{Nearby}(ES_i.l, ER_j.l)$  then
11:      if  $ES_i.[st, et] \subseteq ER_j.[st, et] \parallel ER_j.[st, et] \subseteq ES_i.[st, et]$  then
        // Graph Generation
12:         $w = ER_j.r$ 
13:         $BG.\text{Add\_edge}(ES_i.EID, ER_j.RID, w)$ 
return  $BG$ 

```

with an edge ed_k , it indicates that the service ES_i can fulfill the request ER_j . For instance, in Fig. 7.4(A), the energy services ES_2 falls within the energy request ER_1 duration. Therefore, an edge will be added in the graph BG between the node of ES_2 and ER_1 (see Fig. 7.4(B)).

Algorithm 7.2 represents the step of bipartite graph generation in detail. Algorithm 7.2 takes as an input the available energy services $\mathcal{ES}$ and requests $\mathcal{ER}$ and the time window of the composition T . The output of the algorithm is a Bipartite Graph (BG). Lines 1-2 select the services and requests that fall within the time window T . Lines 3-7 add the available services and requests as nodes to the graph. Then for each energy service and request, the algorithm checks if the service and request are spatially

composable (Lines 8-11). A service and a request are considered spatially composable if, and only if, the distance between them is less than the required distance for a successful energy wireless transfer [23]. Additionally, the algorithm checks the temporal composability between each service and a request (Line 11). We extend the definition of temporal composability by [23] to consider a service and a request are temporally composable if the duration of the service falls within the duration of the request or vice versa [23]. If a service and a request are spatio-temporally composable, an edge will be added to the graph (Lines 12-13). The edge connects the service node to the requesting node, indicating that the service may fulfill the energy request. Line 12 computes the edge's weight w , which is equal to the required energy by request. Line 13 adds the edge between the service's node and the request's node in the bipartite graph BG . The added edge indicates the service node may fulfill the energy request up to its w . The output of the algorithm is a bipartite graph. The graph represents the spatio-temporal composability between the energy services and requests.

7.4.2.2 Network Flow Generation

Traditionally, network flow has been used to solve bipartite graph matching [160]. In this step, we aim to generate a network flow from the bipartite graph resulting from Algorithm 7.2 (see Fig. 7.4(C)). The network flow is generated with respect to the quality of services (QoS) of both services and requests. The network flow generation step consists of two main components: (1) Network Generation and (2) QoS-based edges' capacity computation. The first step of network flow generation converts the bipartite graph to a network by connecting all the nodes to a source and a sink. The source and sink are conceptual nodes that facilitate the construction of the network. They do not represent actual entities but are rather used to define the start and end points for the flow within the network. The source acts as the aggregate origin for all inputs into the system, while the sink serves as the collective exit point for all outputs. Their purpose is to provide a framework within which the algorithm can operate to simulate the flow of services from providers to consumers. The second step of network flow generation uses the QoS attributes to calculate the capacity of each edge.

Algorithm 7.3 represents the services network flow generation in detail. Algorithm 7.3 takes as an input the generated bipartite graph BG , available energy services $\mathcal{ES}$ and requests $\mathcal{ER}$. The output of the algorithm is a network flow NF . Lines 1-3 initialize the network and add a source s and a sink t . Lines 4-13 traverse every node in the bipartite graph BG to add it to the network (Line 5). Additionally, if the node's type is 'U' (i.e., the node represents a service), then the node will be connected to the source s (Lines 6-9). In addition, the capacity of the edge will be the amount of energy that may be provided by the service (Line 8). Moreover, if the node's type is 'V' (i.e., the node represents a request), then the node will be connected to the sink t (Lines 10-13). In addition, the edge's capacity will be the amount of energy that may be inquired by request (Line 12). Lastly, the algorithm retrieves the neighbor nodes of every 'U' node in BG to add an edge between them in the network flow NF . Recall that the 'U' node is a service whose neighbors are requests (Lines 14-16). Line 18 computes the edge's capacity *capacity*, which is equal to the edge's weight w in the bipartite graph BG . Recall that w is equal to the required amount of energy by the request. Line 19 adds the edge between the node and its neighbor to the network flow NF . The added edge indicates that the service node may fulfill the energy request up to its *capacity*. The algorithm's output is a network flow representing the energy flow from services to requests with respect to their QoS attributes.

7.4.2.3 Maximum Flow Matching Composition

In this step, we propose a Maximum-Flow matching Composition (*EnergyFlowComp*). Our approach allocates energy services to requests using a modified maximum flow algorithm. The Maximum-Flow matching Composition *EnergyFlowComp* consists of two components: (1) Maximum flow network algorithm and (2) Energy services matching allocation (see Fig. 7.3). The first component of the composition uses a maximum flow algorithm to find a maximum single-commodity flow. For instance, using the previous example of Fig. 7.4, the result of the first component in the Maximum flow matching Composition is shown in red in Fig. 7.5(B). The arrows indicate the path to get the maximum flow in the network. In our context, the maximum flow

Algorithm 7.3 Services Network Flow Generation**Input:** $BG, \mathcal{ES}, \mathcal{ER}$ **Output:** NF

```

1:  $NF = \{\}$ 
   // Add the source node of the network
2:  $NF.Add\_node(s)$ 
   // Add the sink node of the network
3:  $NF.Add\_node(t)$ 
4: for  $node_i$  in  $BG$  do
5:    $NF.Add\_node(node_i)$ 
6:   if  $node_i.type == 'U'$  then
7:      $es = Get\_service(\mathcal{ES}, node_i.EID)$ 
8:      $capacity = es.ae$ 
9:      $NF.Add\_edge(s, node_i, capacity)$ 
10:  if  $node_i.type == 'V'$  then
11:     $er = Get\_request(\mathcal{ER}, node_i.RID)$ 
12:     $capacity = er.re$ 
13:     $NF.Add\_edge(node_i, t, capacity)$ 
14: for  $node_i$  in  $BG$  do
15:   if  $node_i.type == 'U'$  then
16:      $neighbours = Get\_connected\_nodes(node_i)$ 
17:     for  $n_j$  in  $neighbours$  do
18:        $capacity = n_j.w$ 
19:        $NF.Add\_edge(node_i, n_j, capacity)$ 
return  $NF$ 

```

will be the maximum energy services that can be allocated to the existing requests. Note that a selected edge between two nodes in the network flow does not mean the full use of its capacity. For instance, in Fig. 7.5(B), the capacity of the edge between ES2 and ER2 is 44. However, the capacity of the edge between the source s and ES1 is 16, which means the available energy by ES1 is 16. Hence, the allocated energy is 16, as shown in the edge between ES2 and ER2 in Fig. 7.5(C). The second component of the composition traverses the resulting flow to assign services to requests (see Fig. 7.5(C)).

The *EnergyFlowComp* approach is described in detail in Algorithm 7.4. Algorithm 7.4 takes as an input the providers' provisioning preferences $\mathcal{P}$, available energy requests $\mathcal{ER}$, and the composition's time window T . The algorithm's output is a composition of services allocated to the requests $Comp$. Lines 1-4 define the available energy services $\mathcal{ES}$ based on the available providers $\mathcal{P}$. For each provider P_i , our approach computes its services using the energy services aggregation algorithm described in

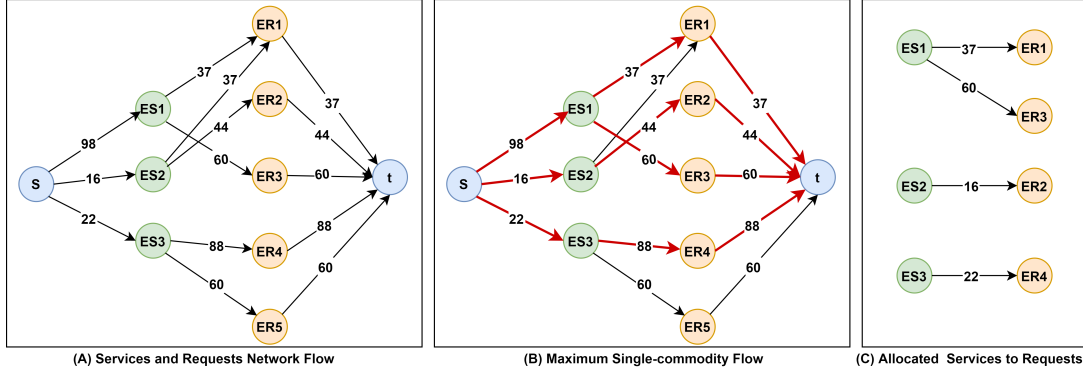


FIGURE 7.5. Maximum flow matching composition

Section 7.4.1. Line 5 generates the bipartite graph BG from the available energy services $\mathcal{ES}$, requests $\mathcal{ER}$, and the time window of the composition T . The bipartite graph BG will be used to match services to requests. The bipartite graph generation is presented in Section 7.4.2.1. Line 6 generates the network flow NF from the generated bipartite graph BG , available energy services $\mathcal{ES}$, and requests $\mathcal{ER}$. The network flow NF will be used to run a maximum flow algorithm to get the best match of services to requests. The network flow generation is presented in Section 7.4.2.2. Line 7 uses a maximum flow algorithm to find a maximum single-commodity flow. There are several state-of-the-art algorithms to compute the maximum flow of a network, including Preflow–Push, Edmonds–Karp, and Boykov–Kolmogorov [160]. All these algorithms result in the same output but differ in time complexity. Therefore, we used the Preflow–Push algorithm, as it is considered one of the most efficient maximum flow algorithms [160]. Line 8 traverses the resulting flow to assign services to requests. The algorithm’s output is a composition that matches services to requests with respect to their QoS attributes.

7.4.2.4 Energy Services Composition Assessment

The last step in the framework is the energy services composition assessment. In what follows, we define the metrics to assess the efficiency of our proposed framework.

Algorithm 7.4 Energy Services Matching Composition**Input:** $\mathcal{P}, \mathcal{ER}, T$ **Output:** $Comp$ **// Energy Services Aggregation**1: $\mathcal{ES} = \{\}$ 2: **for** P_i **in** $\mathcal{P}$ **do**3: $ES = \text{Energy_Services_Aggregation}(P_i)$ 4: $\mathcal{ES}.\text{append}(ES)$ **// Bipartite Graph Generation**5: $BG = \text{Services_Bipartite_Graph_Generation}(\mathcal{ES}, \mathcal{ER}, T)$ **// Network Flow Generation**6: $NF = \text{Services_Network_Flow_Generation}(BG, \mathcal{ES}, \mathcal{ER})$ **// Energy Services Matching Allocation**7: $flow = \text{Maximum_Flow_Network}(NF)$ 8: $Comp = \text{Energy_Services_Allocation}(flow, \mathcal{ES}, \mathcal{ER})$ **return** $Comp$ **Energy Utilization**

Recall that our composition's objective is to maximize available energy utilization. Typically, the available energy services in a crowdsourced IoT environment are limited [24]. Efficiently allocating the available services is essential to maintaining a self-sustained ecosystem [14]. We measure the Energy Utilization (EU) using Eq. 7.1.

Fulfillment Rate

We define the *Fulfillment Rate* (FR) as the percentage of the required energy that has been fulfilled. Intuitively, better energy allocation and utilization may result in a better fulfillment rate. Therefore, a higher fulfillment rate indicates efficiency in fulfilling the demanded energy. The *Fulfillment Rate* FR of a composition may be computed as follows:

$$FR(Comp) = \frac{\sum_{i=1}^n ES_i.al}{\sum_{j=1}^m ER_j.re} \quad \forall ER_j \in \mathcal{ER}, \quad (7.2)$$

if $ES_i.[st, et] \subset T$ **and** ES_i **is serviceable,**

and if $ER_j.[st, et] \subset T$ **and** ER_j **is serviceable**

where $ES_i.all$ is the service's allocated energy to requests, n is the number of available energy services $\mathcal{ES}$, $ER_j.re$ is the required energy by request, and m is the number of available energy requests $\mathcal{ER}$. Note that a service and, similarly, a request will be considered if (1) its time duration $[st, et]$ falls within the time window of the composition T , and (2) it is serviceable, meaning there are energy requests or services that can utilize or fulfill that service.

Consumers and Providers' Satisfaction

Maximizing the energy allocation of the available energy service may *satisfy* more consumers by fulfilling their requests. In addition, it may *satisfy* more providers by using their services, thereby getting more rewards [14]. Typically, consumers may not be satisfied with an efficient allocation if the available resources are limited [161]. However, the efficient allocation of energy services in a crowdsourced IoT environment is claimed to increase consumers' satisfaction [24]. We assess the efficiency of our framework by measuring the satisfaction of consumers. The satisfaction of an energy consumer j is gauged by their perception of the allocated energy $ES_i.all$ to their request $ER_j.re$. Therefore, the used equation for computing the consumer satisfaction CS_i is presented as follows:

$$CS_i = \begin{cases} \frac{ES_i.all}{ER_j.re \times Sf}, & \text{if } ES_i.all \leq ER_j.re \\ 1, & \text{otherwise} \end{cases} \quad (7.3)$$

where $ES_i.all$ is the service's allocated energy to the request, $ER_j.re$ is the required energy by request, and Sf is the satisfaction threshold defined by the total available services [24]. Due to the limited resources in the crowdsourced IoT energy environment, we consider consumers satisfied if they received $Sf\%$ of what they requested [24].

Similarly, we gauge the providers' satisfaction by their perception of the utilized energy $ES_i.all$ from their available services $ES_i.ae$. The higher the utilization of their services, the higher their rewards, as discussed in Chapter 3. Hence, we also assess

our framework's efficiency by measuring the providers' satisfaction. The satisfaction of an energy provider PS_i is computed as follows:

$$PS_i = \begin{cases} \frac{ES_i.all}{ES_i.ae \times Sf}, & \text{if } ES_i.all \leq ES_i.ae \\ 1, & \text{otherwise} \end{cases} \quad (7.4)$$

where $ES_i.all$ is the service's allocated energy to fulfill requests, $ES_i.ae$ is the size of the energy service, and Sf is the satisfaction threshold defined by the total available services [24]. Due to the temporal constraints of services and requests and the limited resources in the crowdsourced IoT energy environment, we consider providers satisfied if they offered $Sf\%$ of their services [24].

Global Consumers and Providers' Satisfaction

We adopt the metric proposed by [24] to define a global satisfaction metric for all providers and consumers. The global consumers' satisfaction is computed by estimating the overall proximity score among the satisfaction scores of all consumers [162][24]. An efficient services composition is reflected by the *closeness* of the satisfaction scores among consumers (i.e., consumers have very similar satisfaction scores). In contrast, the *sparsity* of consumers' satisfaction scores indicate an inefficient service composition. Therefore, similar to [24], the global satisfaction of consumers CS may be captured using the *information entropy* [163]. The information entropy measures the disorder degree of all requests $ER_j \in \mathcal{ER}$ based on their satisfaction score CS_j as follows:

$$H(CS) = - \sum_{j=1}^m CS_j \log_2 CS_j \quad (7.5)$$

where m is the number of available energy requests $\mathcal{ER}$ and CS_j is a consumer j 's satisfaction computed using Eq. 7.3.

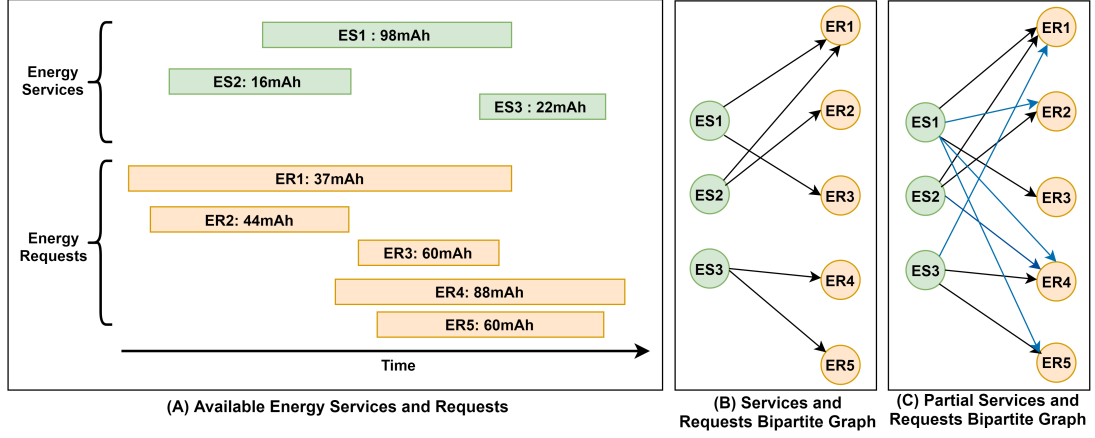


FIGURE 7.6. Example of a bipartite graph and partial bipartite graph

Similarly, the global satisfaction of providers $\mathcal{PS}$ may be captured using the *information entropy* [163]. The information entropy measures the disorder degree of all services $ES_i \in \mathcal{ES}$ based on the providers' satisfaction score PS_i as follows:

$$H(\mathcal{PS}) = - \sum_{i=1}^n PS_i \log_2 PS_i \quad (7.6)$$

where n is the number of available energy services $\mathcal{ES}$ and PS_i is a provider's i 's satisfaction computed using Eq. 7.4.

7.4.2.5 Partial Bipartite Graph Generation

In this step, we propose a *Partial-based* Bipartite Graph Generation (*PartialGraph*). The *PartialGraph* extends our proposed bipartite graph generation approach to include the *partially overlapping* services and requests. Considering the partial overlap between services and requests increases the utilization of the available energy. For example, in Fig. 7.6(A) $ES1$ and $ER2$ are partially overlapping. Using the bipartite graph generation approach described in Algorithm 7.2 will not connect $ES1$ and $ER2$ (see Fig. 7.6(B)). However, using the partial bipartite graph generation approach described in Algorithm 7.5, $ES1$ and $ER2$ will be connected (see Fig. 7.6(C)). The blue arrows in Fig. 7.6(C) represent all additional edges between services and requests. The additional edges will result in a more connected network flow (see Fig. 7.7).

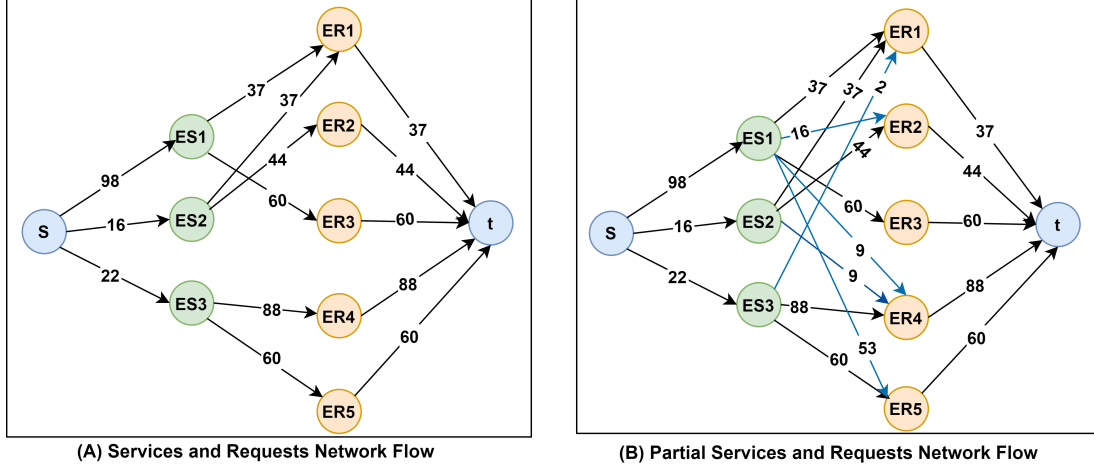


FIGURE 7.7. Example of network flow and partial network flow

Algorithm 7.5 Partial Services Bipartite Graph Generation**Input:** $\mathcal{ES}$, $\mathcal{ER}$, T **Output:** PBG **// Bipartite Graph Generation**

```

1:  $PBG = \text{Services\_Bipartite\_Graph\_Generation}(\mathcal{ES}, \mathcal{ER}, T)$ 
2: for  $ES_i$  in  $\mathcal{ES}$  do
3:   for  $ER_j$  in  $\mathcal{ER}$  do
4:     if  $\text{Nearby}(ES_i.l, ER_j.l)$  then
       // Partial-Temporal Composability
5:       if  $ES_i.[st, et] \cap ER_j.[st, et]$  then
         //Partial Graph Generation
6:          $d = \text{Time\_overlap}(ES_i.[st, et], ER_j.[st, et])$ 
7:          $partialR = d * ER_j.r / ER_j.[st, et]$ 
8:          $PBG.\text{Add\_edge}(ES_i.EID, ER_j.RID, partialR)$ 
return  $PBG$ 

```

Algorithm 7.5 represents the *PartialGraph* in detail. Algorithm 7.5 takes as an input the available energy services $\mathcal{ES}$ and requests $\mathcal{ER}$ and the time window of the composition T . The algorithm's output is a partial bipartite graph (PBG). Line 1 generates the bipartite graph PBG from the available energy services $\mathcal{ES}$, requests $\mathcal{ER}$, and the time window of the composition T . The bipartite graph generation is presented in Section 7.4.2.1. Lines 2-8 extend the bipartite graph generation to include partial services. For each energy service and request, the algorithm checks if the service and request are spatially composable (Lines 2-4). A service and a request are considered spatially composable if, and only if, the distance between them is less than the required distance for a successful energy wireless transfer [23]. Additionally, the algorithm checks the

partial temporal overlap between each service and a request (Line 5). If a service and a request are partially overlapping in time, then an edge will be added to the graph (Lines 5-8). Similar to the bipartite graph approach, the edge connects the service's node to the request's node, indicating that the service may fulfill the energy request. Line 6 computes the duration of overlap between the service and the request. Line 7 computes the edge's weight, which is equal to the part of the request that overlaps with the service, i.e., the required amount of energy by request within the overlap duration. Line 8 adds the edge between the service node and the requesting node in the bipartite graph *PBG*. The added edge indicates that the service node may fulfill the energy request up to its *partialR*. The algorithm's output is a bipartite graph representing the spatio-temporal composability between the energy services and requests.

7.4.2.6 Partial Maximum Flow Matching Composition

In this step, we propose *PartialFlowComp*, an extension of the Maximum flow matching composition proposed in Section 7.4.2.3. *PartialFlowComp* uses the graph generated by *PartialGraph* approach *PBG* (see Section 7.4.2.5). *PBG* will be used instead of the bipartite graph *BG* in Algorithm 7.4 by changing Line 5 to call the function of Algorithm 7.5.

7.5 Experiments Results

We compared our proposed composition techniques (*EnergyFlowComp*) and (*PartialFlowComp*) with two resource matching algorithms, namely, first come first served algorithm (*Baseline-Matching*) and Priority-based algorithm (*Priority-Matching*) [95]. In *Baseline-Matching*, the requests nodes in the graph were matched to services based on their start time order. For instance, using the network of Fig. 7.7(A), ES1 was assigned to ER1. In *Priority-Matching*, the requests nodes in the graph were matched to services based on the size of their request, i.e., the weight of the edge. For instance,

using the network of Fig. 7.7(A), ES1 was assigned to ER3. We ran experiments with different settings to evaluate the *effectiveness* and *efficiency* of each approach.

7.5.1 Dataset Description

We used a real dataset from the developed app in [5]. The dataset consists of energy transfer records between a provider (smartphone) and a consumer (smartphone). The records attributes are the provider ID, consumer ID, transaction date, time, energy services and requests' amount, and transfer duration. We use the energy dataset to generate the QoS parameters for the energy services and requests. For instance, the amount of a wireless charging transfer in mAh defines the amount of requested/provided energy. In addition, the energy dataset records of a wireless charging transfer duration define the end time of each request/service.

We augmented the energy-sharing dataset to mimic the crowd's behavior within microcells by utilizing a dataset published by IBM for a coffee shop chain in New York City². The dataset consists of transaction records of customers' purchases in coffee shops for one month. Each coffee shop consists of, on average, 560 transactional records per day and 16,500 transaction records. We use the IBM dataset to simulate the spatio-temporal features of energy services and requests. Our experiment uses the consumer ID, transaction date, time, location, and coffee shop ID from each record in the dataset to define the spatio-temporal features of energy services and requests, e.g., start and location of energy service or a request. Tables 7.1 and 7.2 present statistics about the used datasets.

7.5.2 Evaluation of the Composition Framework

We ran an extensive set of experiments to determine the *effectiveness* and *efficiency* of our proposed approaches. We evaluate the effectiveness of each approach by measuring the energy utilization, fulfillment ratio, consumers' satisfaction, and providers'

²<https://ibm.co/2O7IvxJ>

TABLE 7.1. Parameters of the experiments' settings

Energy services		
Quality of service	Dataset	Value
Start time	IBM Cafe	Check-in
End time	Wireless Energy Dataset	Transfer Duration
Energy capacity	Wireless Energy Dataset	Provided Energy Amount
Energy requests		
Query parameters	Dataset	Value
Start time	IBM Cafe	Check-in
End time	Wireless Energy Dataset	Transfer Duration
Energy capacity	Wireless Energy Dataset	Requested Energy Amount

TABLE 7.2. Statistics of the experiments' environment

Variables	Value
Total Energy Requests for coffee shop 1 in April	16830
Energy Services	100000
Duration of All Energy Services	5 - 60 minutes
Duration of Energy Requests	5 - 60 minutes
Provided Energy	5 - 100 mAh
Requested Energy	5 - 100 mAh

satisfaction. Moreover, we evaluate the efficiency of each approach by comparing their execution time while varying the number of energy services and energy requests. We run the approaches in different settings by changing the number of (1) requests and (2) services. We gradually increased each setting over the time interval T . We repeated the experiment 10,000 times at each point and considered the average value for each approach.

7.5.2.1 Energy Utilization Evaluation

In this subsection, we evaluate the effectiveness of the approaches in Energy Utilization (EU). As previously stated, the objective of the composition is to maximize the utilization of the available energy. Therefore, a high EU of a composition indicates its effectiveness in utilizing the available energy. The EU is computed using Eq. 7.1.

The first experiment compared the average EU of all approaches by changing the number of energy requests (see Fig. 7.8(A)). The number of services in this experiment was

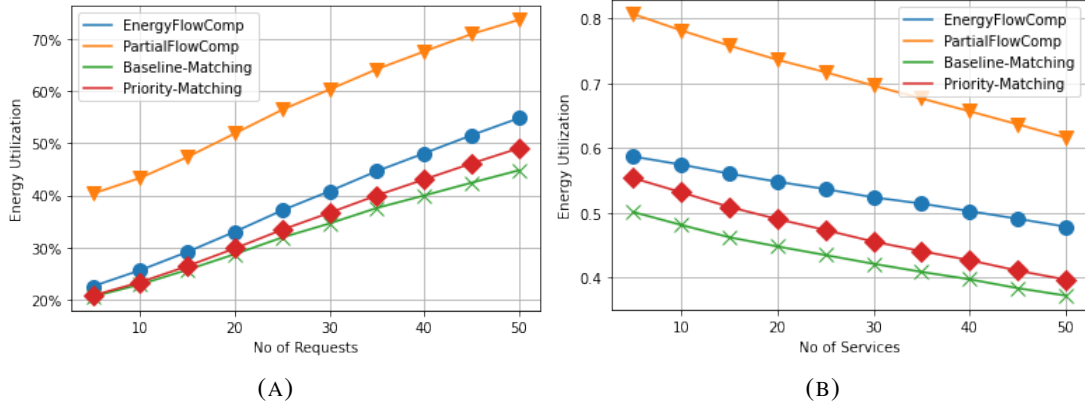


FIGURE 7.8. The average of the energy utilization EU vs. number of requests vs. (A) number of services (B)

a randomly selected set of 20 services with each run. In Fig. 7.8(A), all approaches EU increase as the number of requests increases, which can be explained by the increase in the energy demand. The higher the number of requests, the higher the probability of better allocating the available energy. Furthermore, both proposed approaches perform better than the other two approaches. The *EnergyFlowComp* approach performs better as it leverages the *fragmented* nature of energy. Specifically, when needed, energy may be shared among requests to guarantee the maximum possible energy utilization. Similarly, the *PartialFlowComp* approach outperforms all other approaches as it leverages the *fragmented* nature of energy in addition to considering the *partially overlapping* requests and services. Recall that considering partial requests and services will result in a more connected network flow and a higher chance to *efficiently* allocate the available efficiently energy.

The second experiment compared the EU by varying the number of services (see Fig. 7.8(B)). The number of requests in this experiment was a randomly selected set of 50 requests with each run. In Fig. 7.8(B), the EU decreased with the increase of services for all the approaches. The increase in service availability can explain this observation. As services increase, unused energy will increase regardless of the approach. Furthermore, both proposed approaches performed better than the others for the same reasons in the first experiment.

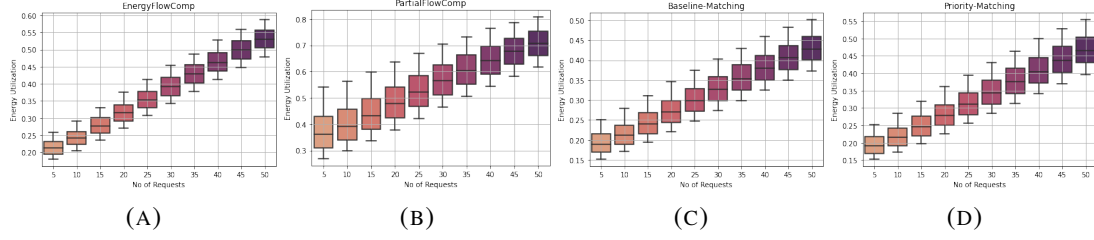


FIGURE 7.9. Distributions of the energy utilization EU score vs. number of requests

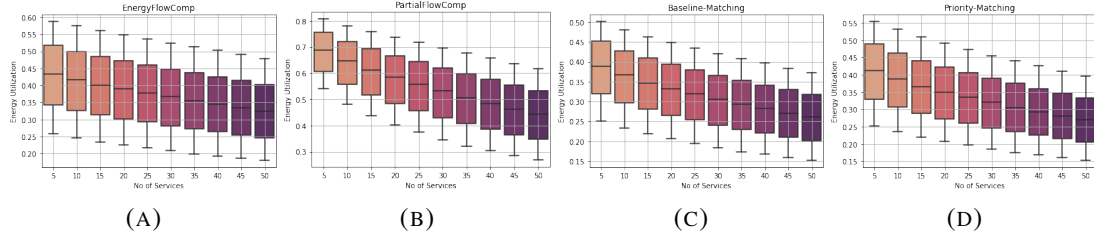


FIGURE 7.10. Distributions of the energy utilization EU score vs. number of services

We analyzed the distribution of the EU score to ascertain the previous EU evaluation. The previous experiments only considered the average EU of the available services. In this set of experiments, we varied two settings: (A) the number of requests (see Fig. 7.9) and (B) the number of services (see Fig. 7.10). All approaches have a lower EU dispersion with changing requests than changing services. The availability of services can justify this observation. Since EU relies on the amount of available energy, changing the requests will still ensure high EU as shown in Fig. 7.8(A). However, changing the services may lower the EU as shown in Fig. 7.8(B).

7.5.2.2 Fulfillment Rate Evaluation

We evaluated the effectiveness of the approaches in fulfilling the available energy requests. As aforementioned, better energy utilization may result in a better Fulfillment Rate (FR). Therefore, a higher FR indicates efficiency in fulfilling the energy demand. FR is computed using Eq. 7.2.

Fig. 7.11(A) presents the average FR for each approach. The number of services

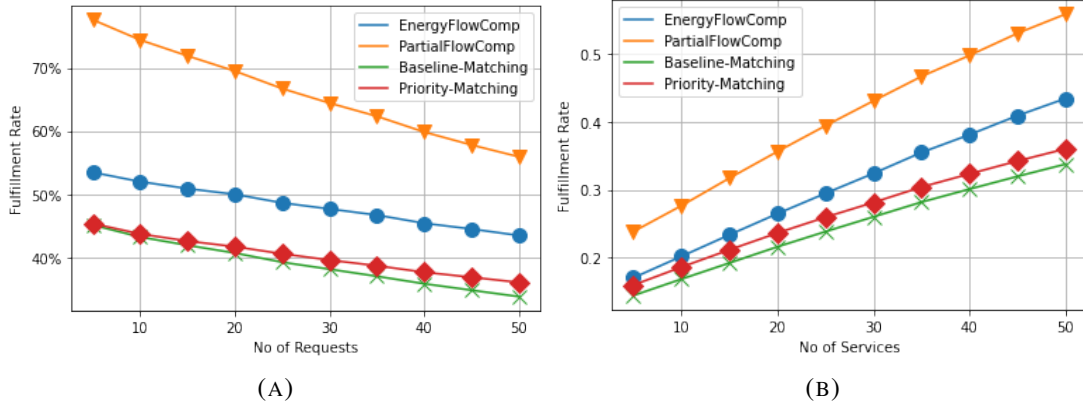


FIGURE 7.11. The average of the fulfillment rate FR vs. number of (A) requests and (B) services

in this experiment was a randomly selected set of 50 services with each run. In Fig. 7.11(A), all approaches FR decreased as the number of requests increased, which can be explained by the increased amount of requested energy. The higher the number of requests with a fixed number of energy services, the lower the percentage of fulfillment. Moreover, the *EnergyFlowComp* approach performed better than other approaches as it leverages the *fragmented* nature of energy, as previously explained. Similarly, the *PartialFlowComp* approach outperformed all approaches as it leverages the *fragmented* nature of energy in addition to considering the *partially overlapping* requests and services.

The fifth experiment compared the FR by varying the number of services (see Fig. 7.11(B)). The number of requests at each run was a randomly selected set of 50 requests. In Fig. 7.11(B), the FR increased with the increase of services for all the approaches. The increase in service availability can explain this observation. As services increased, more requests would be fulfilled regardless of the approach. Furthermore, both proposed approaches performed better than the other two for the same reasons in the previous experiment.

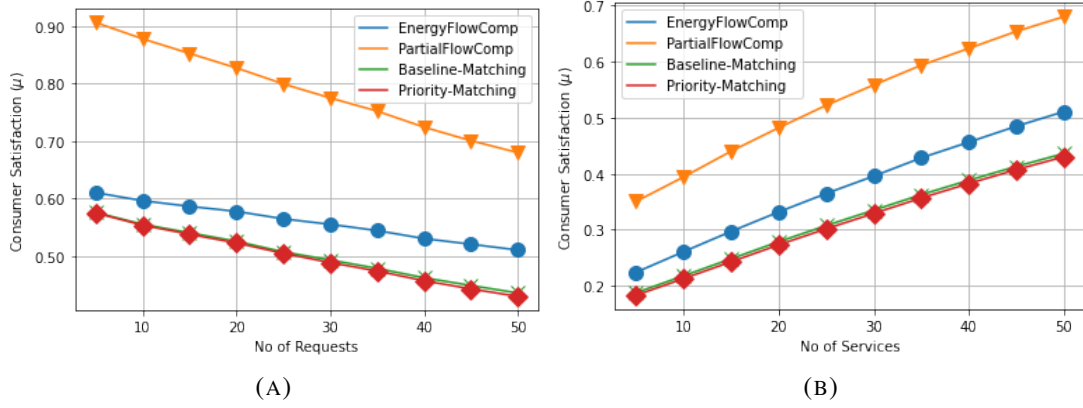


FIGURE 7.12. The mean of consumer satisfaction CS vs. number of (A) requests and (B) services

7.5.2.3 Consumers' Satisfaction Evaluation

In this subsection, we evaluate the effectiveness of the approaches in satisfying consumers. As previously stated, better energy utilization may result in higher consumers satisfaction. We assess Consumers' Satisfaction (CS) through different metrics, namely, the mean, standard deviation, and entropy satisfaction. The CS is computed using Eq. 7.3, which reflects the percentage of acquired energy per request.

Fig. 7.12(A) compares the mean CS of all approaches while changing the number of requests. Intuitively, all approaches CS decrease as the number of requests increases. The higher the number of requests, the lower the percentage of fulfillment and hence the lower the satisfaction. Similarly, Fig. 7.12(B) compares the mean CS of all approaches by changing the number of services. Intuitively, all approaches CS increase with the increase of services. The increase in service availability can explain this observation. As services increase, more requests will be fulfilled, which results in higher CS . Furthermore, both proposed approaches perform better than the others as they have a higher FR as explained in Section 7.5.2.2.

The eighth experiment presented the dispersion of CS around the mean in both settings of changing the number of (1) requests (see Fig. 7.13(A)) and (2) services (see Fig. 7.13(B)). This metric reflects the variation of the CS across requests. In Fig. 7.13(A),

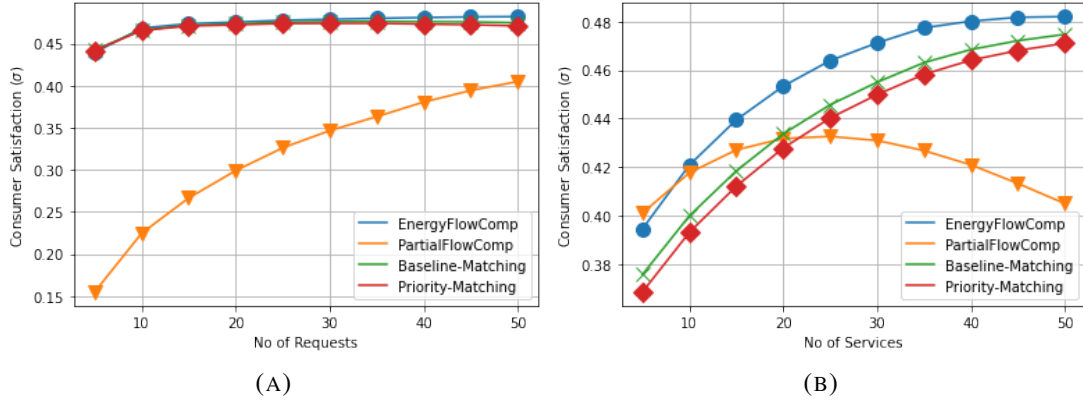


FIGURE 7.13. The standard deviation score of consumer satisfaction CS vs. number of (A) requests and (B) services

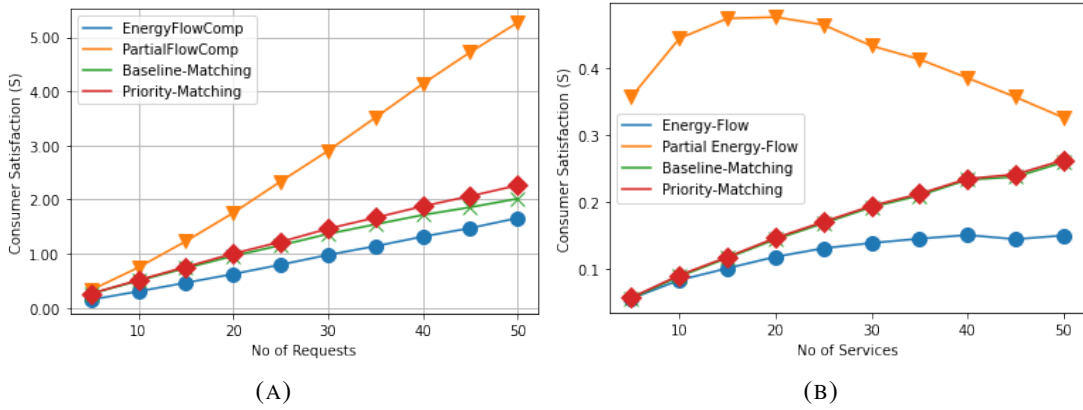


FIGURE 7.14. The entropy score of consumer satisfaction CS vs. number of (A) requests and (B) services

as the number of requests increased, the CS standard deviation converged. This is due to the decrease of CS among most requests, as presented in Fig. 7.12(A). Additionally, *PartialFlowComp* has the least dispersion because it offers the highest CS as shown in Fig. 7.12(A). In Fig. 7.13(B), we used the same CS standard deviation metric with 50 random requests and varied the number of services. The figure shows that as the number of services increased, all approaches CS standard deviations converged. In addition, *PartialFlowComp* has the least dispersion because it offers the highest CS as shown in Fig. 7.12(B).

The information entropy captures the multi-modal dispersion and irregularities in the distribution of the satisfaction score of consumers (see Fig. 7.14). We leverage the information entropy to assess the global consumers' satisfaction (see Section 7.4.2.4). A lower entropy value means a lower disorder and a higher consistency in the satisfaction score among consumers. It is worth mentioning that the entropy metric can capture small variations in the satisfaction score when the number of requests is large. These variations cannot be noticed only with the standard deviation (see Fig. 7.13). We ran two experiments to assess the entropy scores of consumers' satisfaction by varying the number of (1) requests (see Fig. 7.14(A)) and (2) services (see Fig. 7.14(B)). We used Eq. 7.5 to measure the entropy of consumers' satisfaction. In both experiments, both *Baseline-Matching* and *Priority-Matching* approaches have lower scores of entropy than *PartialFlowComp*. This can be explained by the fact that most consumers' satisfaction score using both approaches is closer to zero, unlike *PartialFlowComp*. Moreover, the low score of *EnergyFlowComp* approach demonstrates that not only does it offer a higher satisfaction compared to both *Baseline-Matching* and *Priority-Matching* but also the satisfaction score is highly consistent among consumers. In addition, *PartialFlowComp* has the highest score as requests increase, but it converges when services increase. This shows that *PartialFlowComp* does not guarantee equal individual satisfaction for consumers when the demand is higher than the available energy. This may be addressed in future work.

7.5.2.4 Providers' Satisfaction Evaluation

We evaluate the effectiveness of the approaches in satisfying providers. As previously stated, better energy utilization may result in *satisfying* more providers by using their services and thereby getting more rewards. We assess Providers' Satisfaction (*PS*) through different metrics, namely, the mean, standard deviation, and entropy satisfaction. The *PS* is computed using Eq. 7.4, which reflects the percentage of utilized energy per service.

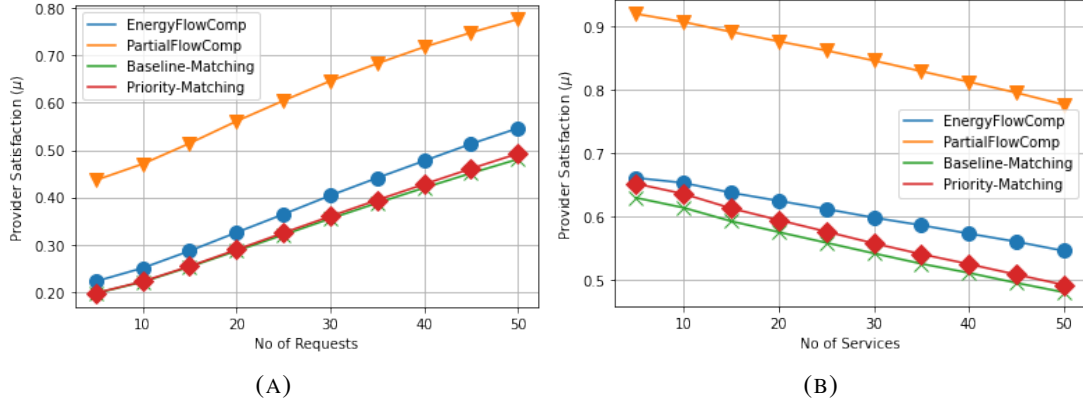


FIGURE 7.15. The mean of providers' satisfaction PS vs. number of (A) requests and (B) services

Fig. 7.15(A) compares the mean PS of all approaches while changing the number of requests. Intuitively, all approaches PS increase as the number of requests increases. This is because the higher the number of requests, the higher the probability of better allocating the available energy and, thereby, the higher the providers' satisfaction. Similarly, Fig. 7.15(B) compares the mean PS of all approaches by changing the number of services. Intuitively, all approaches PS decrease with the increase of services. The increase in service availability can explain this observation. As services increase, more services will not be needed, which results in lower PS . Furthermore, both proposed approaches performed better than the others as they have a higher EU , as explained in Section 7.5.2.1.

The following experiment presents the dispersion of PS around the mean in both settings of varying the number of (1) requests (see Fig. 7.16(A)) and (2) services (see Fig. 7.16(B)). This metric reflects the variation of the PS across services. In Fig. 7.16(A), as the number of requests increases, the PS standard deviation converges. This is due to the increase of PS among most services, as presented in Fig. 7.15(A). Additionally, *PartialFlowComp* has the least dispersion because it offers the highest PS as shown in Fig. 7.15(A). In Fig. 7.16(B), we used the same PS standard deviation metric with 20 random requests and varied the number of services. The figure shows that as the number of services increases, all approaches PS standard deviations converge. This

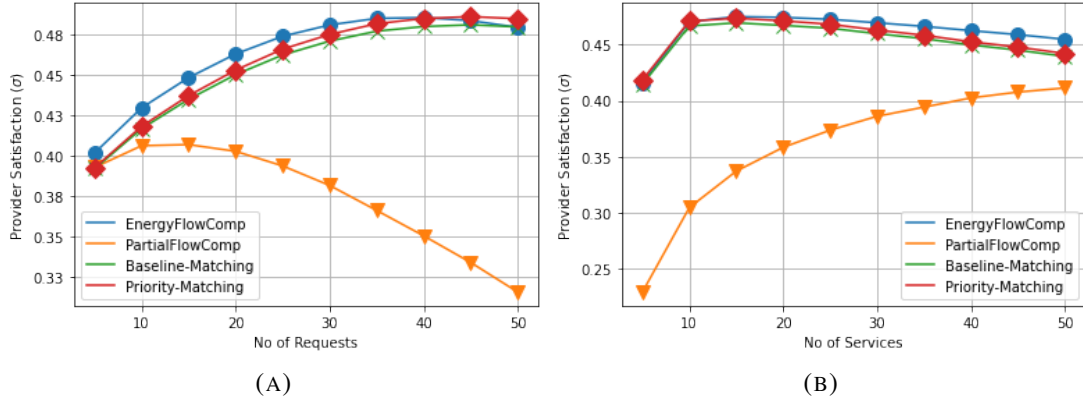


FIGURE 7.16. The standard deviation score of providers' satisfaction PS vs. number of (A) requests and (B) services

is due to the decrease of PS among most services as presented in Fig. 7.15(B). In addition, *PartialFlowComp* has the least dispersion because it offers the highest PS as shown in Fig. 7.15(B).

Similar to consumers' satisfaction, we leverage the information entropy to assess the global providers' satisfaction (see Section 7.4.2.4). As aforementioned, a lower entropy value means a lower disorder and a higher consistency in provider satisfaction scores. We ran two experiments to assess the entropy score of providers' satisfaction by varying the number of (1) requests (see Fig. 7.17(A)) and (2) services (see Fig. 7.17(B)). We used Eq. 7.6 to measure the entropy of providers' satisfaction. In both experiments, both *Baseline-Matching* and *Priority-Matching* approaches have lower scores of entropy than *PartialFlowComp*. This may be explained by the fact that most providers' satisfaction scores using both approaches are closer to zero, unlike *PartialFlowComp*. Moreover, the low score of *EnergyFlowComp* approach demonstrates that not only does it offer a higher satisfaction compared to both *Baseline-Matching* and *Priority-Matching* but also the satisfaction score is highly consistent among providers. Additionally, *PartialFlowComp* has the highest score as services increase, but it converges when requests increase. This shows that *PartialFlowComp* does not guarantee equal individual satisfaction for providers when the available services are higher than the required energy. This may be addressed in future work.

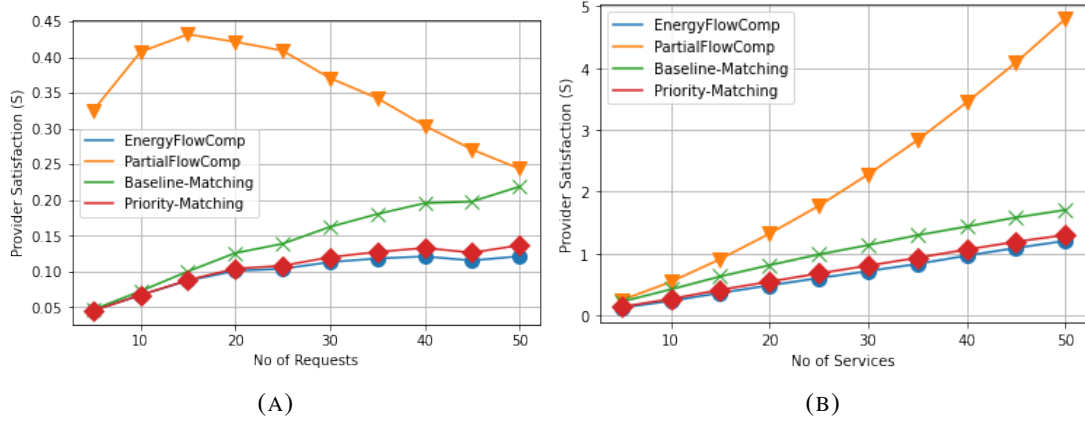


FIGURE 7.17. The entropy score of providers' satisfaction PS vs. number of (A) requests and (B) services

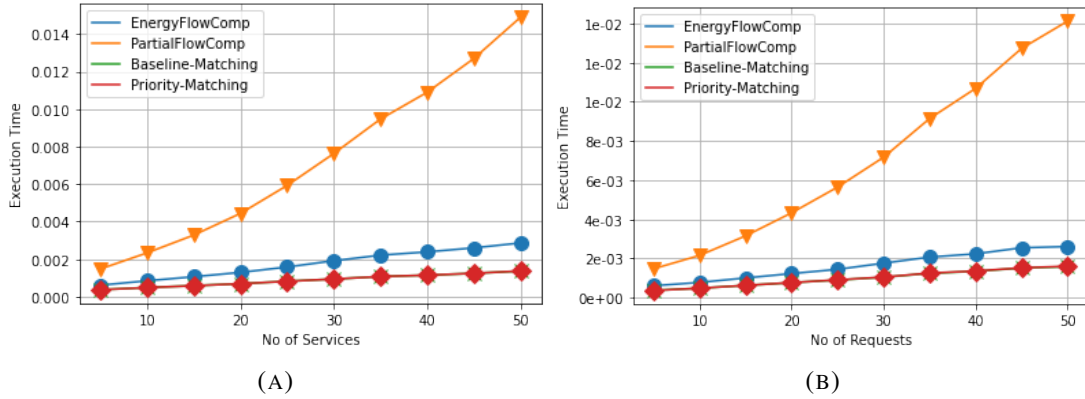


FIGURE 7.18. The average execution time vs. number of (A) requests and (B) services

7.5.2.5 Computation Efficiency Evaluation

The last set of experiments evaluated the computation cost of all approaches. Fig. 7.18 shows the average execution time of each approach. The execution time for all approaches increases as the number of requests/services increases. Intuitively, the more requests/services available, the more processing is required to match these requests to services. Moreover, *PartialFlowComp* has the highest computation cost as it considers services and requests that partially match. Recall that *PartialFlowComp* results in a more connected network that requires more processing (see Fig. 7.7).

7.6 Conclusion

We proposed a novel spatio-temporal service composition framework for sharing IoT energy services among consumers. We designed a service model for IoT energy services that aggregate the wearable-based harvested energy per energy proxy. The aggregated energy will be stored in an energy container (i.e., a smartphone's battery). Energy services are delivered to consumers through wireless power transfer channels. The proposed framework aims at fulfilling the requirements of multiple energy requests. We reformulated the problem of spatio-temporal service composition to provision multiple energy requests as a matching problem. We designed and developed two composition approaches that efficiently provision multiple IoT energy services to accommodate multiple energy requests. We conduct a set of extensive experiments to assess the effectiveness and efficiency of the proposed framework. Managing the energy storage limitation and the dynamic behavior of energy usage will be the focus of our future work.

Recommendation-based Crowdsourcing of Energy Services

8.1 Introduction

Typically, the dynamic nature of the crowdsourced energy environment may severely affect the *balance* between available and required energy services within a microcell. The *imbalance* between energy *demand* and *supply* within a microcell may impact the sustainability of the energy crowdsourcing ecosystem [48][39][14]. We define the microcell *supply* as the total available energy in the microcell within a given period. Similarly, we define microcell *demand* as the total requested energy in the microcell. The imbalance between energy supply and demand may result in *oversupplied* and *undersupplied* microcells. Excess energy services may be wasted in oversupplied microcells. Concurrently, undersupplied microcells may not fulfill consumers' energy requests. This disparity can adversely affect the efficiency and sustainability of an energy crowdsourcing ecosystem.

Existing research addresses the problem of energy service imbalance by proactively planning for when and where consumers can request energy [39]. Planning ahead enables consumers to avoid undersupplied areas. However, this approach assumes, which may not always be true, that consumers will always have at least one microcell in their path with available energy to share. Another proposed solution is to use an incentive model to encourage providers to change their modes of transportation to go to undersupplied areas [96]. Nevertheless, incentivizing providers to change their route may incur higher reward costs. Moreover, it is *challenging* to convince providers to change their preferred route solely for the purpose of service provisioning.

We propose a novel *recommendation-based* framework to crowdsource IoT energy services. Our approach balances under-supplied microcells by *recommending* an optimal provisioning plan that aligns with the spatio-temporal preferences of providers. The framework distributes energy providers to demanding microcells according to the provider's spatio-temporal preferences. Intuitively, considering providers' preferences may *reduce their resistance* to go to a recommended microcell. Reducing resistance results in a lower incentive cost [96]. Hence, we focus on fulfilling microcell energy demands by recommending the best available providers. The best providers are determined by (i) the amount of energy they can offer, (ii) the *alignment* of their spatial preferences with microcell *locations*, and (iii) the *alignment* of their temporal provisioning preferences with microcell *time-constrained* energy demands. The alignment of providers' spatio-temporal preferences with the spatio-temporal energy demands of microcells minimizes the cost of incentives while effectively fulfilling the energy demand. Therefore, we propose a new mobile energy service model that captures the spatio-temporal preferences of providers based on their mobility patterns. The main contributions of this chapter are as follows:

- A novel *mobile* energy service model that captures the provider's *dynamic* spatio-temporal provisioning preferences.
- A *multi-objective* recommendation framework to *maximize* the energy fulfillment across the city and *minimizes* the cost of providers' rewards.
- A heuristic-based approach to select the optimal providers for the under-supplied microcells.

8.2 Motivating Scenario

Assume a smart city, e.g., Sydney, is divided into four confined areas (i.e., microcells) such as coffee shops and restaurants (see Fig. 8.1(A)). Each microcell has users with IoT devices capable of transmitting and receiving energy through wireless charging technologies such as Powercast¹. IoT users can request energy services as energy

¹powercastco.com

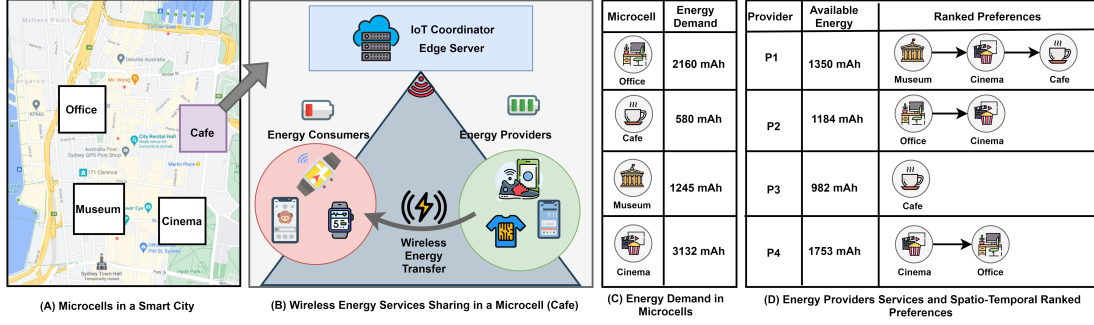


FIGURE 8.1. Crowdsourcing IoT energy services in the microcells of a Smart city with different energy demands

consumers or advertise their energy services as energy providers. All energy advertisements and requests are submitted to an IoT coordinator within the microcell (e.g., a router at the edge). Each microcell has energy demand at a specific period t (see Fig. 8.1(C)). The energy demand of a microcell may be estimated using its visitors' energy history records [164]. A *centralized super IoT coordinator* located in the cloud aims to fulfill the city's energy demands. The super IoT coordinator has prior knowledge of energy providers and their mobility patterns (see Fig. 8.1(D)). The providers' movement and provisioning estimations can be computed using their history in the city [39]. We assume that providers may have a set of preferred microcells to visit at time t . These places are determined to be frequently visited microcells based on the mobility patterns of the providers. The super IoT coordinator seeks to *proactively* fulfill the energy demands in the city by using incentives to encourage and recommend providers to the undersupplied microcells. We assume that an effective incentive model ensures that providers follow the recommendations [165]. Providers will use these rewards to request energy when they act as consumers.

There are several possible solutions to assign providers to microcells. The first recommendation approach may assign providers by suggesting the most demanding microcell for energy providers. For example, in Fig. 8.1, P_1 is recommended to go to the cinema microcell, as the cinema has the highest energy demand among all microcells. This approach may perform well in fulfilling energy demands and utilizing available energy services. However, it ignores the energy provider's preferences, i.e., the cinema

is the second preference of P_1 . This may result in recommending a microcell outside the spatial preferences of the providers. Consequently, energy providers may resist going to the recommended microcell. The possible resistance of providers may result in a higher reward cost. A better recommendation approach is to utilize the spatio-temporal provisioning preferences of the providers. For example, in Fig. 8.1, P_4 will be recommended to go to the cinema microcell. The limited energy services and constraints of providers' preferences represent challenges for efficiently provisioning IoT energy services. We reformulate our service provisioning problem as a multi-objective time-constrained optimization problem, i.e., (1) maximize the fulfillment of microcells' energy demand, and (2) minimize the cost of rewards offered to providers to go to a microcell.

8.3 Preliminaries

To formalize the problem, we use the following definitions. We adopt the definitions of energy services from [25].

Definition 1: IoT Energy Service (ES). An IoT energy service ES captures the associated attributes of an energy service. It can be formally defined as a tuple $ES(id, oid, F, Q)$.

- id is a unique IoT energy service ID.
- oid is a unique ID for the owner of the IoT device.
- $F = \{f_1, \dots, f_n\}$ is the set of service functionalities provided by the IoT device.
- $Q = \{q_1, q_2, \dots, q_n\}$ is a tuple of the energy service's Quality of Service (QoS) properties.

Definition 2: IoT Energy Quality of Service (QoS) Attributes The QoS attributes provide additional information regarding the energy service. It allows energy consumers to distinguish between crowdsourced IoT energy services [25]. It can be formally defined as a tuple $\langle loc, st, et, ae \rangle$.

- loc represents the location of the energy provider.
- st is the start time of the crowdsourced energy service.
- et is the end time of the crowdsourced energy service.
- ae is the available energy to be provided by the IoT energy service.

Definition 3: Provider Mobility Pattern (MP). A mobility pattern MP captures the frequent spatio-temporal sequence of an IoT user during a daily routine [166]. Therefore, the mobility pattern MP can be formally defined as a tuple $MP(TS, LOC)$.

- $TS = \{ts_1, ts_2, \dots, ts_n\}$ refers to the sequence of timestamps.
- $LOC = \{loc_1, loc_2, \dots, loc_n\}$ is the sequence of corresponding location under a certain timestamp ts_i .

Definition 4: Mobile Energy Service (MES). A mobile energy service MES captures the spatio-temporal preference of a provider and the available energy amount to share within a specific time interval. Thus, the mobile energy service MES can be formally defined as a tuple $MES(PT, ES, M)$.

- $PT = \{t_1, t_2, \dots, t_n\}$ is the set of time intervals for energy provisioning.
- $ES = \{es_1, es_2, \dots, es_n\}$ is the set of corresponding available energy services for each time interval.
- $M_p = \{m_1, m_2, \dots, m_n\}$ is the set of possible microcells that the energy provider p regularly visit.

Definition 5: City energy Demand model (CD). The city energy demand model CD captures the energy demand of all microcells in the city within a specific time interval ct . Therefore, the city energy demand model CD can be formally defined as $CD(ct, M_c, D_i)$.

- ct is a time interval in the smart city.
- $M_c = \{m_1, \dots, m_n\}$ is the set of microcells in the smart city c .

- $D = \{d_1, \dots, d_n\}$ is the amount of energy demand for each microcell within ct .

Definition 6: Rewards Model (RM). We extend the incentive model presented in Chapter 3 to consider the provider's spatio-temporal preferences in calculating the provisioning rewards. The reward model RM determines the amount of reward rw given to an energy provider P to motivate them to go to a microcell within a certain time interval. The IoT coordinator at the city level gives rewards. We assume that the system limits the amount of requested energy by consumers. As an energy provider may later become a consumer, they can utilize the earned rewards to increase the limit of requested energy. As previously mentioned, providers may have a preferred set of microcells to visit. These microcells may be ranked based on the provider's preferences, i.e., convenience, experience, etc. Hence, the IoT coordinator at the city level may have to pay a higher reward to assign an energy provider to a less preferable microcell. Thus, the energy provider reward can be defined as the following:

$$rw(es_i, m_k, t_i) = es_i.ae / Rate * PrefRank(m_k, t_i) \quad (8.1)$$

where rw represents the reward of changing the behavior of provider P_l at a time interval t_i , $es_i.ae$ is the available energy service of provider P_l at time interval t_i , $Rate$ is a value determined by the IoT coordinator at the city level that reflects the credit of providing a certain amount of energy, $PrefRank$ is a function that retrieves the rank of the provider preference for a given microcell m_k at a time t_i . $PrefRank$ value varies from $[1-n]$ where 1 refers to the most preferred microcell, and n represents the least preferred microcell.

We use the following assumptions to formulate the problem.

- Providers' energy size is fixed during composition.
- Providers' mobility patterns are given by estimating their historical records [167].
- Providers will follow the recommendation.
- Providers and consumers have fixed locations at the time interval (t) .

- There is no energy loss in sharing. As the technology matures, we anticipate that the devices will be able to share more energy, and the energy loss of sharing will become minimal [24].
- The exact amount of required energy for each microcell is given.
- The allocated energy for a microcell will be completely utilized, and allocating it to the available requests is out of the scope of this study.
- A reward system incentivizes providers to offer their service.
- A trust framework is used to preserve the privacy and integrity of the IoT devices [168][21].

8.3.1 Problem Definition

Given a city demand model CD with k microcells and n energy providers' energy services and mobility patterns in a time interval t . The IoT coordinator utilizes the providers' services and mobility patterns to define mobile energy services $MES = \{MES_1, MES_2, \dots, MES_n\}$. The IoT coordinator at the city level uses the MES to assign the providers' services to microcells in demand. The recommendation approach aims at *maximizing* the energy fulfillment FR across microcells while *minimizing* the cost C of the reward model RM . We transform the problem of recommendation-based energy provisioning into a time-constrained resource allocation problem as follows:

$$\begin{aligned}
 & \text{Maximize} \quad FR(MES, CD, t) \\
 & \text{Minimize} \quad C(MES, CD, t) = \sum_{i=1}^n rwd(es_i, m_k, t_i) \\
 & \text{Subject to} \quad MES_{j.t_i} \subset ct \\
 & \text{Where} \quad \forall MES_j \in MES, \\
 & \quad \quad MES_{j.t_i} \text{ is the provisioning time of the } j^{th} \text{ mobile} \\
 & \quad \quad \text{energy service}
 \end{aligned} \tag{8.2}$$

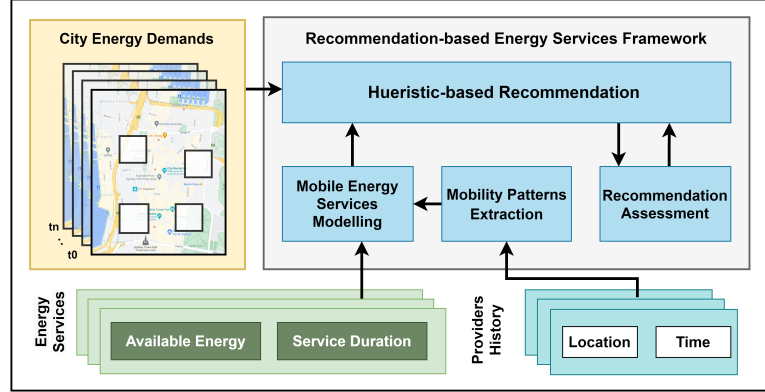


FIGURE 8.2. Recommendation-based energy services framework

8.4 Recommendation-based Energy Services Framework

As previously stated, our framework will be managed by a centralized super IoT coordinator in the city. Assume we have a set of IoT energy providers in a smart city, $P = P_1, P_2, \dots, P_n$, and a set of all microcells energy demand across the city within the same time interval t , $D = d_1, \dots, d_n$. The *super IoT coordinator* aims at *maximizing* the energy fulfillment across the microcells while *minimizing* the cost of rewards within a time interval t using our proposed recommendation-based energy services framework (see Fig. 8.2). The framework is divided into four phases: (1) Mobility patterns extraction, (2) Mobile energy services modeling, (3) Heuristic-Based Recommendation, and (4) Recommendation assessment. The framework takes as input the providers' energy service, providers' movement history, and the city's energy demands. In the first phase, the super IoT coordinator will extract the mobility patterns of providers from their history. In the second phase, the super IoT coordinator will model the mobile energy services based on their energy services and mobility patterns. In the third phase, providers will be recommended to microcells based on the demand in these microcells and the providers' provisioning preferences. The super IoT coordinator will assess the resulting recommendation in the last phase. In the following subsections, we discuss each phase in detail.

8.4.1 Mobility Patterns Extraction

Users' mobility patterns (MP) are needed to identify the preferred locations and times for providers to visit at the city. The mobility patterns of providers may be extracted from their movement history in the city. In this phase, the super IoT coordinator extracts the provider's spatio-temporal preferences by statistically analyzing the mobility pattern using the *Prefix-Projected Sequential Patterns Mining (PrefixSpan)* [169]. PrefixSpan is a sequential pattern mining technique that discovers frequent sub-sequence patterns from a large sequence database. The set of frequently visited locations and time intervals for a provider will be assigned as a sequence of timestamps (TS) and as a sequence of corresponding locations LOC under each timestamp (see Definition 3).

8.4.2 Mobile Energy Services Modelling

The goal of this phase is to capture providers' spatio-temporal *mobile patterns* and *provisioning* preferences to define the Mobile Energy Services set (MES) within a time interval t as described in Definition 4. The super IoT coordinator will utilize the extracted mobility patterns from the previous phase to define a set for each provider. The set of frequently visited locations and time intervals for a provider will be assigned as the set of regularly visited microcells (M_p) and the provider's time preferences for energy provisioning (PT) of the mobile energy service (see Definition 4). Then, the energy services ES will be filtered based on their time interval (i.e., st and et) and available energy ae . An energy service will be selected if (1) its time interval falls with t , and (2) its available energy ae is larger than 0 mAh. The set of energy services that a provider may offer over time will be defined as ES of the mobile energy service where each energy service es_i is offered on time t_i at location m_i .

8.4.3 Heuristic-Based Recommendation

This phase leverages our proposed Heuristic-Based recommendation approach (HB) to recommend undersupplied microcells to providers to offer their services. The HB

Algorithm 8.1 Heuristic-based Recommendation Algorithm

Input: $\mathcal{MES}, CD$ // $\mathcal{MES}$ is the set of MES in time interval t , CD is the city demand model in time interval t

Output: Rec // Rec is the set of recommendations for every provider in the form of (P, m) in time interval t

```

1:  $Rec = \emptyset$ 
2:  $k = \text{length}(CD.M)$ 
3:  $n = \text{length}(\mathcal{MES})$ 
4:  $sMES \leftarrow \text{sort}(\mathcal{MES}; es_i.ae : \text{descending})$ 
5:  $sCDM, sCDD \leftarrow \text{sort}(CD.M, CD.D; d : \text{descending})$ 
6: for  $i \leftarrow 1$  to  $n$  do
7:    $mIndex = -1$ 
8:   for  $j \leftarrow 1$  to  $k$  do
9:     if  $sCDD[j] > 0$  and  $sCDM[j] \in sMES[i].M_p$  then
10:       $mIndex \leftarrow j$ 
11:      BREAK
12:   if  $mIndex == -1$  then  $mIndex \leftarrow 1$ 
13:    $Rec.add((P_i, sCDM_{mIndex}))$ 
14:   if  $sCDD[mIndex] > sMES[j].es.ae$  then
15:      $sCDD[mIndex] - = sMES[j].es.ae$ 
16:   else
17:      $sCDD[mIndex] = 0$ 
18:    $sCDM, sCDD \leftarrow \text{sort}(sCDM, sCDD; d : \text{descending})$ 
19: return  $Rec$ 

```

algorithm takes as its input the defined mobile energy services MES of all providers and the city energy demand model at time t . The HB algorithm outputs the recommended microcell for each provider Rec . Our proposed algorithm aims to align the energy demands across microcells to the providers' spatio-temporal preferences. This alignment may result in maximizing the fulfillment of energy demands while minimizing the cost of rewards.

Algorithm 8.1 presents the pseudocode of the heuristic-based recommendation algorithm. First, the recommendation list is initialized (Line 1). Then, the number of microcells and mobile energy services are computed (Lines 2-3). $\mathcal{MES}$ is then sorted in descending order based on the amount of available energy ae at time t (Line 4). Line 5 sorts in descending order the microcells and their corresponding demands based on the size of the energy demand. For each mobile energy service, the algorithm aims to

find the most demanding microcell in the provider's spatial preferences (Lines 8-11). If none of the microcells matches the provider's preferences, the algorithm assigns the provider to the most demanding microcell in the smart city (Lines 12-13). The corresponding microcell demand is updated once a provider is assigned to a certain microcell (Lines 14-17). Subsequently, the microcells and the corresponding demands will be sorted again to be used in the recommendation of the following mobile energy service (Line 18). Line 19 returns the recommended microcell for each provider *Rec*.

8.4.3.1 Complexity Analysis

The time complexity of the proposed algorithm can be estimated based on the number of providers n , the number of microcells m , and the number of preferences for each energy provider. Given that a provider may have, at maximum, all microcells in their preferences m , the time complexity of the algorithm is $O(nm^2)$.

8.4.4 Recommendation Assessment

As mentioned previously, the framework aims to maximize fulfilling the energy demands across microcells and minimize the cost of rewards. We rely on the following metrics to evaluate the performance of our proposed algorithm.

- **Fulfillment Ratio *FR*:** We define the fulfillment ratio score to gauge the energy fulfillment across microcells. This score describes the ratio of allocated energy services Al to total energy demand $CD.d$ across all microcells at time interval t . For example, assume that a provider P_1 offers 1,365 mAh energy, and they are assigned to a microcell with a 1,659 mAh energy demand. The energy fulfillment score of the microcell is $1,365/1,659 = 0.87$. The *FR* score ranges from 0 to 1, where 0 represents that none of the microcell's energy demands were fulfilled, and 1 represents that all microcell's energy demands

have been fulfilled. We use the following equation to determine FR :

$$FR(MES, CD, t) = \sum_{i=1}^n Al_i / \sum_{j=1}^k CD.d_j \quad (8.3)$$

where Al_i refers to the allocated energy by provider P_i to a microcell m_j , d_j is the microcell m_j 's energy demand, k represents the number of microcells, and n represents the number of providers at time interval t .

- **Rewards Cost C :** The cost score measures the expense of motivating providers to offer their services at the recommended microcells using our reward model. The cost score determines the reward amount to be paid to providers. An expensive recommendation plan is reflected by having a higher score. Contrarily, a lower cost score indicates a less costly recommendation plan. We use the following equation to determine the C :

$$C(MES, CD, t) = \sum_{i=1}^n rwd(MES_i.es_j, CD.m_k, t) \quad (8.4)$$

where rwd refers to the reward given to the energy providers when they are assigned to a certain microcell m_k , rwd is computed using Eq. 8.1, n represents the number of energy providers in the system at time interval t .

- **Energy Usage EU** The energy usage EU assesses the effectiveness of our approach in allocating and leveraging the available energy services. EU represents the ratio of the allocated energy services Al to all the available services MES_{es} at time interval t . The score ranges from 0 to 1, where 0 represents that none of the available energy is allocated to the microcells, and 1 represents that all available energy has been completely allocated. We use the following equation to determine the EU :

$$EU(MES, CD, t) = \sum_{i=1}^n Al_i / \sum_{i=1}^n MES_i.es_j.ae \quad (8.5)$$

where Al_i refers to the allocated energy service by provider P_i , ae is the available energy service offered by the provider, n refers to the number of available energy providers in the city at time interval t .

8.5 Evaluation

We compare our proposed Heuristic-Based recommendation (HB) approach with two greedy recommendation approaches (i.e., Energy-Aware Recommendation (EA) and Preference-Aware Recommendation(PA)). EA recommends to providers the most demanding microcells regardless of the providers' spatial preferences. In contrast, PA recommends to providers their most preferred microcells regardless of the microcells' energy demands. Additionally, we compare our approach with a state-of-the-art resource Allocation algorithm based on Genetic algorithms (GA). GA is a modified Non-dominated Sorting Genetic Algorithm II (NSGA-II) [170]. NSGA-II extends the traditional genetic algorithm to solve multi-objective optimization problems. Recall that the objectives of our framework are to maximize the energy fulfillment ratio FR across microcells and minimize the cost of rewards C . We ran a set of experiments with different settings to evaluate the *effectiveness* and *efficiency* of each approach.

8.5.1 Dataset Description

We used a real dataset from the developed app in [6]. This app records the wireless energy-sharing process occurring via coils linked to a pair of smartphones. Users can request energy services from nearby devices based on capacity (e.g., 500 mAh) or duration (e.g., a 15-minute charge). The dataset contains records of energy transfers between provider and consumer smartphones, with details such as provider ID, consumer ID, transaction date and time, energy service quantity, request quantity, and transfer duration. The energy dataset generates QoS parameters for energy services and requests. For instance, the wireless charging transfer amount in mAh is applied to determine the requested/provided energy quantity. Moreover, the energy dataset is

utilized to establish the completion time for each request or service. Due to the limitations of the current technology, the amount of transferred energy between devices is small. Therefore, we had to scale up the services/requests size. As a result, the size of the services/requests was scaled up between 200 mAh and 500 mAh. In addition, the energy demand of each microcell was computed as an aggregation of the energy requests located in that microcell. Hence, the energy demand in each microcell is between 1,000 mAh and 2,000 mAh.

We augmented the energy sharing dataset to mimic the crowd's behavior across microcells using *Foursquare*, a geo-tagged real social dataset [166]. The microcells' locations were defined from the available venues in the dataset. Moreover, the spatio-temporal mobility patterns of the energy providers within a city were defined from the users' history in the dataset. The dataset consists of check-in records of the platform users in New York City. There are 227,428 records in the dataset. The dataset contains data from April 2012 to February 2013. The attributes of the records are user ID, venue ID, venue category ID, venue name ID, time, and latitude and longitude of the venue. We used the user records between April 2012 to June 2012. We also removed users with fewer than fifty days of records within that period as they do not have enough history to define their mobility patterns [169]. We used the remaining users as the energy providers and extracted from their history the providers' mobility patterns using *PrefixSpan* [169]. We used a minimum support threshold of 50% of the users' records. For the sake of simplicity, we assumed that all mobility patterns are aligned into a unified time interval (e.g., 24 hours). The time interval is further partitioned into a set of smaller intervals based on the length of the mobility patterns. For example, within the first time interval t_1 , the first element of a provider mobility pattern consists of a set of possible microcells that they may visit. To define the rank of the spatio-temporal preferences of an energy provider, we ranked the set of microcells based on their frequencies in a user's history, i.e., the most frequent microcell in a user's records at a given time is the most preferential microcell of the provider. We ran a total of 12,000 experiments. In each run, the energy providers' mobility

TABLE 8.1. Experiments Variables

Variables	Value
Check-in records in Foursquare dataset	227,428
Number of users extracted from dataset	34
Average check-in records for each user	275
Number of providers	5-30
Number of microcells	5-30
Amount of services	200-500 mAh
Amount of requests	1000-2000 mAh

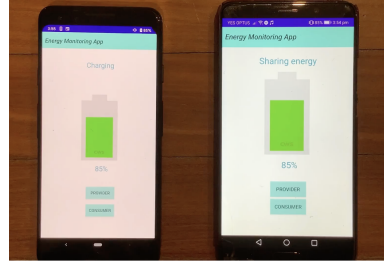


FIGURE 8.3. Wireless Energy Dataset Collection Experiment Using the App Developed by [5]

patterns were selected randomly from the Foursquare dataset. Table 8.1 presents the experiments' parameters and the datasets' statistics.

8.5.2 Evaluation of the Recommendation Framework

We ran five experiments to determine the *effectiveness*, and *efficiency* for all approaches. The experiments evaluated the approaches in terms of their fulfillment ratio, cost of rewards, energy usage, the impact of responsive rate, and computation cost. We evaluated the approaches under different setups by changing the ratio between the number of providers and microcells in the smart city in time interval t . The ratio gradually increases from 0.33 to 2. In the first setup, we fixed the number of providers and gradually increased the number of microcells. In the second setup, we fixed the number of microcells and gradually increased the number of providers. For each setup, we repeated the experiments 1,000 times at each point. Then, we considered the average value for each technique at each point.

8.5.2.1 Effectiveness Evaluation

The first experiment compares the energy fulfillment ratio FR for all approaches. As formerly mentioned, FR represents the energy fulfillment across all microcells in a time interval t . A higher FR refers to a better fulfillment of microcells' energy demands across the city. The FR is computed using Eq. 8.3. Fig. 8.4 depicts the FR

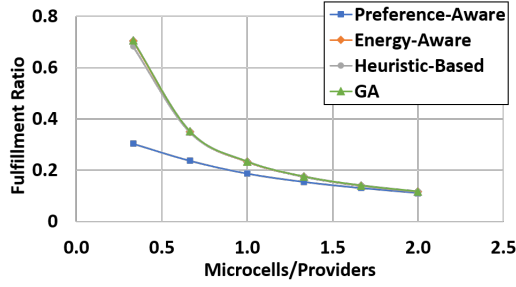


FIGURE 8.4. Average FR of different number of microcells

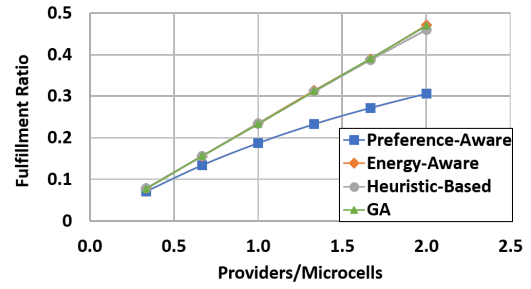


FIGURE 8.5. Average FR of different number of providers

of the four approaches when the number of providers is fixed, and the number of microcells increases. Generally, the FR of all four approaches decreases as the number of microcells increases. This observation is intuitively logical as the number of microcells increases, the overall microcell demand in the city increases accordingly. Hence, FR declines as the number of microcells increases. Simultaneously, it is observed that PA has a lower FR than all approaches because it only considers the providers' first spatial preferences regardless of the energy demand. Fig. 8.5 shows the FR of the four approaches when the number of microcells is fixed, and the number of providers increases. In general, the FR of all four approaches increases as the number of providers increases. This is due to the increase in the available energy by the increase in the number of providers. Similar to the previous experiment, PA has a lower FR than all approaches because it only considers the providers' first spatial preferences regardless of the energy demand.

The second experiment compared the cost of the rewards (C) for all approaches. As previously mentioned, using our reward model, the C measures the expense of motivating providers to offer their services at the recommended microcells. A higher C of the approach indicates a less preferred recommendation among the energy providers. The C is computed using Eq. 8.4. Fig. 8.6 shows the C of the four approaches when the number of providers is fixed, and the number of microcells increases. Generally,

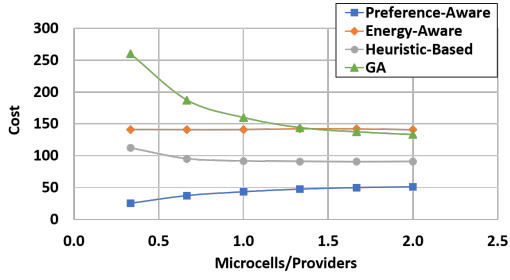


FIGURE 8.6. Average C of different number of microcells

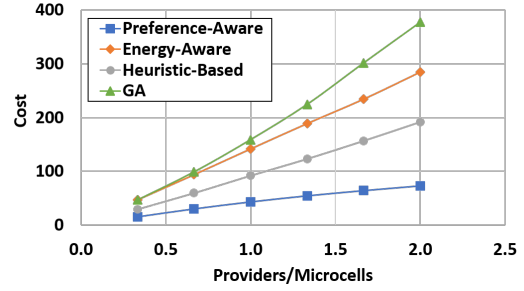


FIGURE 8.7. Average C of different number of providers

the cost of the *EA* approach is the highest as it does not consider the provider's spatial preferences in the recommendation. Thus, selecting a microcell that is not in the provider's preference is possible, which increases the cost of rewards. The *PA* approach has the lowest cost as it only considers the providers' first spatial preferences, which require minimum rewards. However, as shown in the previous experiment, *PA* has the lowest *FR* for the same reason. *HB* has the middle cost as it considers both the energy demand of the microcells and the providers' preferences, making it more expensive than *PA* but lower than *EA*. Simultaneously, the cost of *GA* is the highest among the four approaches. This can be due to the improper initialization of the original population or because *GA* converges at some local optimum, which results in a higher cost. Fig. 8.7 depicts C of the four approaches when the number of microcells is fixed, and the number of providers increases. Generally, the cost of all four approaches increases as the number of providers increases. As the number of providers increases, the cost to incentivize the providers will increase accordingly. *GA* has the highest cost as it either does not have a proper initialization or because *GA* converges at some local optimum, which results in a substandard outcome.

The third experiment compared all approaches in terms of energy usage. As formerly mentioned, the energy usage score (*EU*) represents the ratio of utilized energy services. The *EU* of a time interval t was computed using Eq. 8.5. Fig. 8.8 presents the *EU* for all approaches when the number of providers is fixed, and the number of microcells increases. Generally, *EA*, *GA*, and *HB* performed well as the number

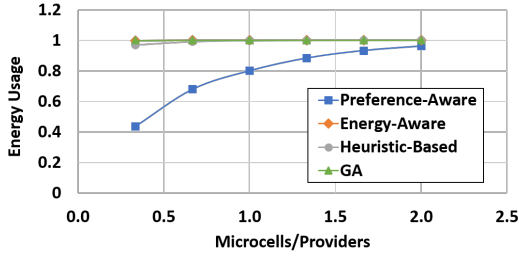


FIGURE 8.8. *EU* of different number of micro-cells

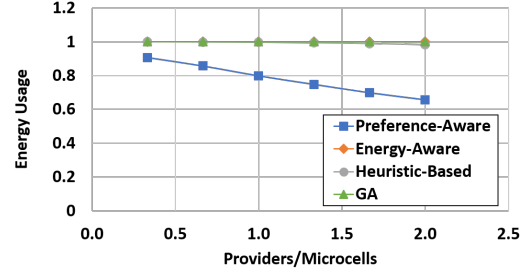


FIGURE 8.9. *EU* of different number of providers

of microcell increases. This is because they aim to leverage all the available energy to fulfill the energy demand. *PA* had the lowest score because it only considers the providers' first spatial preferences regardless of the available energy. Fig. 8.9 presents the *EU* of the four approaches when the number of microcells is fixed and the number of providers increases. Typically, *EA*, *GA*, and *HB* perform well as the number of microcells increases because they aim to leverage all the available energy to fulfill the demand. For *PA*, *EU* decreases as the number of microcells increases because *PA* only considers the providers' first spatial preferences regardless of the energy demand.

The fourth experiment studied the impact of responsive rate in terms of *EU* and *FR* across the three approaches (i.e., *PA*, *EA*, *HB*). The responsive rate measures the impact when only a percentage of the providers follow the recommendation. In the experiment, we varied the responsive rate from 20% to 100%. All approaches behaved similarly for *EU* and *FR* as the allocated energy impacts them both. Thus we only present the result in terms of *EU*. Fig. 8.10 depicts the *EU* across the three approaches when we changed the responsive rate of the providers. Generally, all three approaches showed a higher *EU* as we increased the response rate. However, *EA* and *HB* generally have a higher *EU* than *PA*. This observation is due to the design of the methods. *EA* and *HB* are designed to prioritize energy fulfillment. Thus, their *EU* should be higher than that of *PA*'s.

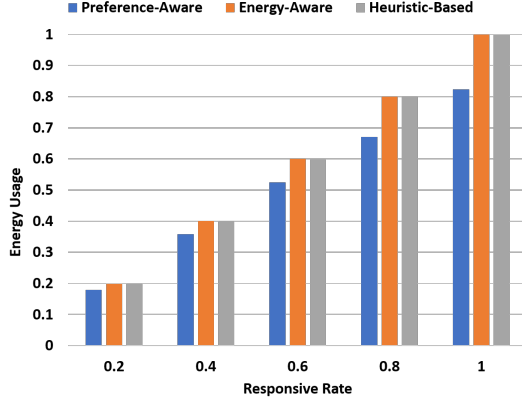


FIGURE 8.10. Average *EU* when changing the responsive rate

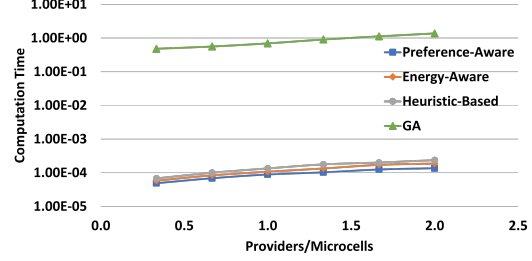


FIGURE 8.11. Computation Time

8.5.2.2 Computation Efficiency Evaluation

The fifth experiment compares the execution time between the approaches. The execution time reflects the efficiency of the algorithm. A lower execution time indicates a more efficient recommendation algorithm. Fig. 8.11 depicts the average execution time of the four approaches on a logarithmic scale. The execution time increases as the number of providers increases. When the number of providers increases, more computation is required for the approaches to assign the providers. Simultaneously, it is observed that *GA* spends a relatively longer time than the other approaches. This is because *GA* requires numerous rounds of crossover and mutation before it converges.

8.6 Conclusion

We proposed a recommendation-based framework to address the imbalance between energy services and requests across microcells in a smart city. Our Heuristic-Based recommendation approach (*HB*) leverages providers' *mobility patterns and provisioning preferences* to effectively assign them to undersupplied microcells. Experimental results show that *HB* performs comparably to the genetic algorithm (*GA*) regarding

energy fulfillment. However, *HB* has a relatively lower cost of rewards and computational power than does *GA*. The proposed framework represents one of the first attempts to address the incongruity between providers and consumers in the crowd-sourced IoT energy environment. Future work may consider the probability of providers not following the recommendations.

QoE-based Energy Services Composition

9.1 Introduction

Super-providers typically focus on ensuring that customers keep coming back to their businesses. Their revenue is usually directly related to *foot traffic* [52]. Customer *satisfaction* is therefore paramount as a strategy to either maintain or increase the business target revenue [51]. A key objective is ensuring customers have the *best experience*. In this chapter, we propose to use energy sharing as a key ingredient to provide customers with the best quality experience when visiting the business. For example, a case study showed that "Sacred", a cafe in London, had a noticeable increase in foot traffic after installing wireless charging points¹.

We define a *Quality of Experience (QoE)* metric to represent the level of *satisfaction* across energy consumers over a period of time in a specific microcell. Note that QoE is different from Quality of Service (QoS). QoE uses QoS as a base to express satisfaction with a service over a period of time. QoE has traditionally been used in domains that assess how users perceive a service [88][171][91]. Our proposed environment requires the use of a different type of QoE. In particular, we identify three aspects that shape the new QoE definition: (1) crowdsourced environment resources are usually limited and cannot fulfill all consumers' requirements. Hence, assessing consumers' satisfaction should consider the limited available resources. Energy services may be provided partially due to the limited resources and the shareable nature of energy services, e.g., a single service may be split into smaller services and provided

¹air-charge.com

among multiple consumers. In a limited resource environment, consumers' experience with partial services differs from their experience with complete services. (3) Consumers' satisfaction with energy services will indirectly impact their experience with the super-provider's microcell. Therefore, our research focuses on the *super-provider's perspective* of QoE.

Assessing the QoE from a super-provider's perspective usually entails measuring the *aggregated* satisfaction of consumers over time. *Consumer's satisfaction* is defined as *meeting or exceeding* a set of expected service goals [172]. In this context, we define consumer satisfaction as *receiving the requested energy or part of it*. We focus on *optimizing* the *QoE* by efficiently provisioning and fulfilling the consumers' energy requirements.

The limited energy availability is a key challenge that may hinder the super-provider from optimizing the consumers' QoE [24]. For instance, an energy consumer might not find their requested energy at a certain time in the microcell, resulting in an unsatisfying experience. In this context, using traditional resource allocation algorithms may incur uneven energy sharing for some consumers. Therefore, we propose a QoE-driven service provisioning framework to satisfy energy consumers in a crowdsourced IoT environment. The framework requires prior knowledge of providers' temporal preferences and the microcell energy demands. The proposed framework leverages the *shareable* nature of energy services to split the energy between consumers if the required energy is more than the available energy [25]. Intuitively, the super-provider may prefer offering part of the required services to all consumers than offering it to some. Hence, we propose a heuristic Partial-Based (PB) approach which splits services among consumers in the case of low energy availability. Another possible solution is to leverage *flexible* providers that offer services on *multiple* time slots by allocating their services to the most demanding slots. Intuitively, this may ensure a better distribution of the available services. Therefore, we additionally propose a heuristic

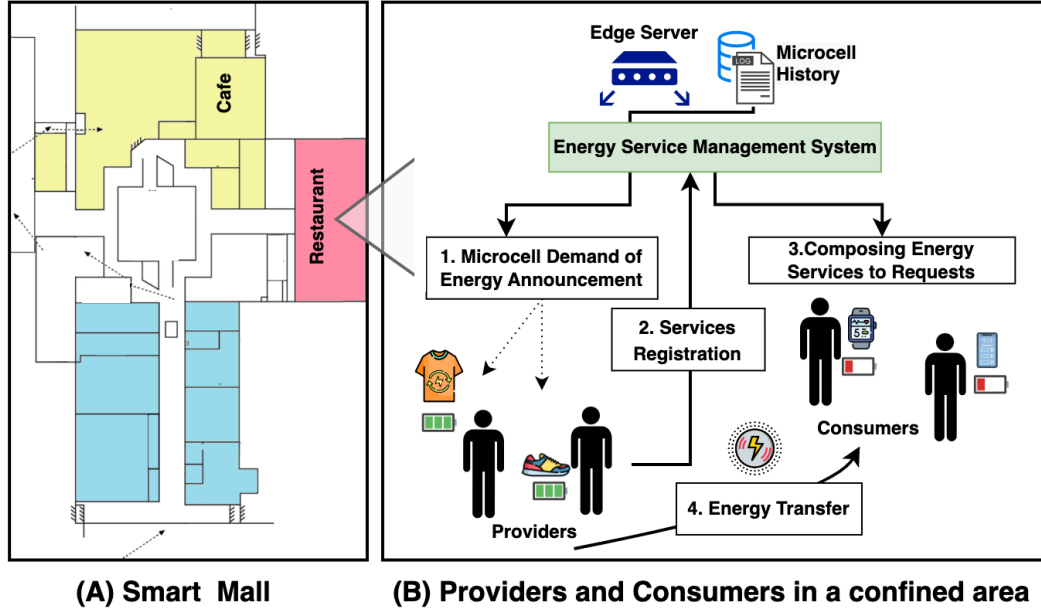


FIGURE 9.1. IoT energy services environment

Demand-Based (DB) approach. The DB approach extends the PB approach by prioritizing allocating services based on the highest demanding time slots. The main contributions of this chapter are:

- A novel Quality of Experience (QoE) model for crowdsourced energy services.
- A framework for QoE-driven composition of IoT energy services.
- An experimental analysis with two implementations of the proposed QoE-driven energy composition framework.

9.1.1 Motivating Scenario

We describe a scenario in a confined place (i.e., microcell) where people congregate, e.g., cafes and restaurants (see Fig. 9.1(A)). Each microcell may have several IoT devices acting as energy providers or consumers (see Fig. 9.1(B)). The super-provider

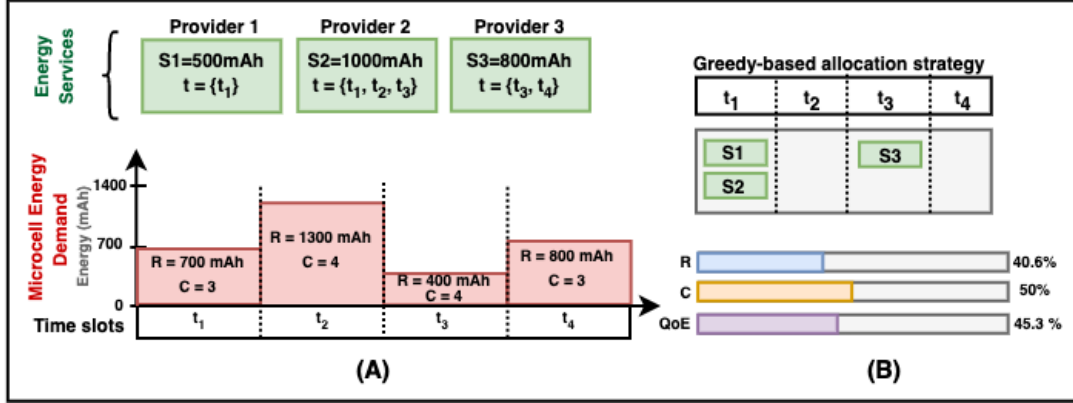


FIGURE 9.2. (A) Microcell energy demand and providers services (B) Greedy energy provisioning approach

aims to leverage the crowdsourced energy services to enhance the consumers' *experience*. We assume all local energy services and requests are submitted and managed at the *edge*, e.g., a router in the microcell (see Fig. 9.1(B)). We assume that the super-provider offers incentives to encourage energy sharing in the form of credits. These would be used to receive more energy when (the providers act as consumers in the future [24]. We assume the super-provider has a prior knowledge of the *Microcell Energy Demand* ($\mathcal{MED}$) in the microcell over a period of time (T) (see Fig. 9.2(A)). The $\mathcal{MED}$ may be estimated based on previous history [39]. The $\mathcal{MED}$ is represented in terms of the requested energy in each time slot, e.g., 700 mAh at time slot t_1 . The granularity of the time slots can also be estimated based on the previous history of the microcell [24].

We also assume that the super-provider has prior knowledge of the providers' preferences regarding time and energy service attributes. An incentive model is employed to predict the energy available for consumption. For instance, provider 1 in Fig. 9.2(A) wants to offer the energy service $S1$ with 500 mAh at time t_1 . In another example, provider 2 wants to offer $S2$ at time slots t_1 , t_2 , or t_3 . We assume the provider would stay for the full-time slot. We also assume that the provider's service amount is fixed and can be split among multiple time slots. For instance, provider 2 may share part of their service $S2$ on t_1 , e.g., 300 mAh, and the other part at t_2 or t_3 . We also assume that a single energy provider may share their spare energy with multiple energy

consumers within *a specific time*. The super-provider uses *rewards* to encourage providers to share energy. Rewards may come in the form of stored credits to providers. A provider receives a reward based on an incentive model as presented in Chapter 3.

The super-provider will allocate services to time slots to serve as many consumers as possible to *maximize* their quality of experience in the microcell. However, it is challenging to fulfill multiple energy requirements with limited energy services [24]. For example, in Fig. 9.2(A), the total energy demand ($\sum R$) is 3,200 mAh. The total available energy services ($\sum S$) is 2,300 mAh. The available services may fulfill 71.9% of the energy requests, which cannot be fully provisioned with the temporal constraints of services and requests. Therefore, satisfying all consumers with their under-provisioned requests is *more challenging*.

Fig. 9.2(B) presents the outcome of a greedy FCFS, i.e., first come first served, allocation strategy for the available energy [95]. In greedy, the time slots and services will be scheduled based on their start time. For instance, in Fig. 9.2(A), even though $S2$ can be offered in t_1 , t_2 and t_3 , $S1$ will be allocated to t_1 because it comes first in terms of time. The greedy strategy does not leverage the shareable nature of energy services or the providers' flexibility which may affect the energy allocation efficiency and impact the consumers' experience. Therefore, the greedy strategy may not fit well in this context. For example, in Fig. 9.2(B), the greedy-based approach could only fulfill 1,300 mAh from the total demand, equivalent to 40.6%. Moreover, the total number of consumers ($\sum C$) in Fig. 9.2(A) is 14. The greedy approach could offer energy to seven consumers, which is equivalent to 50%. In this context, we consider the size of fulfilled requests and the number of fulfilled consumers in assessing the quality of experience. In this example, using the greedy approach resulted in 45.3% of consumers' QoE.

Allocating the limited available energy with the time constraints of both services and requests represents critical challenges for efficient and QoE-aware provisioning of IoT energy services. We propose a framework to compose the energy services to maximize the consumers' experience. Our framework leverages the providers' *flexibility*

and *shareable* nature of energy services to allocate services and optimize the QoE efficiently.

9.2 Preliminaries

We consider the scenario of energy sharing in a microcell M during a time interval T . T is divided into a set of $\{t_1, \dots, t_n\}$ where t_i is a predefined time period, e.g., one hour. We use the below definitions to formalize the problem.

Definition 1: Energy-as-a-Service (EaaS). We adopt the definition of $EaaS$ in [25]. An energy service ($EaaS$) is a tuple of $\langle E_{id}, E_{pid}, F, Q \rangle$, where E_{id} is an energy service ID, E_{pid} is a provider ID, F is the function of sharing wireless energy, Q is a set of non-functional (QoS) attributes, including:

- p_{ae} is the amount of energy shared by the provider.
- p_{loc} is the location of the energy provider $\langle x, y \rangle$.
- p_t is the set of time intervals $\langle t_s, t_e \rangle$ a provider may offer their energy.

Definition 2: Energy Service Request (ER). We adopt the definition of ER presented in Chapter 3. An ER is a tuple of $\langle E_{id}, E_{cid}, F, QR \rangle$, where ER_{id} is an energy request ID, E_{cid} is a consumer ID, F is the function of receiving energy wirelessly by an IoT device, QR is a set of non-functional attributes, including:

- c_{re} is the amount of requested energy.
- c_{loc} is the location of the energy consumer $\langle x, y \rangle$.
- c_t is the time interval $\langle t_s, t_e \rangle$ of requiring energy.

Definition 3: Microcell Energy Demand $\mathcal{MED}$. $\mathcal{MED}$ is the total amount of requested energy during a time interval T (see Fig. 9.2). T is divided into time slots. We define $\mathcal{MED}$ by aggregating the amount of required energy per time slot. Therefore, the definition of $\mathcal{MED} = \{t_1, t_2, \dots, t_n\}$ where t is a tuple of $\langle d, rwd, re, nc, ER \rangle$. Here d is a predefined time in the time interval of the microcell T , e.g., [9:00 a.m.

-10:00 a.m.]. rw_d is the reward of providing the required energy re . We compute rw_d using the incentive model presented in Chapter 3. We assume the super-provider will use the microcell history to compute the energy demand in advance. nc is the number of consumers in the microcell at time slot t . ER is the set of available requests in the microcell at time slot t .

Definition 4: Quality of Experience (QoE). QoE is an objective function to measure consumers' satisfaction with energy provisioning in a microcell M within a predefined time interval T . The function definition is:

$$QoE(M) = F(T, \mathcal{ES}, \mathcal{MED}) \quad (9.1)$$

where $\mathcal{ES}$ is the set of energy services and $\mathcal{MED}$ is the microcell energy demand.

9.2.1 Problem Definition

Given a set of n energy services $\mathcal{ES} = \{EaaS_1, EaaS_2, \dots, EaaS_n\}$ and a set of m energy requests $\mathcal{ER} = \{ER_1, ER_2, \dots, ER_m\}$ in a microcell M . The super-provider advertises the microcell energy demand $\mathcal{MED}$. Energy providers register their services in terms of (1) the amount of energy p_{ae} , and (2) the time slots t_i to offer their services. The super-provider uses the providers' preferences to allocate their services to time slots. The allocation approach aims to fulfill the maximum requests and maximize the QoE. We formulate the service composition problem to a time-constrained optimization problem as follows:

- Maximize $QoE(M) = F(T, \mathcal{ES}, \mathcal{MED})$,

subject to:

- $t_i.re > 0$ for each $t_i \in T$,
- $EaaS_j.P_t \subset t_i.d$ for each $EaaS_j \in \mathcal{ES}$.

where P_t is the time interval $\langle t_s, t_e \rangle$ a provider of $EaaS_j$ may offer their energy, $t_i.d$ is the duration of a time slot i in the time interval of the microcell T , and $t_i.re$ is the required energy re at time slot i .

The composition's goal is to allocate the available energy services to time slots efficiently. The objective function attempts to assign energy services according to their spatio-temporal features optimally, providers' preferences, and required energy in time slots. The spatial aspect of energy service focuses on a geographical cell. The temporal aspect focuses on the times of energy service provisioning.

We use the following assumptions to formulate the problem:

- Providers' energy size is fixed during composition.
- Providers are available in all their selected time slots.
- Providers may offer partial services to multiple consumers simultaneously.
- Consumers' time windows do not overlap with time slots.
- Providers and consumers have fixed locations during energy sharing.
- The microcell has *multiple* providers and *multiple* consumers.
- There is no energy loss in sharing. As the technology matures, we anticipate that the devices will be able to share more energy, and the energy loss of sharing will become minimal [24].
- The exact amount of required energy for a microcell is given [164].
- A reward system incentivizes providers to offer their service.
- A trust framework is used to preserve the privacy of IoT devices [49].

9.3 Quality of Experience Model

The Quality of Experience (QoE) in a microcell is measured based on the *number of satisfied consumers* and the *amount of fulfilled requests*. Recall that the time interval of the microcell is divided into time slots. Therefore, QoE for each time slot t_i will be computed using the following attributes:

- **Satisfaction Ratio:** We define the Satisfaction Ratio ($\mathcal{SR}$) as the number of consumers who received their requested energy or part of it. We compute $\mathcal{SR}$ per time slot t as follows:

$$\mathcal{SR} = \frac{|\{ER \in \mathcal{ER} \mid ER \text{ is completed} \ \& \ c_t \in d\}|}{|\mathcal{ER}|} \quad (9.2)$$

where $\mathcal{ER}$ is the set of all requests in time slot t , $|\cdot|$ is the cardinality of the set, c_t is the request time, and d is the time duration of t .

- **Fulfillment Ratio:** The satisfaction ratio is not enough to measure QoE. For example, if we have a set of energy requests in mAh $\mathcal{ER} = \{10, 20, 20, 70\}$, serving the first three consumers is not equal to serving the last three due to the different amount of requested energy. Therefore, we define the Fulfillment Ratio ($\mathcal{FR}$) based on the percentage of fulfillment for each request. We compute $\mathcal{FR}$ per time slot t as follows:

$$\mathcal{FR} = \sum_{i=1}^n \left(w_i \times \frac{\text{Received_Energy}_i}{\text{Requested_Energy}_i} \right) \quad (9.3)$$

where n is the number of all energy requests in t , and w_i is the weight of the request over the total amount of requested energy in t .

Quality of Experience: As previously stated, We define the QoE in a microcell based on the satisfaction ratio $\mathcal{SR}$ and fulfillment ratio $\mathcal{FR}$ of each time slot t . Therefore, we compute the $QoE(M)$ as the following:

$$QoE(M) = \alpha \times \left(\sum_{i=1}^m \mathcal{SR}_i \times \beta_i \right) + (1 - \alpha) \times \left(\sum_{i=1}^m \mathcal{FR}_i \times \gamma_i \right) \quad (9.4)$$

where m is the number of time slots in the microcell's time interval T . $\mathcal{SR}_i$ is the satisfaction ratio of a time slot computed by Eq. 9.2. β_i is the weight of a time slot t_i , which is its number of consumers over the total number of consumers in T . $\mathcal{FR}_i$ is the ratio of the fulfillment of the time slot computed by Eq. 9.3. γ_i is the weight of a time slot t_i , which is its total required energy over the total required energy in T . α is

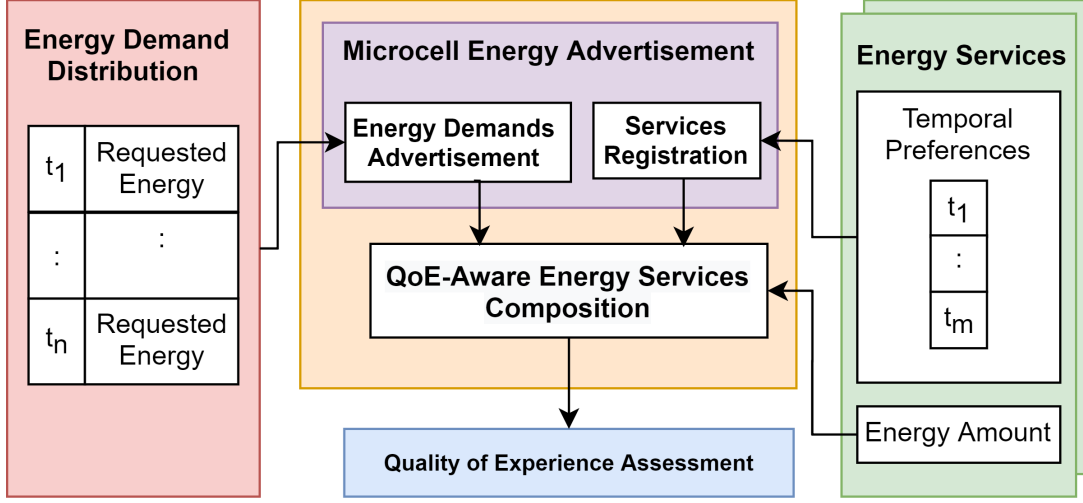


FIGURE 9.3. Quality of experience driven service composition framework

a user-defined weight between zero and one to define the weight of SR_i and FR_i in QoE .

9.4 Quality of Experience Framework

We introduce a quality of experience composition framework for managing energy services to enhance consumers' QoE (see Fig. 9.3). The framework is divided into three phases: (1) Microcell energy advertisement, (2) Composing energy services, and (3) Quality of experience assessment. In the first phase, the super-provider will advertise the microcell's energy demand and receive providers' preferences. In the second phase, the super-provider will compose energy services to maximize the QoE. In the last phase, the super-provider will assess the QoE for the resulting composition.

9.4.1 Microcell Energy Demand Advertisement

In this phase, the super-provider computes the reward for each time slot based on the amount of required energy using the incentive model in Chapter 3. Then, the system will announce the required energy and rewards for the whole microcell using

Definition 3. Energy providers will register based on their preferences regarding their energy amount and the time slots during which they will be available (see Fig. 9.2).

9.4.2 Energy Services Composition

This phase aims to compose energy services to maximize the QoE. We propose two heuristic approaches to compose energy services: *Partial-Based* (PB) and *Demand-Based* (DB). The PB composition is inspired by the FCFS resource allocation algorithm [95]. The PB approach splits services among consumers if the required energy is more than the available energy. Intuitively, offering part of the services will satisfy more consumers than offering it to some of them. The DB composition is inspired by the priority allocation algorithm [95]. The DB approach extends the PB approach by prioritizing slots with the highest demand to ensure service availability. We discuss each approach below.

9.4.2.1 Partial-based Energy Services Composition

The Partial-Based (PB) composition aims at maximizing the QoE by composing services for each time slot based on the first come first served approach. For example, if a provider offers services in two-time slots, the algorithm will assign the service to the earlier time slot. If the time slot does not need the service, the service will be assigned to the next time slot. Moreover, PB chunks services between energy consumers if the available energy services are insufficient to fulfill the total required energy in the time slot. Intuitively, offering part of the required services to all consumers is more satisfying than offering it to some only.

Algorithm 9.1 presents the PB service composition. For every time slot t_i , the algorithm retrieves the required energy amount (Line 3). Then, for each registered service es in t , the algorithm keeps adding services to the selected services until the required energy is fulfilled or all the available services have been selected (Lines 4-13). Note that if a service is partially needed, then the service available amount will be

Algorithm 9.1 Partial-based Composition of Services**Input:** $\mathcal{MED}$, $\mathcal{ES}$, $threshold$ **Output:** $energy_comp$

```

1: for  $t_i$  in  $\mathcal{MED}$  do
2:    $selectedES = \{\}$ 
3:    $demand = t_i.re$ 
4:   for  $es_j$  in  $t_i.\mathcal{ES}$  do
5:     if  $demand > 0$  then
6:        $demand = demand - es_j.p_{ae}$ 
7:        $energy\_comp.add(t_i, es_j)$ 
8:        $selectedES.add(es_j)$ 
9:     if  $demand < 0$  then
10:       $es_j.p_{ae} = demand * -1$ 
11:       $demand = 0$ 
12:     else
13:       Remove_Service( $es_j, \mathcal{MED}$ )
14:   if  $demand = 0$  then
15:     Assign_Energy( $t_i.\mathcal{ER}, selectedES$ )
16:   else
17:     Assign_Partial_Energy( $t_i.\mathcal{ER}, selectedES, nc, threshold$ )
18: return  $energy\_comp$ 

```

updated to be used by other registered time slots (Lines 9-11). Moreover, if a time slot fully uses a service, it will be removed from other registered time slots (Lines 12-13). After processing all services, if the energy demand of the slot is zero, the algorithm assigns the selected services to requests (Lines 14-15). If the energy demand is not fulfilled, the algorithm distributes the available services among available requests (Line 17). If the service chunks are smaller than the threshold, consumers will be removed, and the service will be shared among the rest. The threshold prevents dividing services into small neglectable chunks. The composition of the selected services will be returned in Line 18.

9.4.2.2 Demand-based Energy Services Composition

The Demand-Based (DB) composition goal is to maximize QoE by prioritizing time slots with higher energy demand. The intuitive idea of the DB approach is that high-demanding time slots will require more services. Thus, services should be assigned to them before less demanding time slots, which may ensure a better distribution of

Algorithm 9.2 Demand-based Composition of Services**Input:** $\mathcal{MED}$, $\mathcal{ES}$, $threshold$ **Output:** $energy_comp$

```

1:  $SMED = \text{sort}(\mathcal{MED}, nc : \text{descending}, re : \text{descending})$ 
2: for  $t_i$  in  $SMED$  do
3:    $selectedES = \{\}$ 
4:    $demand = t_i.re$ ;
5:    $sortedES = \text{sort}(t_i.\mathcal{ES}, nt : \text{ascending})$ 
6:   for  $es_j$  in  $sortedES$  do
7:     if  $demand > 0$  then
8:        $demand = demand - es_j.p_{ae}$ 
9:        $energy\_comp.add(t_i, es_j)$ 
10:       $selectedES.add(es_j)$ 
11:     if  $demand < 0$  then
12:        $es_j.p_{ae} = demand * -1$ 
13:        $demand = 0$ 
14:     else
15:       Remove_Service( $es_j, SMED$ )
16:   if  $demand = 0$  then
17:     Assign_Energy( $t_i.\mathcal{ER}, selectedES$ )
18:   else
19:     Assign_Partial_Energy( $t_i.\mathcal{ER}, selectedES, nc, threshold$ )
20: return  $energy\_comp$ 

```

the available services. For instance, if a provider offers their service in two-time slots, the algorithm will assign the service to the more demanding time slot. If that time slot does not need the service, the service will be assigned to the next time slot. This indicates that the order of time slots in composing services matters because, if a service is used in a time slot, it will be removed from others. Removing a service from a time slot may affect the amount of available energy and, thus, the number of served and satisfied consumers. Moreover, the DB approach maximizes the QoE by chunking services between energy consumers if the available services are insufficient.

Algorithm 9.2 presents the DB service composition. The algorithm starts by sorting the time slots in a descending order based on the number of consumers nc , then the amount of requested energy re (Line 1). The goal of sorting is to start composing services for the most demanding time slots. As some services may be registered in multiple services, using these services for the most demanding time slots may offer a better experience. Line 4 retrieves the total required energy for each time slot t_i . Then,

for every time slot, the registered services will be sorted in ascending order based on the number of time slots in which a service was registered. This sort will allow us to start with the least-connected services. In other words, using such services may impact a fewer number of time slots than using services registered in many time slots. Then, for each registered service es in t , the algorithm keeps adding services to the selected services until the required energy is fulfilled or all the available services have been selected (Lines 6-15). Similar to the PB approach, if a service is partially needed, the available service amount will be updated to be used by other registered time slots (Lines 11-13). Moreover, if a time slot fully uses a service, it will be removed from other registered time slots (Lines 14-15). After processing all services, if the energy demand of the slot is zero, the algorithm assigns the selected services to requests (Lines 16-17). If the energy demand is not fulfilled, the algorithm distributes the available services among available requests (Line 19). If the service chunks are smaller than the threshold, consumers will be removed, and the service will be shared among the rest. The threshold prevents dividing services into small neglectable chunks. Line 20 returns the composition of the selected services.

9.4.3 Assessing Quality of Experience

The super-provider assesses the QoE of each proposed composition in this phase. The QoE is computed using the model discussed in Section 9.3. The assessment of QoE gives an indicator of consumers' satisfaction in the microcell.

9.5 Evaluation

We compare the proposed composition approaches, Partial-Based composition (PB), and Demand-Based Composition (DB), with the resource allocation algorithms, namely, first come first served allocation (*Greedy*), and Max-Min Fair allocation (*Max-Min*) [95][24]. The *Greedy* approach is a modified FCFS algorithm where the time slots

and services will be scheduled based on their start time. The *Max-Min* is a modified Max-Min Fair allocation where services that can be offered in multiple time slots will be split among these time slots using the Max-Min technique. We evaluate each approach's *effectiveness* and *efficiency*.

9.5.1 Dataset Description

We used a real dataset from the developed app in [5]. The dataset consists of energy transfer records between a provider (smartphone) and a consumer (smartphone). The records attributes are the provider ID, consumer ID, transaction date, time, energy services and requests' amount, and transfer duration. We used the energy dataset to generate the QoS parameters for the energy services and requests. For instance, the amount of a wireless charging transfer in mAh defines the amount of requested/provided energy. In addition, the energy dataset records of a wireless charging transfer duration define the end time of each request/service.

We augmented the energy sharing dataset to mimic the crowd's behavior within microcells by utilizing a dataset published by IBM for a coffee shop chain with three branches in New York City². The dataset consists of transaction records of customers' purchases in each coffee shop for one month. Each coffee shop consists of, on average, 560 transnational records per day and 16,500 transaction records. We use the IBM dataset to simulate the spatio-temporal features of energy services and requests. Our experiment uses the consumer ID, transaction date, time, location, and coffee shop ID from each record in the dataset to define the spatio-temporal features of energy services and requests, e.g., start and location of energy service or a request. We ran a total of 7,000 experiments with six time slots. Each time slot was an hour long. The providers' temporal provision preferences were registered randomly to 1-3 time slots in each run. In addition, the number of services and requests varied between 300 to 2,000 per run, depending on the experiments' setting. For each run, we used the proposed approaches

²<https://ibm.co/2O7IvxJ>

to compose energy services. We then measured the QoE for each composition. Table 9.1 presents the parameters of the experiments.

9.5.2 Evaluation of the Composition Framework

We ran six experiments to determine the effectiveness and efficiency of the proposed approaches. The experiments evaluated the approaches in terms of their satisfaction rate, fulfillment rate, quality of experience, the impact of thresholds, and computation cost. We ran the approaches in different settings by changing the ratio of services to requests in the time interval T . We gradually increased the ratio from 15% to 90%. We repeated the experiment 1,000 times at each point and considered the average value for each approach.

9.5.2.1 Quality of Experience Evaluation

As previously stated, we computed the QoE based on $\mathcal{SR}$ and $\mathcal{FR}$ (see Section 9.3). In this subsection, we study the impact of each ratio; then, we evaluate the QoE.

The first experiment compared the $\mathcal{SR}$ of the proposed approaches PB and DB, against Greedy and Max-Min. As previously stated, $\mathcal{SR}$ represents the number of consumers who received energy, whether fully or partially. Therefore, a high $\mathcal{SR}$ of a composition ensures a higher number of satisfied consumers and, thereby, a better QoE. The $\mathcal{SR}$ of a time slot was computed using Eq. 9.2 and then averaged for the microcell similar to the first part of Eq. 9.4. Fig. 9.4 presents the average $\mathcal{SR}$ in the microcell for each approach. The x-axes in Figs 9.4-9.7 represent the ratio of the number of energy services to requests. In Fig. 9.4, the $\mathcal{SR}$ increases when the number of available services increases for all the composition approaches. For instance, when the ratio of services to requests is 80%, all approaches provide a higher $\mathcal{SR}$ compared to when the ratio is 20%. This observation can be explained by the availability of services to offer energy. The more services available, the more requests that can be fulfilled. The proposed approach PB performs better than Greedy as it splits the available energy

TABLE 9.1. Experiments variables

Variables	Value
Energy dataset for coffee shop 8 in April	16,830
Number of services & requests	[300-2,000]/run
Number of time slots	6
Provided energy	5 - 100%
Requested energy	5 - 100%
Time interval	6 hours
Service registration	[1-3] time slots / service

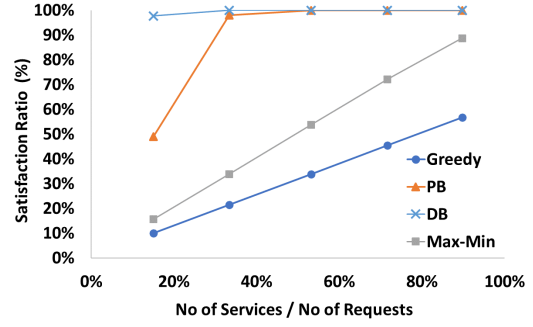


FIGURE 9.4. The average of satisfaction ratio

between the consumers as partial services, unlike the Greedy approach, which fulfills a request fully before serving the next request. For the same reason, PB also performs better than Max-Min. Even though Max-Min has a better energy utilization by splitting energy services fairly between time slots (see Fig. 9.5), a fair distribution of energy does not necessarily result in equally satisfied consumers. This is due to the different energy requirements of consumers. In addition, the proposed approach DB gives the best results as it prioritizes the time slots with the highest demand regarding the number of consumers and the required energy. Recall that the order of time slots in composing services is crucial because if a service is used in a time slot, it will be removed from others. Removing a service from a time slot may affect the amount of available energy and, thus, the number of served consumers. Prioritizing the most demanding time slots allows DB to have more services to use. It increases $\mathcal{SR}$ by increasing the number of fulfilled consumers.

The second experiment compared each approach's $\mathcal{FR}$. As previously stated, $\mathcal{FR}$ presents the fulfillment rate for each request. Therefore, a high $\mathcal{FR}$ of a composition ensures higher consumer satisfaction and, thereby, a better QoE. The $\mathcal{FR}$ of a time slot was computed using Eq. 9.3 and then averaged for the microcell similar to the second part of Eq. 9.4. Fig. 9.6 represents each approach's average $\mathcal{FR}$ in the microcell. In Fig. 9.6, the $\mathcal{FR}$ increases when the number of available services increases for all the approaches. This observation can be explained by the availability of services to offer energy. PB performs similar to Greedy in terms of $\mathcal{FR}$. This is an expected behavior,

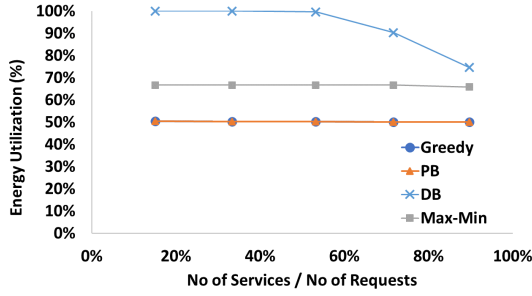


FIGURE 9.5. The average of energy utilization

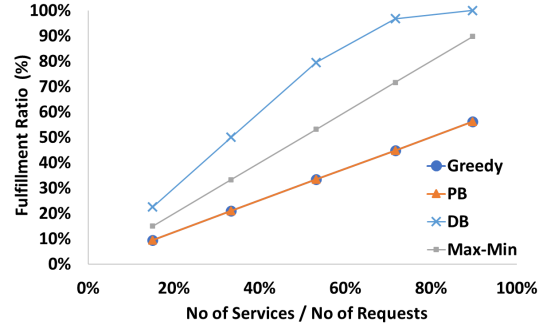


FIGURE 9.6. The average of fulfillment ratio

since both approaches start with the same time slots and, therefore, have the same set of available services. The difference between both approaches is in how they share energy among consumers, i.e., complete services in Greedy and partial services in PB. Moreover, Max-Min has a better $\mathcal{FR}$ because it has better energy utilization (see Fig. 9.5). Higher energy utilization is achieved by splitting energy services fairly between time slots. DB gives the best results as it prioritizes the time slots that have the highest demand, as discussed in the previous experiment.

The third experiment compared the QoE using all approaches. As previously stated, the QoE presents consumers' overall satisfaction across time. Therefore, a high QoE of a composition indicates a higher level of satisfaction for consumers. The QoE is computed using Eq. 9.4. Note that we used $\alpha = 0.5$ to give equal weight for both $\mathcal{SR}$ and $\mathcal{FR}$. Fig. 9.7 presents the average QoE using each approach. In Fig. 9.7, similar to the previous experiments, the QoE increases when the availability of services increases. PB approach performed better than Greedy in terms of QoE due to its higher $\mathcal{SR}$ as discussed in the first experiment. Additionally, PB performed better than Max-Min when the number of energy services was less than the request. This is because PB will satisfy more consumers in a limited resources environment (higher $\mathcal{SR}$) by partially fulfilling their requests. However, with enough services, Max-Min will better utilize the energy to fulfill requests (higher $\mathcal{FR}$). Moreover, the DB approach gives the best results due to its higher $\mathcal{SR}$ and $\mathcal{FR}$.

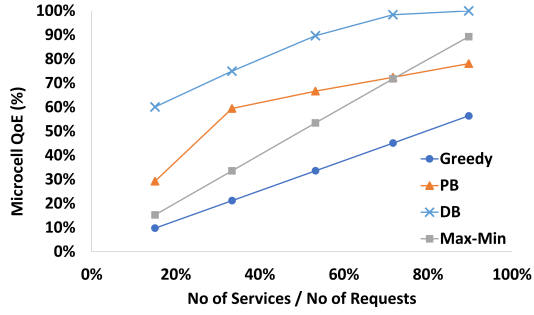


FIGURE 9.7. The average of quality of experience

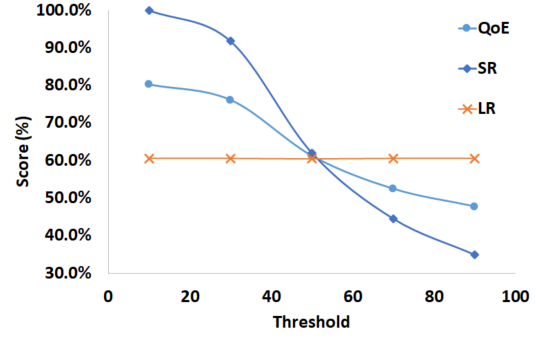


FIGURE 9.8. The average of $\{QoE, SR, FR\}$ using PB composition

9.5.2.2 Threshold Impact Evaluation

The following two experiments studied the impact of thresholds on the PB approach. Recall that PB and DB approaches split energy between consumers based on a defined threshold. The threshold prevents dividing services into small neglectable chunks. The experiments of both PB and DB gave the same behavior. Thus, we are only presenting the results of PB.

Fig. 9.8 represents the impact of the threshold on the three previously tested attributes: SR , FR , and QoE . We tested the PB approach with a 99% ratio of services to requests. The x-axis in Fig. 9.8 represents the threshold of partial services. FR does not change as the threshold increases because it relies on the order of time slots and not the distribution size (threshold), as discussed in the previous experiment. Also, both SR and QoE decrease as the threshold increases due to the thresholds' size. When the threshold's size increases, fewer consumers will be served. A lower number of fulfilled consumers results in low SR and thereby a low QoE .

The fifth experiment compared the impact of the threshold on the QoE with different ratios of services to requests. We tested the PB approach with thresholds of $\{10, 30, 50, 70, 90\}$. In Fig. 9.9, the QoE increases when the number of available services increases for all threshold values. Additionally, the QoE for threshold 10 is the highest among all due to the threshold's size. When the threshold size is small, more consumers will

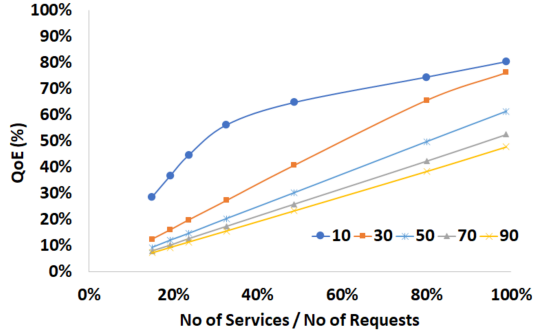


FIGURE 9.9. The average of QoE using PB composition with various thresholds in a microcell

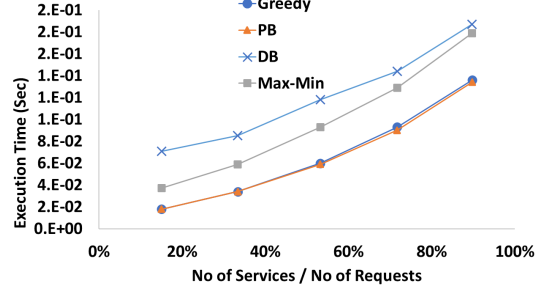


FIGURE 9.10. The average execution time of all compositions

be served. A higher number of fulfilled consumers results in high $\mathcal{SR}$ and, thereby, a high QoE.

9.5.2.3 Computation Efficiency Evaluation

The execution time for all approaches increases with the availability of services (see Fig. 9.10). This is due to the increase in processing time to assign these services.

9.6 Conclusion

We proposed an energy service composition framework that evaluates QoE in a microcell. A new QoE based-assessment was proposed to capture the overall satisfaction across consumers over a period of time. A two QoE-driven compositions of energy service were proposed. The Partial-Based (PB) approach uses partial services to maximize the number of satisfied consumers and thereby increase the QoE. The Demand-Based (DB) approach uses partial services and prioritizes the most demanding time slots to maximize the number of satisfied consumers and their level of fulfillment and thereby increase the QoE. Experimental results show that DB outperforms all the evaluated approaches. The efficiency of the proposed approaches was investigated against

a Greedy approach. A future direction may consider the probability of change in the microcell energy demand.

Preference-aware Energy Service Composition

10.1 Introduction

Chapter 9 presented the use of energy services as a tool to provide customers with the best quality of experience when visiting the business. A *Quality of Experience (QoE)* model was proposed as a metric to capture the customers' satisfaction level according to the super-provider preferences. In the energy services context, consumer satisfaction is defined as *receiving the requested energy*. In this chapter, we propose a QoE-aware energy provisioning framework. The framework requires prior knowledge of providers' spatio-temporal preferences and the super-provider's time-based business model. The concept of CP-Nets captures the energy providers' temporal preferences and flexibility [173]. We leverage the providers' preferences and flexibility to cope with the time constraints of the energy demands. The super-provider maps the existing energy demands to the available providers according to their preferences. A better allocation of energy services would be reflected in a better quality of experience for consumers. Sometimes, however, assigning providers to energy demands at their least preferred time would require a higher incentive [96][1]. The super-provider aims to maximize the quality of experience by fulfilling all the energy demands with the least cost according to the providers' availability and preferences. The QoE-aware energy-sharing framework employs *importance-based* and *heuristic-based* strategies. The importance-based approach defines the temporal priorities from a super-provider perspective. The heuristic-based approach combines the energy distribution demand and the providers' temporal preferences. The main contributions of this chapter are:

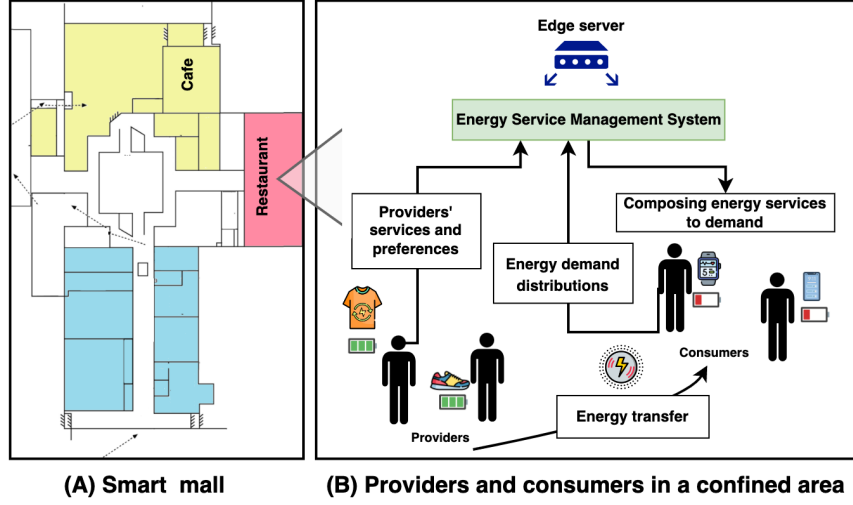


FIGURE 10.1. IoT energy services environment

- A CP-Net-based provisioning preference model for energy providers.
- A novel Quality of Experience (QoE) model for crowdsourced energy services.
- A QoE-aware framework for composing energy services.
- An importance-based composition approach to select the best composition of energy services that maximize the QoE.
- A multi-objective heuristic-based composition approach to maximize the QoE and minimize the cost of incentives.

10.1.1 Motivating Scenario

We describe a scenario in a confined place, i.e., a microcell, where people congregate, e.g., cafes, restaurants, and study spaces (see Fig. 10.1(A)). Each microcell may have several IoT devices acting as energy providers or consumers (see Fig. 10.1(B)). The super-provider aims to leverage the crowdsourced energy services to enhance the consumers' *experience*. We assume all local energy services and requests are submitted and managed at the *edge*, e.g., a router in the microcell (see Fig. 10.1(B)). We assume the super-provider has prior knowledge of the *Energy Demand Distribution* ($\mathcal{EDD}$) in the microcell over a period of time (T) (see Fig. 10.2(A)). The $\mathcal{EDD}$ is represented

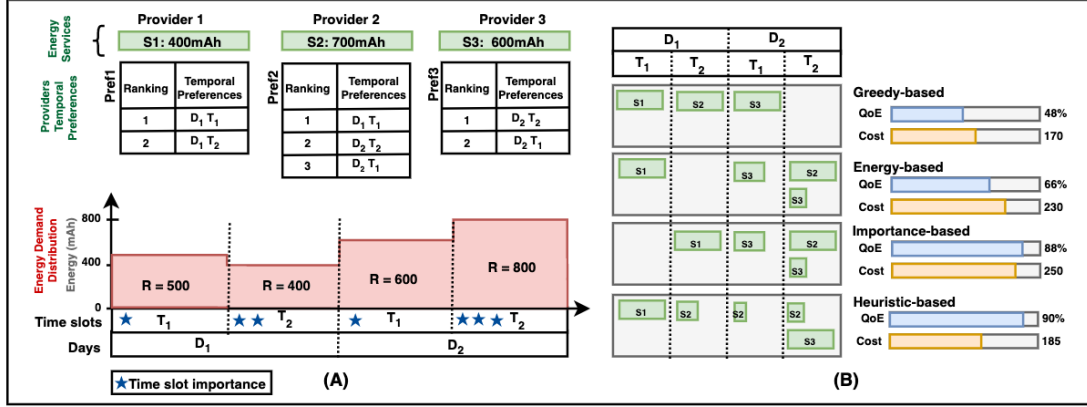


FIGURE 10.2. (A) Microcell energy demand and providers temporal preferences (B) Energy provisioning approaches

by the requested energy in each time slot, e.g., 500 mAh at time slot T_1 in day D_1 . Additionally, $\mathcal{EDD}$ is represented in terms of the importance of the time slot based on the super-provider's business model. For instance, T_2 in day D_1 has two stars because of its importance to the business, e.g., lunch hour.

We also assume that the super-provider has prior knowledge of the providers' preferences in terms of time and energy service attributes. For instance, provider 1 in Fig. 10.2(A) wants to offer the energy service $S1$ with 400 mAh. Also, provider 1 prefers to offer $S1$ on the first day D_1 at time slot T_1 rather than time slot T_2 ($Pref1$ in Fig. 10.2(A)). Another example, Provider 3 prefers to offer $S3$ on the second day D_2 at time slot T_2 rather than time slot T_1 ($Pref3$ in Fig. 10.2(A)). We assume the provider would stay for the full-time slot. We also assume that the provider's service amount is fixed and can be split among multiple time slots. For instance, provider 1 may share part of their service $S1$ on T_1 , e.g., 300 mAh, and the other part at T_2 . The super-provider uses *rewards* to encourage providers to share energy. Rewards may come in the form of stored credits to providers. A provider receives a reward based on an incentive model [1]. A provider's reward gets impacted by their temporal preferences, e.g., a less preferred time would require a higher incentive.

The super-provider will allocate services to time slots to serve as many consumers as possible to increase their quality of experience in the microcell. However, fulfilling

multiple energy requirements with limited energy services is challenging. Moreover, the super-provider's business model may require prioritizing time slots regardless of their energy demand because of their importance, e.g., T_2 in D_1 is more important than T_1 . In addition, the super-provider needs to consider the temporal preferences of the providers while allocating the services. As the super-provider will endure the rewards cost, assigning providers to their most preferred time will minimize that cost.

Fig. 10.2(B) presents the outcome of different allocation strategies for the available energy. We aim to efficiently allocate energy services to time slots while *maximizing* the QoE and *minimizing* the cost. The first allocation approach follows a greedy FCFS, i.e., first come first served, composition of services strategy [95]. The time slots and services will be processed in Greedy based on their start time. For instance, in Fig. 10.2(A), even though $S1$ can be offered in T_1 and T_2 , $S1$ will be allocated to T_1 because it comes first in terms of time. The Energy-based strategy in Fig. 10.2(B) follows a modified priority-based allocation composition. This approach prioritizes assigning flexible services to the most demanding time slots. For example, the Energy-based approach starts by assigning services to T_2 in D_2 as it has the highest demand (800 mAh). Both strategies may not be a good fit in this context as they do not consider the importance of the time slot or leverage providers' flexibility, which may affect the energy allocation efficiency and impact the consumers' experience. For example, in Fig. 10.2(B), the greedy-based approach could achieve only 48% of enhancing consumers' experience. The energy-demand approach achieved 66% of enhancing consumers' experience while costing 230 reward points. The importance-based and heuristic-based allocation strategies are a good illustration of the effect of a QoE-aware provisioning plan. Both strategies achieved 88% and 90% of enhancing the QoE. Additionally, the heuristic approach costs 185 reward points, which is comparatively lower than other strategies.

Allocating the limited available energy with the time constraints of both services and requests represents critical challenges for efficient and QoE-aware provisioning of IoT energy services. We reformulate our service provisioning problem as a multi-objective

time-constrained resource allocation problem, i.e., (i) maximizing the QoE by efficiently allocating the available energy, and (ii) minimizing the cost of incentivizing the providers to offer their energy. We propose a framework that will compose the best energy services to enhance the consumers' experience with respect to the super-provider's business goals. Our framework leverages energy providers that offer services on *multiple* time slots by assigning them to the most important slots while considering the cost of their preferences. Considering providers' temporal *flexibility* may improve the services' distribution, fulfilling more energy requests and offering a better experience.

10.2 System Model

We first adopt the definitions of IoT energy services and requests from [23]. We then introduce the formal presentation of the energy demand distribution, providers' flexible temporal preferences, and super-provider preferences. We also introduce the concept of the quality of experience in a microcell. This work considers a provisioning framework for stationary services and requests to focus only on the temporal constraints of allocating energy to multiple time slots in a microcell within a predefined time interval. The goal is to ensure the best consumer experience over a predefined time while minimizing costs and meeting the super-provider's business goals.

10.2.1 Preliminaries

DEFINITION 1. *Energy Service (S)* is a tuple of $\langle id, pid, F, Q \rangle$ where:

- *id* is the service ID,
- *pid* is the provider ID,
- *F* is the function of sharing wireless energy,
- *Q* is a tuple of non-functional attributes $\langle ae, loc \rangle$ where:
 - *ae* is the amount of energy shared by the provider.

- loc is the location of the energy provider $\langle x, y \rangle$.

DEFINITION 2. **Energy Request (R)** is a tuple of $\langle id, cid, F, Q \rangle$, where:

- id is the request ID,
- cid is the consumer ID,
- F is the function of receiving wireless energy,
- Q is a tuple of non-functional attributes $\langle ae, loc, st, et \rangle$ where:
 - re is the amount of energy required by the consumer.
 - loc is the location of the energy consumer $\langle x, y \rangle$.
 - st is the start time of the energy request.
 - et is the end time of the energy request.

DEFINITION 3. **Energy Demand Distribution ($\mathcal{EDD}$)** is the captured consumers' energy demand distribution in the microcell $\mathcal{M}$ over a period of time $\mathcal{T}$.

The $\mathcal{EDD}$ of a microcell is represented as a tuple of $\langle ED_1, \dots, ED_n \rangle$ where ED_i is a tuple of $\langle t_i, r_i \rangle$. Here, t_i is a time slot in the time interval $\mathcal{T}$, e.g., [9:00 a.m. -10:00 a.m.], r_i is the aggregation of the required energy in t_i .

10.2.2 Providers' Flexible Temporal Provisioning Preferences

Providers may be flexible when provisioning their service, i.e., they can come at multiple time slots. However, they may prefer certain times over others, e.g., a provider may prefer to come in the morning rather than the evening (see Fig. 10.3). As previously mentioned, assigning the providers to a more preferred time slot will require fewer rewards. Thus, ranking providers' temporal preferences may help the super-provider reduce the incentives they have to pay.

DEFINITION 4. **Provider Temporal Preferences (PTP)** is a set of provider's ranked temporal provisioning preferences.

Formally, the provider temporal preferences PTP_i is a tuple $\langle t_i, k_i \rangle$ where:

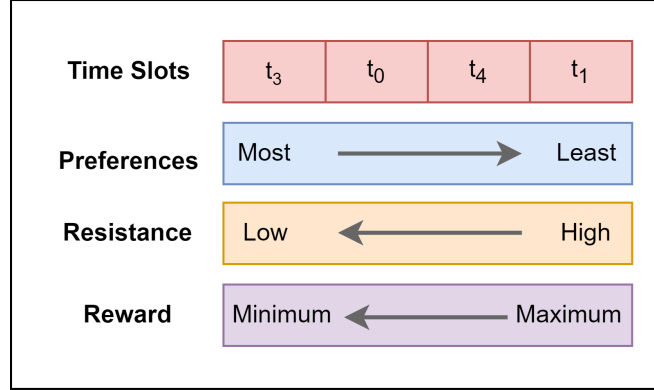


FIGURE 10.3. a Provider's temporal preferences

- $t_i = \{t_1, \dots, t_n\}$ is a set of time slots where the provider may offer their service.
- $k_i = \{k_1, \dots, k_n\}$ is a set of preferences ranking for the time slots t_i .

10.2.3 Super-Provider Preferences

The super-provider uses the revenue and traffic history of the microcell M to define five preferences for the microcell M in terms of (1) Service management Time interval ($\mathcal{T}$), (2) a *Business Model* (BM), (3) an *Incentive Model* (IM), (4) a Quality of Experience model (QoE), and (5) a Cost ($\mathcal{C}$) function. In what follows, we present each of these preferences.

- **Service management Time interval ($\mathcal{T}$):** $\mathcal{T}$ is the time scale for managing the services provisioning, i.e., day, week, etc. For example, a super-provider may want to focus on a certain week, e.g., back-to-school week, or a certain day, e.g., Black Friday. $\mathcal{T}$ is represented as a sequence of time slots $\langle t_1, t_2, \dots, t_n \rangle$ where t_i is a sub-interval in the time interval $\mathcal{T}$, e.g., [9:00 a.m. -10:00 a.m.].
- **Business Model (BM):** A BM is a series of $\langle I_1, I_2, \dots, I_n \rangle$ defined over $\mathcal{T}$ where I_i represents the importance of time slot t_i . The importance weight I_i of a time slot is defined by the foot traffic and the revenue made during t_i .

The value of I varies from zero to one. The importance score will drive the allocation of energy services.

- **Incentive Model (IM):** *IM* is an incentive model designed to entice providers to offer their energy. The *IM* determines the reward rwd earned by an energy provider for sharing energy. Rewards rwd are granted by the IoT coordinator in the form of stored credits. We assume the system limits the amount of requested energy. The earned credits may be used later by the provider to increase the limit of requested energy as a consumer. A study on using incentives to increase energy provision participation has been conducted [1]. The study shows that the amount of provided energy and the time of provisioning impact the value of the reward. In our context, the super-provider defines its *IM* based on their budget. As mentioned, providers may have different ranked preferences to offer their services. Thus, a super-provider will pay a higher reward to providers assigned to a less preferable time slot. For example, in Fig. 10.3, the provider prefers to offer their service in time slot t_3 and slightly prefers t_1 . Consequently, the provider resistance will be higher in t_1 . Thus, the super-provider will pay a higher reward if they allocate the provider to t_1 . Therefore, a provider reward may be computed as the following:

$$rwd(S_i, t_j) = S.ae_i / rate * credit * PrefRank(t_j) \quad (10.1)$$

where rwd is the reward of providing the energy service S_i by provider P_i at the time slot t_j , $S.ae$ is the amount of energy shared by the provider, $credit$ is the reward value determined by the super-provider, and $PrefRank$ is a function to retrieve the provider's temporal preference rank a time slot t_j . $PrefRank$ value varies from one to n where one is the most preferred time slot, and n is the least preferred time slot. Note that the less preferred time slots will require a higher reward.

- **Quality of Experience (QoE) Model:** We define the QoE for a microcell M within a time interval $\mathcal{T}$ as an objective function to measure the aggregated satisfaction of energy consumers with respect to the super-provider's business

model BM . The consumers' satisfaction is determined by the allocated services to the energy demand distribution $\mathcal{EDD}$. The QoE function definition is:

$$QoE(\mathcal{T}, BM, \mathcal{EDD}, \mathcal{STR}) = \left(\sum_{i=1}^n (\mathcal{STR}.Al_i / \mathcal{EDD}.r_i * e^{BM.I_i}) \right) / n \quad (10.2)$$

where $\mathcal{STR}$ is the strategy of allocating services to time slots, n is the number of time slots, $\mathcal{STR}.Al_i$ is the allocated services for the time slot t_i , $\mathcal{EDD}.r_i$ is the aggregated required energy in t_i , and $BM.I_i$ is the importance weight of t_i . We used an exponential function to emphasize the impact of the importance.

- **Cost ($\mathcal{C}$):** $\mathcal{C}$ is the total amount of rewards a super-provider will pay to the providers based on the service allocation strategy ($\mathcal{STR}$).

$$\mathcal{C}(\mathcal{STR}) = \sum_{i=1}^n rwd(P_i, t_j) \quad (10.3)$$

where $rwd(P_i, t_j)$ is the function to compute the reward of provider P_i for sharing their service at time slot t_j . rwd is computed using Eq. 10.1. Note that t_j will be assigned based on the super-provider strategy $\mathcal{STR}$.

10.2.4 Problem Definition

Given a microcell energy demand distribution $\mathcal{EDD} = \langle ED_1, ED_2, \dots, ED_m \rangle$ over the temporal preferences of the super-provider in time interval $\mathcal{T} = \langle t_1, t_2, \dots, t_m \rangle$, a super-provider's business model $BM = \langle I_1, I_2, \dots, I_m \rangle$ over the same time interval $\mathcal{T}$, a super-provider's incentive model IM and set of n energy services $\mathcal{ES} = \{S_1, S_2, \dots, S_n\}$ where each service may have a set of its provider's temporal preferences $\mathcal{PTP} = \langle t, k \rangle$. Energy providers register their services in terms of: (1) the amount of energy $S.ae$, and (2) their temporal provision preferences $\mathcal{PTP}_i$. The super-provider will use their business model BM , their incentive model IM , and the

providers' temporal preferences $\mathcal{PTP}$, to allocate energy services to time slots. The allocation approach aims at maximizing the QoE by fulfilling the maximum energy demand $\mathcal{EDD}$ while minimizing the cost of incentives C with respect to the BM . We transform the problem of quality-aware energy provisioning into a time-constrained resource allocation problem as follows:

$$\begin{aligned}
 &\text{Maximize} \quad QoE(BM, \mathcal{EDD}, STR, \mathcal{T}) \\
 &\text{Minimize} \quad \mathcal{C}(STR) = \sum_{i=1}^n rwd(P_i, t_j) \\
 &\text{Subject to} \quad t_i \subset S_i.t_i \\
 &\text{Where} \quad \forall S_i \in \mathcal{ES}, \\
 &\quad S_i.t_i \text{ is the start and end of time slot } i
 \end{aligned} \tag{10.4}$$

The strategy's goal is to efficiently allocate the available energy services to time slots. The objective function attempts to optimally assign energy services according to their spatio-temporal features, providers' preferences, the business model, and the required energy in the time slots.

We use the following assumptions to formulate the problem:

- Providers' energy size is fixed during composition.
- Providers may offer their services partially in multiple time slots if they have enough energy.
- Providers and consumers have fixed locations during energy sharing.
- The microcell business model BM , incentive model IM , and energy demand distribution $\mathcal{EDD}$ is given as input, and computing them is out of the scope of this chapter.
- The energy demand distribution $\mathcal{EDD}$ is deterministic.
- There is no energy loss in sharing.
- A trust and a secure framework are used to preserve the security and privacy of the IoT devices [174].

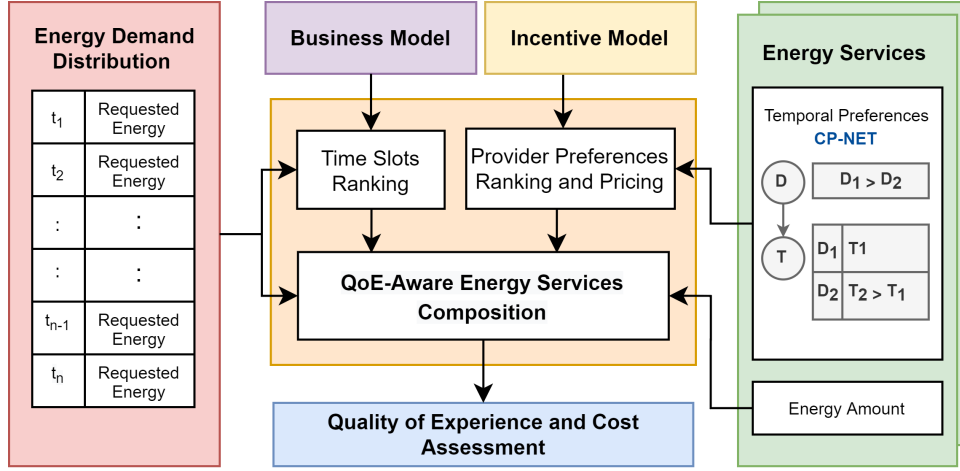


FIGURE 10.4. Quality of experience aware service composition Framework

10.3 Quality of Experience Framework

We introduce a quality of experience-aware composition framework for managing energy services to enhance consumers' experience QoE with respect to the super-provider business model BM (see Fig. 10.4). The framework aims to maximize the QoE while minimizing the cost C of using incentives. The framework is divided into three phases: (1) preferences ranking, (2) QoE -aware energy services composition, and (3) quality of experience and cost assessment. In the first phase, the framework ranks the time slots' importance and the providers' temporal preferences in parallel. The framework will also compute the price of the providers' provisioning services. In the second phase, An allocation energy provisioning strategy will be executed to maximize the QoE and minimize the cost of incentives. The last phase includes an assessment of the QoE and the cost of the resulting composition.

10.3.1 Providers' Temporal Preferences Ranking

In the first phase, the framework runs two parallel tasks: (1) rank the time slots' importance based on a business model BM , and (2) rank and price the providers' temporal provisioning preferences.

10.3.1.1 A CP-Net-based preferences ranking

Providers may be flexible when provisioning their service, i.e., they can come at multiple time slots. However, they may prefer certain times over others, e.g., a provider may prefer to come in the morning rather than the evening. As previously mentioned, assigning the providers to a more preferred time slot will require fewer rewards. Thus, ranking providers' temporal preferences may help the super-provider reduce the incentives they have to pay. *We use CP-Net to represent and rank the providers' temporal preferences.*

A CP-Net is a graphical model formally representing preferences with dependent relations among multiple attributes [173]. Given a set of attributes $V = \{X_1, \dots, X_n\}$ over which a user has preferences. A CP-Net over V consists of a directed graph (G) whose nodes are annotated with conditional preference tables ($CPT(X_i)$) for each $X_i \in V$. The directed graph G represent the dependency between the attributes $X_1, \dots, X_n$. A child node in a dependency graph depends on a set of direct parent nodes $Pa(X_i)$. The child node is connected by an arc from $Pa(X_i)$ to X_i in the dependency graph. Parent attributes affect the user's preferences over the value of X_i . Each node X_i in the dependency graph has $Pa(X_i)$, except for the root nodes. The conditional preference table $CPT(X_i)$ is defined over the finite set of domains $D(X_i)$. Each $CPT(X_i)$ associates a preference order between the values of $D(X_i)$ with each instantiation u of X_i 's parents $Pa(X_i) = U$.

In our context, we present each provider's temporal preferences using a CP-Net. For instance, the provider of S2 in Fig. 10.2 prefers to offer their service at day D_1 rather than D_2 . Therefore, the day (D) is the most important attribute in the dependency graph of CP_2 followed by the time slot (T) (see Fig. 10.5). The arc from "D" to "T" implies that the provider's time slot preferences depend on the selection of the day. The preference of days is expressed as $D_1 > D_2$ in the conditional preference table (CPT) of CP_2 . It implies that the provider prefers to come on the first day D_1 to the second day D_2 . The choice of days decides the choices of time slots in CP_2 . For

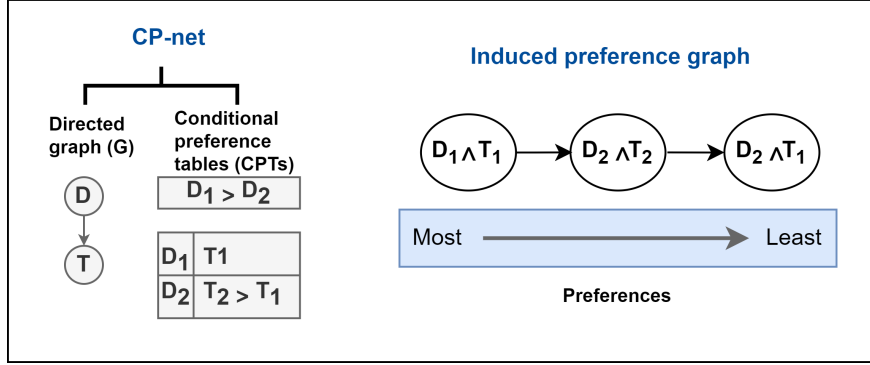


FIGURE 10.5. Provider's CP-Net-based temporal preferences ranking

example, if the provider was assigned to the second day D_2 , the provider prefers the second time slot T_2 to the first time slot T_1 according to CP_2 . If the choice of days is D_1 , the provider prefers the first time slot T_1 . Therefore, we generate each provider's ranked preferences PTP_i using a tuple of $\langle CP_i, CPT_i \rangle$ where CP_i is the CP-Net of provider P_i and CPT_i is the conditional preference table of provider P_i . PTP_i is computed by generating an induced graph G from the CP_i and CPT_i [173]. A preference outcome is a combination of values of all attributes of a CP-Net [173]. For example, $\{D_2 \wedge T_2\}$ and $\{D_2 \wedge T_1\}$ are two preference outcomes for CP_2 denoted by $o1$ and $o2$. According to the value of attribute T , it can be shown that $o1 > o2$ or $o1$ dominates $o2$. The dominant relationship between two preference outcomes is defined as a pre-order. Fig. 10.5 depicts the induced graph of a CP-Net (CP_2) with all preference outcomes.

The outcome preferences of an induced graph G will be used as the value of the provider temporal ranked preferences PTP_i . The computed ranks will be used to compute the provider's reward using the super-provider incentive model IM and Eq. 10.1. As previously mentioned, computing the providers' rewards depends partially on the ranked temporal preferences of the provider.

10.3.1.2 Time slots ranking

In this step, the importance of each time slot of the energy demand distribution $\mathcal{EDD}$ will be determined using the business model BM provided by the super-provider.

Then each time slot t_i in the $\mathcal{EDD}$ will be sorted by importance. There are several ways to compute the importance of time slots [51]. In this work, we assume the importance will be given to the system as input.

10.3.2 QoE-Aware Energy Services Composition

This phase aims to compose energy services to maximize the QoE and minimize the cost of rewards. We propose two approaches to compose energy services: *Importance-based* and *Heuristic-based*. The Importance-based composition starts with the most important time slots. The heuristic-based (heuristic) composition starts with the most important time slots with the least service availability. It also starts allocating the services with the closest temporal preference, e.g., a provider who selected the time slot as the first preference. We discuss each approach below.

10.3.2.1 Importance-based Energy Services Composition

The Importance-based composition is inspired by the priority resource allocation algorithm [95]. The Importance-based composition aims at maximizing the QoE by composing services for each time slot based on their importance. For example, if a provider offers services in two-time slots, the algorithm will assign the service to the more important time slot. If the time slot does not need the service, the service will be assigned to the next time slot.

Algorithm 10.1 presents the importance-based service composition. First, the algorithm sorts the time slots in $\mathcal{T}$ based on their importance (Line 1). Then the algorithm ranks the temporal preferences of the providers using CP-Net (Line 2). For every time slot t_i , the algorithm retrieves the total required amount of energy (Lines 3-4). Then, for each registered service s in t , the algorithm keeps adding services to the time slot until the required energy is fulfilled or all the available services have been selected (Lines 5-15). Additionally, for each added service, the algorithm will compute its reward using Eq. 10.1, which computes the reward based on the amount of provided

Algorithm 10.1 Importance-based Composition of Services**Input:** $\mathcal{T}, BM, IM, \mathcal{EDD}, priority, \mathcal{ES}, \mathcal{PTP}$ **Output:** $energy_comp, cost, QoE$

```

1:  $\mathcal{EDD} = \text{Sort}(\mathcal{EDD}, \mathcal{T}, I : \text{descending})$ 
2:  $\mathcal{PTP} = \text{CP\_Net\_Ranking}(\mathcal{ES})$ 
3: for  $t_i$  in  $\mathcal{EDD}$  do
4:    $demand = t_i.r$  ;
5:   for  $s_j$  in  $t_i.\mathcal{ES}$  do
6:     if  $demand > 0$  then
7:        $demand = demand - s_j.ae$ 
8:        $energy\_comp.add(t_i, s_j)$ 
9:        $rwd = \text{Compute\_Reward}(S_i, t_j, \mathcal{PTP}, IM)$ 
10:       $cost = cost + rwd$ 
11:     if  $demand < 0$  then
12:        $s_j.ae = demand * -1$ 
13:        $demand = 0$ 
14:     else
15:       Remove\_Service( $s_j, \mathcal{EDD}$ )
16:  $QoE = \text{Compute\_QoE}(\mathcal{T}, BM, \mathcal{EDD}, energy\_comp)$ 
17: return  $energy\_comp, cost, QoE$ 

```

energy, the provider-ranked preference of the current time slot, and the super-provider incentive model (Line 9). Moreover, the cost of the composition will be updated by adding the service's reward (Line 10). Lastly, if a service is partially needed, then the service available amount will be updated to be used by other registered time slots (Lines 11-13). Moreover, if a time slot fully uses a service, it will be removed from other registered time slots (Lines 14-15). Once the service allocation is completed, a quality of experience function will be invoked to assess the resulting composition (Line 16). The composition of selected services and their cost and quality of experience will be returned in Line 17.

10.3.2.2 Heuristic-based Energy Services Composition

The heuristic-based composition aims at maximizing the QoE by composing services for each time slot based on their importance and the available services. The intuitive idea of our approach is that the most important time slots with less available energy should be managed first. Thus, services should be assigned to them before time slots with higher availability. For instance, if a provider offers service on equally important

Algorithm 10.2 Heuristic-based Composition of Services**Input:** $\mathcal{T}, BM, IM, \mathcal{EDD}, \mathcal{ES}, \mathcal{PTP}$ **Output:** $energy_comp, cost, QoE$

```

1:  $\mathcal{EDD} = \text{Sort}(\mathcal{EDD}, \mathcal{T}, importance : descending, energy : descending, availability :$ 
    $ascending)$ 
2:  $\mathcal{PTP} = \text{CP\_Net\_Ranking}(\mathcal{ES})$ 
3: for  $t_i$  in  $\mathcal{EDD}$  do
4:    $demand = t_i.r$  ;
5:    $t_i.\mathcal{ES} = \text{Sort}(t_i.\mathcal{ES}, \mathcal{PTP} : ascending)$ 
6:   for  $s_j$  in  $t_i.\mathcal{ES}$  do
7:     if  $demand > 0$  then
8:        $demand = demand - s_j.ae$ 
9:        $energy\_comp.add(t_i, s_j)$ 
10:       $rwd = \text{Compute\_Reward}(S_i, t_j, \mathcal{PTP}, IM)$ 
11:       $cost = cost + rwd$ 
12:      if  $demand < 0$  then
13:         $s_j.ae = demand * -1$ 
14:         $demand = 0$ 
15:      else
16:        Remove\_Service( $s_j, \mathcal{EDD}$ )
17:  $QoE = \text{Compute\_QoE}(\mathcal{T}, BM, \mathcal{EDD}, energy\_comp)$ 
18: return  $energy\_comp, cost$ 

```

two-time slots, the algorithm will assign the service to the more demanding time slot. If that time slot did not need the service, the service will be assigned to the next time slot. This indicates that the order of time slots in composing services matters, because if a service is used in a time slot, it will be removed from others. Removing a service from a time slot may affect the amount of available energy and, thus, the number of served and satisfied consumers. The heuristic also starts allocating the services with the closest temporal preference, e.g., a provider which selected the time slot as the first preference. The intuitive idea is that assigning providers that selected a time slot as their first preference will cost less reward.

Algorithm 10.2 presents the heuristic-based service composition. First, the algorithm sorts the time slots in $\mathcal{T}$ based on their importance, energy demand, and service availability (Line 1). Then, the algorithm ranks the temporal preferences of the providers using CP-Net (Line 2). For every time slot t_i , the algorithm retrieves the total required amount of energy (Lines 3-4). The algorithm also sorts the services of each time slot based on the services' providers' temporal preferences (Line 5). The goal is to start

with services that selected the time slot as the first choice as their service will cost less reward. Then, for each service s in t , the algorithm keeps adding services to the time slot until the required energy is fulfilled or all the available services have been selected (Lines 6-16). Additionally, for each added service, the algorithm will compute its reward using Eq. 10.1, which computes the reward based on the amount of provided energy, the provider ranked preference of the current time slot, and the super-provider incentive model (Line 10). Moreover, the cost of the composition will be updated by adding the service's reward (Line 11). Lastly, if a service is partially needed, then the service available amount will be updated to be used by other registered time slots (Lines 12-14). Moreover, if a time slot fully uses a service, it will be removed from other registered time slots (Lines 15-16). Once the service allocation is completed, a quality of experience function will be invoked to assess the resulting composition (Line 17). The composition of selected services and their cost and quality of experience will be returned in Line 18.

10.4 Experiments Results

We compare our proposed composition techniques *importance-based* and *heuristic-based* with two baseline traditional resource allocation algorithms, namely, first come first served allocation algorithm (*greedy1*) and priority-based allocation algorithm (*greedy2*) [95]. In *greedy1*, the time slots and services will be processed based on their start time. In *greedy2*, the time slots will be processed based on the size of their energy demand. We also compare our approaches with a third baseline algorithm named *Preference-based*. In the preference-based algorithm, providers will be allocated to their most preferred time slots. We consider energy services as resources, and the super-provider's business model defines the priority in our composition problem. We evaluate the effectiveness of each approach by measuring the quality of experience, cost, the impact of preferences, and energy utilization. Moreover, we evaluate the efficiency of each approach by comparing their execution time.

10.4.1 Dataset Description

We used a real dataset from the developed app in [5]. The dataset consists of energy transfer records between a provider (smartphone) and a consumer (smartphone). The records attributes are the provider ID, consumer ID, transaction date, time, energy services and requests' amount, and transfer duration. We used the energy dataset to generate the QoS parameters for the energy services and requests. For instance, the amount of a wireless charging transfer in mAh defines the amount of requested/provided energy. In addition, the energy dataset records of a wireless charging transfer duration are used to define the end time of each request/service.

We augmented the energy-sharing dataset to mimic the crowd's behavior within microcells by utilizing a dataset published by IBM for a coffee shop chain with three branches in New York City ¹. The dataset consists of transaction records of customers' purchases in each coffee shop for one month. Each coffee shop consists of, on average, 560 transactional records per day and 16,500 transaction records in total. We used the IBM dataset to simulate the spatio-temporal features of energy services and requests. Our experiment used the consumer ID, transaction date, time, location, and coffee shop ID from each record in the dataset to define the spatio-temporal features of energy services and requests, e.g., start and location of energy service or a request. We generated the providers' ranked temporal preferences randomly. Tables 10.1 and 10.2 present the experiment parameters and the statistics about the used datasets.

TABLE 10.1. Parameters of the experiments' settings

Energy services		
Quality of service	Dataset	Value
Start time	IBM Cafe	Check-in
End time	Wireless Energy Dataset	Transfer Duration
Energy capacity	Wireless Energy Dataset	Provided Energy Amount
Energy requests		
Query parameters	Dataset	Value
Start time	IBM Cafe	Check-in
End time	Wireless Energy Dataset	Transfer Duration
Energy capacity	Wireless Energy Dataset	Requested Energy Amount

¹<https://ibm.co/2O7IvxJ>

TABLE 10.2. Statistics of the experiments' environment

Variables	Value
Total Energy Requests for coffee shop 1 in April	16830
Energy Services	10000
Duration of All Energy Services	5 - 60 minutes
Duration of Energy Requests	5 - 60 minutes
Provided Energy	5 - 100 mAh
Requested Energy	5 - 100 mAh
Number of time slots	8
Providers' preferences	1 - 4

10.4.2 Evaluation of the Composition Framework

We ran five experiments to determine the *effectiveness* and *efficiency* of our proposed approaches. The experiments evaluated the approaches' quality of experience, cost, energy utilization, the impact of preferences, and computation cost. We ran the approaches in different settings by changing: (1) the number of requests, (2) the number of services, and (3) the number of provider temporal preferences. We gradually increased each setting over the time interval T . We repeated the experiments 2,000 times at each point and considered the average value for each approach.

10.4.2.1 Quality of Experience Evaluation

The first experiment compared the proposed approaches' QoE against the three baselines. As previously stated, the QoE presents the satisfaction of consumers across time with respect to the super-provider's business model. Therefore, a high QoE of a composition indicates a higher level of satisfaction for consumers from a super-provider perspective. The QoE was computed using Eq. 10.2. Fig. 10.6 presents the average QoE in the microcell for each approach; the x-axis represents the number of energy requests in the microcell. The number of services in this experiment is a randomly selected set of 340 services with each run.

In Fig. 10.6, both proposed approaches perform better than the other three baselines. The importance-based approach performs better as it prioritizes the provisioning for

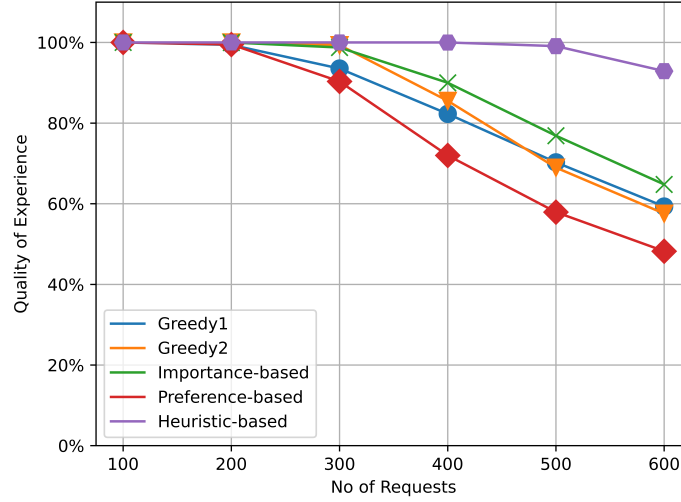


FIGURE 10.6. The average of QoE vs. number of requests

time slots with higher importance I score. Recall that the QoE metric is impacted by the time slot importance and is limited by the availability of services. Additionally, service providers may be flexible and register in multiple time slots. Thus, starting the service allocation by assigning them to important time slots will ensure they will not be allocated to less important ones. Similarly, the heuristic approach outperforms all other approaches as it prioritizes time slots with high importance, high energy demand, and less service availability. As QoE depends on the energy demand, prioritizing a time slot with high demand will increase its QoE. Another observation is the decrease of QoE with the increase of requests for all the composition approaches (see Fig. 10.6). For instance, all approaches provide a higher QoE at 100 requests than QoE at 600 requests. The increase in energy demand can explain this observation. If the demand increases with fixed service availability, many consumers' needs will not be fulfilled, and thereby the QoE will decrease.

The second experiment compared the QoE with a different number of services (see Fig. 10.7). The number of requests in this experiment was a randomly selected set of 353 requests with each run. In Fig. 10.7, the QoE increases with the increase of services for all the approaches. The increase in service availability can explain this

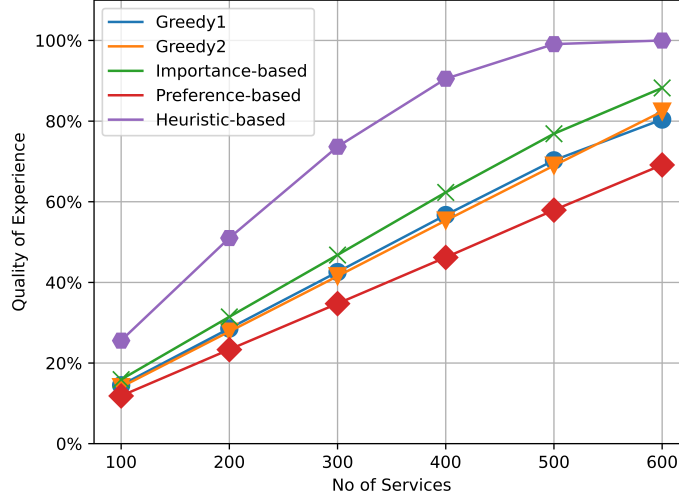


FIGURE 10.7. The average of QoE vs. number of services

observation. As services increase, most of the demand will be fulfilled regardless of the approach. Thus the QoE of all approaches is high. The heuristic approach outperforms all other approaches as it prioritizes time slots with high importance, high energy demand, and less service availability. As QoE depends on the energy demand, prioritizing a time slot with high demand will increase its QoE .

The third experiment compares the impact of the number of providers' temporal provisioning preferences on the QoE while increasing the number of requests. Recall that the number of temporal provisioning preferences is the number of time slots a provider may offer their service. In this experiment, the time interval T is divided into eight time slots. We tested the importance-based approach with provider preferences to offer their service on $\{1, 2, 3, 4\}$ time slots. In Fig. 10.8, the QoE is the highest when the number of providers' preferences were four. The temporal flexibility of the providers can justify this outcome. Providers' temporal flexibility increases the time to offer the service, which eases the time constraints of provisioning. We tested all the other approaches with the same setting, and they had similar behavior, but due to the page limit, we have omitted their figures.

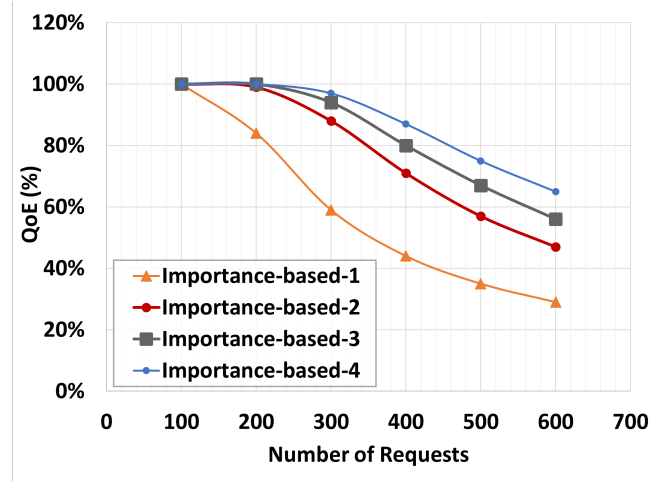


FIGURE 10.8. The average of QoE with different providers' temporal preferences

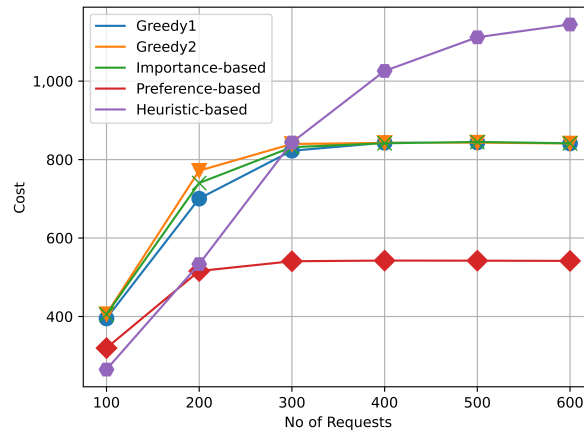


FIGURE 10.9. The average of cost vs. number of requests

10.4.3 Cost Evaluation

The fourth experiment evaluated the cost $\mathcal{C}$ of allocating services using the resulting composition of each approach. Recall that the cost is the summation of the rewards paid by the super-provider to entice providers to offer their services. In Fig. 10.9, the cost increases for all approaches as the number of requests increases. The increase in demands results in better utilization of the available energy (see Fig. 10.10). Therefore, increasing energy utilization will increase the rewards and the cost. Another observation in Fig. 10.9 is the low cost of the preference-based approach compared

to the greedy and importance-based approach. However, the preference-based has the lowest QoE (see Fig. 10.6), as it assigns providers to their first preference regardless of the time slots demand, which ensures a low reward cost but entails a lower QoE . The last observation in this experiment is the performance of the heuristic approach, which was better than all other approaches. This outcome resulted from considering the providers' ranked preferences in the heuristic approach while allocating services, i.e., if two services are available, the one with a lower reward will be selected (see Section 10.2.3). Note that the outcome of the heuristic approach becomes more costly than the preference-based one after a while, as the heuristic approach has better energy utilization, which eventually costs more rewards.

10.4.3.1 Energy Utilization Evaluation

In this subsection, we evaluate the effectiveness of the approaches in Energy Utilization (EU). EU is computed as the ratio of the amount of used energy services over the total amount of available services. Therefore, a high EU of a composition indicates its effectiveness in utilizing the available energy.

The first experiment compares the average EU of all approaches by changing the number of energy requests (see Fig. 10.10). In Fig. 10.10, all approaches EU increased as the number of requests increased, which can be explained by the increase in the energy demand. The higher the number of requests, the higher the probability of better allocating the available energy. Furthermore, the heuristic approach outperformed all other approaches as it prioritizes time slots with high energy demand, and less service availability. Another observation in this experiment is the low EU of the preference-based approach as it assigns providers to their first preference regardless of the time slots demand.

The second experiment compares the EU by varying the number of services (see Fig. 10.11). In Fig. 10.11, the EU decreased with the increased services for all the approaches. The increase in service availability can explain this observation. As services

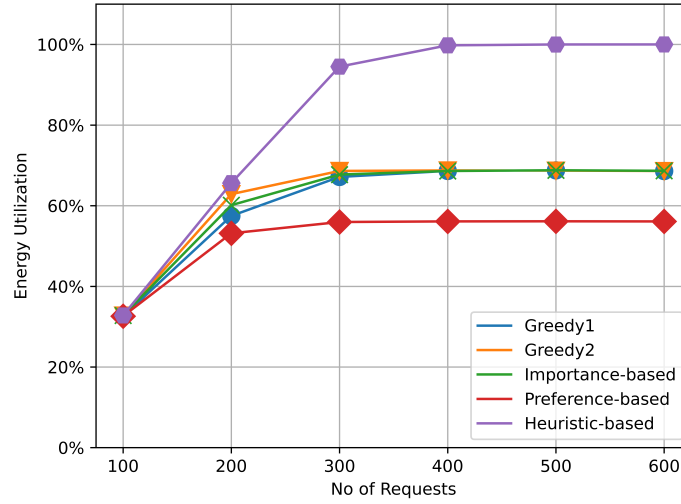


FIGURE 10.10. The average of energy utilization vs. number of requests

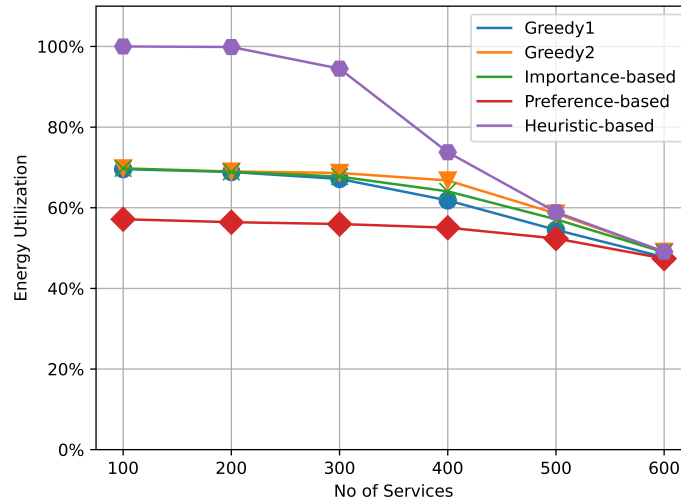


FIGURE 10.11. The average of the energy utilization EU vs. number of services

increase, unused available energy will increase regardless of the approach. Furthermore, the heuristic approach performed better than the other two approaches for the same aforementioned reasons in the first experiment.

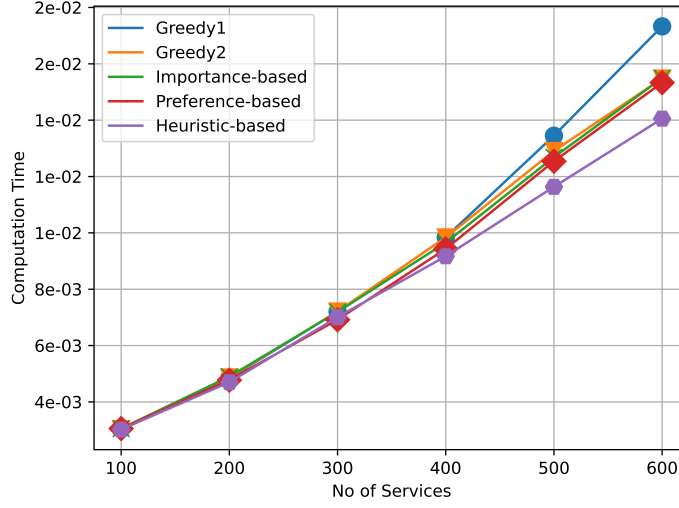


FIGURE 10.12. The average execution time vs. number of requests

10.4.3.2 Computation Efficiency Evaluation

The seventh experiment evaluates the computation cost of all approaches. Fig. 10.12 shows the average execution time of each approach. The execution time for all approaches increases as the energy demand increases. This observation can be explained by the availability of services to offer energy. The more services available, the more processing is required to assign these services.

10.5 Conclusion

We proposed a QoE-aware energy provisioning framework that evaluates QoE in a microcell. A new QoE model was proposed to capture the overall satisfaction across consumers over a period of time. Two QoE-driven composition algorithms of energy service were proposed. The importance-based approach prioritizes time slots with high importance to maximize the QoE and thereby increase the microcell owner's revenue. The heuristic-based approach prioritizes the most important and demanding time slots with the least available services. It also considers the temporal provisioning preferences to reduce the cost of rewards. The effectiveness of the proposed approaches

was investigated against three baseline approaches. Experimental results show that the heuristic-based approach outperforms all the evaluated approaches. In the future, we will extend the framework to fit into a dynamic crowdsourced environment by dealing with moving services and requests.

Constraints-aware Energy Service Composition

11.1 Introduction

Chapter 10 presented a QoE-based energy service composition to ensure consumers' satisfaction. Super-providers must address several QoE-based energy service composition challenges to maximize QoE. The *discrepancies in spatio-temporal preferences* between providers and consumers may significantly affect the energy sharing process within a microcell [39]. For example, an energy consumer may not find the energy necessary for their needs at a specific time and location [13][39][14]. The *limited availability of services* is another challenge caused by provider resistance to offering services [13][14]. Providers' resistance may be due to the critical nature of energy or their temporal preferences, e.g., resisting providing services in the morning [1]. Rewards may be used to encourage providers to participate in crowdsourcing energy services [48][1]. Although rewards may increase the likelihood of participation, they do not guarantee provider commitment [1]. The *uncertainty* in provider behavior may result in real-time re-compositions when a participating provider fails to commit. Finding nearby energy services to replace failed services in real-time is not guaranteed [49]. This may result in a negative impact on consumers' QoE in the microcell.

The QoE-based energy-service composition approaches proposed in Chapter 9 and Chapter 10 do not consider providers' uncertain availability. Moreover, they assume a super-provider has an *unlimited budget* of rewards to encourage providers' participation. However, an unlimited budget to invest in customer QoE in real-world scenarios

may not be feasible [175]. *This study particularly focuses on addressing the uncertain availability of providers and the constraint of the super-provider's budget.* Traditional resource allocation algorithms are not applicable in this context because they do not consider the uncertain availability of providers and the super-provider's context-sensitive perspective. For instance, a super-provider may prioritize smaller requests that have a higher weight in the super-provider's revenue, such as allocating a larger budget for a time slot that includes high-profit consumers with lower energy demand.

We propose an efficient energy service provisioning framework to maximize QoE. The framework effectively allocates uncertain available energy services while adhering to the super-provider's perspective and limited budget. Our framework extends the CP-Nets provisioning model proposed in Chapter 10 to capture the fine-grained providers' provisioning preferences and availability [176]. Our framework leverages providers' probabilistic preferences and flexibility to cope with time and budget constraints. We formulate the problem of QoE-based energy provisioning as a Mixed-Integer Linear Programming (MILP) problem. Our formulation considers the constraints of service provisioning and super-provider budget to provide optimal solutions in terms of QoE and rewards. In this regard, several solver models were used to solve the MILP problem. The main contributions of this chapter are:

- A QoE-aware super-provider-based framework for composing energy services.
- A novel probabilistic uncertainty-aware PCP-Net-based preference model for energy providers.
- A formulation of the energy service composition problem as a mixed-integer linear program.
- An extensive evaluation using a real dataset to show the effectiveness of the proposed approach.

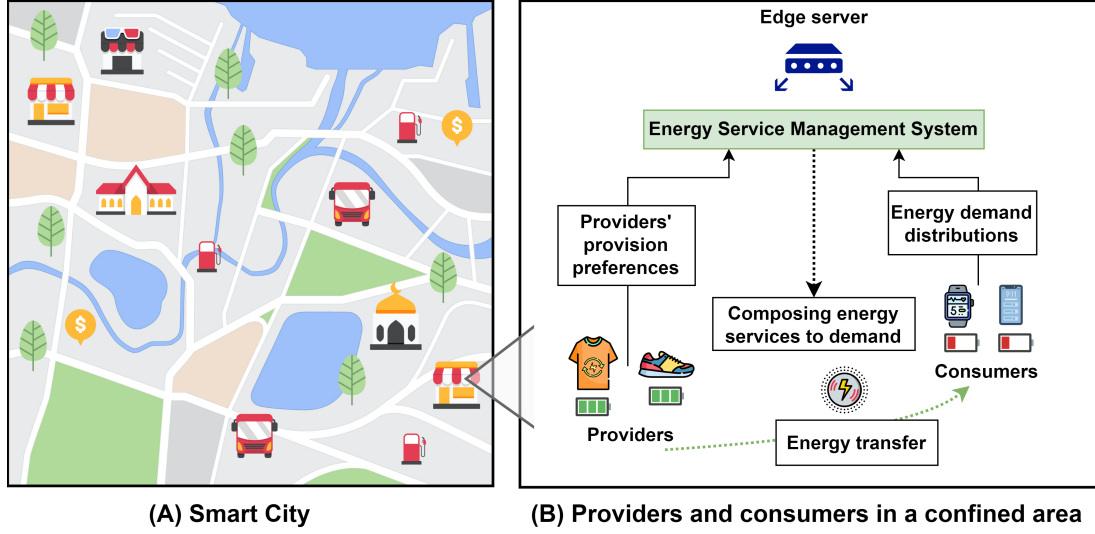


FIGURE 11.1. IoT energy services environment

11.1.1 Motivating Scenario

We describe a scenario at a coffee shop in which a microcell owner (super-provider) seeks to enhance consumers' *experience* by leveraging crowdsourced energy services (see Fig. 11.1(A)). Customers, i.e., IoT users, may act as energy providers or consumers. All local energy services and requests are submitted to the *edge* managed by the super-provider (see Fig. 11.1(B)). The super-provider has prior knowledge of the *Energy service Demand distribution (ED)* in the microcell over a time window (W) (see Fig. 11.2(A)). ED represents the requested energy in each time slot, e.g., 500 mAh at time slot T_1 . Moreover, the super-provider uses an important business model (IM) to estimate the importance of each time slot based on their business goals (see Fig. 11.2(B)). For instance, T_2 has an importance weight of 0.6 due to its significance to the business, e.g., lunch hours. In addition, the super-provider uses a reward budget model (RB) to compute the allocated budget for each time slot and the reward per energy unit ru . The budget will be used as *rewards* to encourage providers to share energy [1]. ru is used as a reservation price to give providers a benchmark of what rewards amount the super-provider is willing to pay for their service.

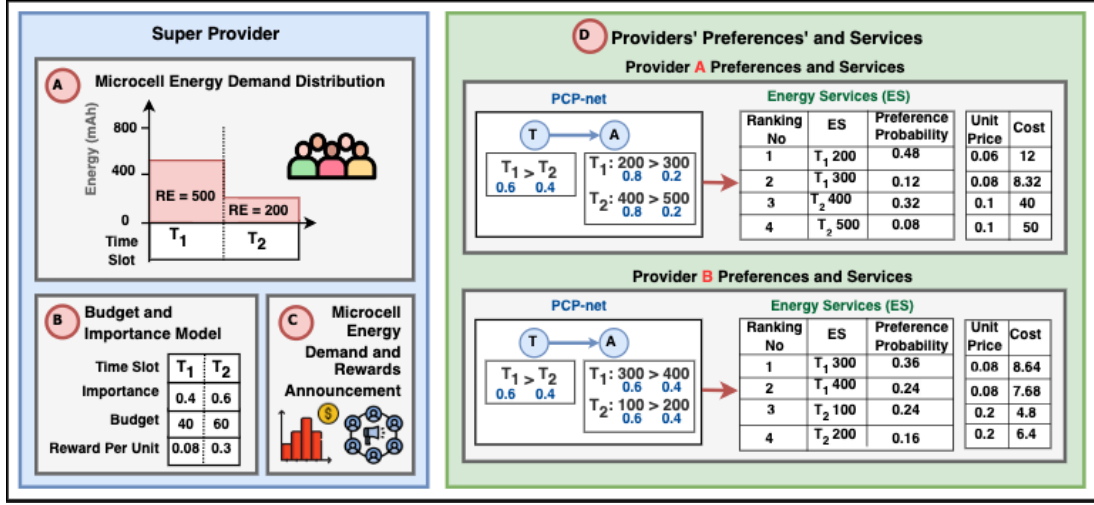


FIGURE 11.2. Example of the motivating scenario

The microcell energy demand is announced to providers a day ahead of the service day through a registry. The announcement includes the requested amount, duration, and reward per unit, ru for each time slot. For example, the time slot T_1 in Fig. 11.2 is advertised with energy requirement = 500 (mAh), $ru = 0.08\$$, and the time [10:00 a.m. - 12:00 p.m.]. Providers may express interest in offering their services the night before the service day, detailing the amount, availability, and price per energy unit (see Fig. 11.2(D)). Energy providers may offer services for one or more time slots, and their availability to commit may differ according to their preferences. Therefore, we propose the use of *probabilistic conditional preference networks (PCP-Net) as an effective tool for expressing fine-grained probabilistic preferences* [176]. The super-provider generates all providers' possible Expressions of Interest (*EoI*) to offer their services using their PCP-Net, i.e., a provider may offer different services based on their preferences. For example, provider A expressed interest in offering energy service S_1 with 200 mAh or S_2 with 300 mAh at T_1 , with a higher preference for offering S_1 . Providers are assumed to stay for the full-time slot with a fixed service amount.

The super-provider aims to serve as many consumers as possible by efficiently allocating services to the time slots. However, fulfilling multiple energy requests with limited energy services is a challenge. Furthermore, the super-provider's business model may prioritize time slots based on their importance, even with low energy demand. For

example, T_2 may be more important than T_1 . Additionally, the super-provider has a limited budget to incentivize providers. Therefore, the super-provider considers the provider's provisioning preferences while allocating services. Assigning providers to their preferred services will minimize the cost of rewards and increase the provider's likelihood to commit to the provisioning [10].

Allocating the limited available energy with the time constraints of services and requests, budget constraints of the service provider, and the uncertainty of providers' commitment represents critical challenges for efficient and QoE-aware provisioning of IoT energy services. We reformulate our service provisioning problem as a time and budget-constrained resource optimization problem, i.e., (i) maximizing the QoE by efficiently allocating the most certain available energy services and (ii) adhering to the budget of rewards. We propose a framework that will compose the best energy services to enhance the consumer experience with respect to the super-provider's business goals and budget. Our framework leverages flexible energy providers that offer services on *multiple* time slots by assigning them to the most important slots while considering the cost of their preferences. Considering providers' temporal *flexibility* may improve the services' distribution, fulfilling more energy requests and offering a better experience.

11.2 System Model

We first adopted the definitions of IoT energy services and requests from [23]. We then introduce a formal presentation of the super-provider preferences. We also present the concept of quality of experience in a microcell. Lastly, we present the problem formulation. This study considers a provisioning framework for stationary services. It focuses on the temporal and budget constraints of allocating energy to multiple time slots in a microcell within a predefined time window. The goal is to ensure the best consumer experience over a predefined time while addressing providers' uncertain behavior, adhering to the cost, and meeting the super-provider's business goals.

11.2.1 Energy Service and Request Models

DEFINITION 1. *Energy Service (S). S is defined as a tuple of $\langle s_{id}, p_{id}, F, Q \rangle$, where:*

- s_{id} is a unique service identifier,
- p_{id} is a unique provider identifier,
- F is the function of delivering wireless energy from one IoT device to another IoT device,
- Q is a tuple of $\langle q_1, q_2, \dots, q_n \rangle$, where each q_i denotes the Quality of Service (QoS) parameter of the energy service S , e.g., energy amount and geographical location.

DEFINITION 2. *Energy Quality of Service (QoS). QoS attributes enable users to differentiate between energy services [39]. We extend the definition of QoS attributes from [23]. QoS is presented as a tuple of $\langle a, l, d, b \rangle$ [23], where:*

- a is the amount of energy offered by the provider,
- l is the geographical location of the provider,
- d is the service's time interval $\langle st, et \rangle$,
- ap is the provider's availability to offer their service,
- ru is the provider's expected reward per energy unit.

The provider's *availability* ap is its *probability* of offering the service. ap is computed using the provider's probabilistic conditional preference network *PCP-Net* (see Section 11.3.1). The reward per energy unit ru may be computed using statistical methods that rely on the provider's preferences, such as the market history, temporal preferences, risk analysis, and the announced reservation price [177].

DEFINITION 3. *Energy Service Provider (P). P is an IoT device with spare energy to be shared as a service. P is defined as a tuple of $\langle p_{id}, PCP-Net \rangle$, where:*

- p_{id} is a unique provider identifier,

- *PCP-Net is a probabilistic conditional preferences network of the provider's provisioning preferences.*

DEFINITION 4. *Energy Request (R). R is a tuple of $\langle r_{id}, c_{id}, F, Q \rangle$, where:*

- *r_{id} is a unique request identifier,*
- *c_{id} is a unique consumer identifier,*
- *F is the function of receiving wireless energy by IoT devices from a nearby IoT device,*
- *Q is a tuple of $\langle q_1, q_2, \dots, q_n \rangle$, where each q_i denotes the Quality of Service (QoS) parameter of the energy request R , e.g., required energy amount and request duration. In our context, we define the energy request quality Q as $\langle re, loc, d \rangle$ where:*
 - *re is the requested amount of energy by the consumer,*
 - *loc is the geographical location of the energy consumer,*
 - *d is the request's time interval $\langle st, et \rangle$.*

11.2.2 Super-Provider Preferences

The super-provider uses the revenue and traffic history of the microcell to define five preferences : (1) time Window (W), (2) Importance Model (IM), (3) Energy Demand distribution (ED), (4) microcell Rewards Budget model (RB), and (5) Quality of Experience model (QoE). In what follows, we present each of these preferences.

DEFINITION 5. *Time Window (W). W is the time scale for managing the services provisioning, i.e., day, week, etc. For instance, a super-provider may want to focus on a certain week (e.g., back-to-school week) or a certain day (e.g., Black Friday). W is represented as a sequence of time slots $\langle t_1, t_2, \dots, t_n \rangle$ where t_i is a sub-interval in the time window W , e.g., [11:00 a.m. - 12:00 p.m.].*

DEFINITION 6. *Importance Model (IM). An IM is a sequence of importance values assigned to each time slot in the service time window, denoted as $\langle im_1, im_2, \dots, im_n \rangle$ and defined over W . The importance weight im_j of a time slot is defined by the foot*

traffic and revenue generated during t_j [52]. The value of im varies from zero to one. The importance score impacts the allocation of energy services.

DEFINITION 7. *Energy Demand Distribution (ED).*

ED is the predicted consumers' energy demand distribution in each time slot t in the service time window W . ED is defined as a sequence of $\{d_1, d_2, \dots, d_n\}$ where d_i is the aggregation of the predicted energy requests time slot t_i . The prediction of energy requests may be obtained using prediction techniques applied to the historical records h_i for all IoT users who visited the microcell at time slot t_i [178][179]. Energy requests may be aggregated according to their spatiotemporal features [49]. Given a set of predicted energy requests, the super-provider aggregates the energy requests using the composition approach proposed by [23]. The approach considers the time interval of each request to define a composite energy request that includes all available requests. If two or more energy requests overlap within a time interval, the super-provider sums the requested energy by all the available requests at that time interval using the following:

$$d_i(t_i) = \sum_{j=0}^m f(h_j, t_i) \quad (11.1)$$

where d_i is the aggregated energy demand at time slot t_i , h_j is the provider's j history of energy requests at time t_i , m is the number of providers who have visited the micro-cell at t_i in the past and f is the prediction function. Several models may predict foot traffic in a time slot and their energy demand [179].

DEFINITION 8. *Rewards Budget Model (RB): The super-provider uses rewards to entice providers to offer their energy. The super-provider grants rewards in the form of stored credits. We assume that the system limits the amount of requested energy. The provider may later use the earned credits to increase the limit of the requested energy as a consumer. A study on using incentives to increase energy provision participation was conducted [1]. The study shows that the amount of energy provided and the time of provisioning affect the value of the reward. Super-providers typically have a limited reward budget based on the super-provider business model [180]. The rewards budget*

(RB) for microcell M is distributed over the services time window W based on the importance model IM and the business goals of the super-provider [10]. RB is defined as a tuple of $\langle W, B, RU, tb \rangle$, where:

- $W = \{t_1, t_2, \dots, t_n\}$ is the set of time slots t that represent the service time window W ,
- $B = \{b_1, b_2, \dots, b_n\}$ is the corresponding budget limit for each time slot t in the service time window W ,
- $RU = \{ru_1, ru_2, \dots, ru_n\}$ is the corresponding reward per energy unit for each time slot t in the service time window W . The reward per uni (ru) is paid by the super-provider to providers based on the services they offer and the time of their provisioning. ru is also used as a reserved price, i.e., the super-provider will not pay higher than the ru , which may guide providers in determining their reward when they express their interest in offering energy [177]. ru_i is computed based on the allocated budget b_i for time slot t_i over the expected aggregated amount of energy demand d .

$$ru_i(t_i) = b_i/d(t_i) \quad (11.2)$$

- tb is the total budget the super-provider is willing to pay. Note that $\sum b_i$ may be greater than tb . Therefore, the super-provider may adjust the budget for each time slot but will not exceed the total budget tb .

11.2.3 Quality of Experience (QoE) Model:

We define the Quality of Experience (QoE) for a microcell during a specific time window W to measure the satisfaction of energy consumers with respect to the super-provider's importance model IM . We determine consumer satisfaction by assessing the ratio of allocated services to the energy demand distribution ED . Therefore, the

QoE in a microcell energy services over time window W is:

$$QoE = \sum_{j=1}^n \left((|ED.d_j - \sum_{i=1}^m (p_i.es_k.a * es_k.ap)|) * IM.im_j \right) \quad (11.3)$$

$$, \forall p_i \in SelP_j$$

where n is the number of time slots, ED is the microcell's energy demand distribution, d_j is the aggregated energy demand at time slot t_j , d_j is computed using Eq. 11.1, m is the number of selected providers that offer their services at time slot t_j , $p_i.es_k.a$ is provider i energy service k 's amount, $p_i.es_k.ap$ is provider i energy service k 's availability probability, IM is the service time importance model, $IM.im_j$ is the importance weight of time slot t_j , and $SelP_j$ is all the set of selected providers that offer their services at time slots t_j .

Cost (C): C is the total amount of rewards a super-provider will pay to the providers based on the allocation strategy.

$$C = \sum_{j=1}^n \sum_{i=1}^m (min(p_i.es_k.ru, RB.ru_j) * p_i.es_k.a) \quad (11.4)$$

where n is the number of time slots, m is the number of selected providers, $min(p_i.es_k.ru, RB.ru_j)$ is the minimum function for selecting the smaller reward unit between the reward ru requested by provider i to offer their service k ($p_i.es_k.ru$) and the reward determined by the super-provider's rewards budget RB for services offered in time slot j ($RB.ru_j$); $RB.ru_j$ is computed using Eq. 11.2, and $p_i.es_k.a$ is provider i 's energy service k 's amount. Note that t_j will be assigned based on the super-provider strategy.

11.2.4 Problem Definition

Given a microcell energy demand distribution $ED = \langle d_1, d_2, \dots, d_n \rangle$ over the temporal preferences of the super-provider in the time window $W = \langle t_1, t_2, \dots, t_n \rangle$, a super-provider's importance model $IM = \langle im_1, im_2, \dots, im_n \rangle$, the super-provider's

reward budget RB and a set of m energy providers $P = \{p_1, p_2, \dots, p_m\}$, each provider may have a set of Expression of Interest (EoI) services $S = \{s_1, s_2, \dots, s_k\}$ to offer different QoSs or different time slots. These EoI services are extracted from each provider's probabilistic conditional preference network $PCP-Net$. Energy providers register their services in terms of (1) the amount of energy $s.a$, (2) temporal provision preferences $s.d$, (3) availability probability $s.ap$, and (4) the reward per unit they expect $s.ru$. The super-provider will use their importance model IM , their reward budget RB , and the providers' services $P.S$ to allocate energy services to time slots. The allocation approach aims to maximize the QoE by *minimizing* the difference between the energy demand ED and allocated services $P.S$, while adhering to the reward budget RB and importance model IM . We transform the problem of quality-aware energy provisioning into a time-constrained resource allocation problem as follows:

Minimize

$$\sum_{j=1}^n \left((|ED.dj - \sum_{i=1}^m (p_i.es_k.a * es_k.ap)|) * IM.im_j \right)$$

Subject to

$$\begin{aligned} & - \sum_{i=1}^m (p_i.es_k.ru * p_i.es_k.a) < b_i, \forall b_i \in RB.B \\ & - \sum_{j=1}^n \left(\sum_{i=1}^m (p_i.es_k.ru * p_i.es_k.a) \right) < RB.tb \\ & - t_i \subset s_k.d \end{aligned} \tag{11.5}$$

Where $\forall S_k \in P_j.S$,

$S_k.d$ is the start and end of the provider's service

The goal of the strategy is to efficiently allocate the available energy services to time slots. The objective function attempts to optimally assign the most certain energy services according to their spatio-temporal features, providers' preferences, the importance model, and the required energy in the time slots.

We use the following assumptions to formulate the problem.

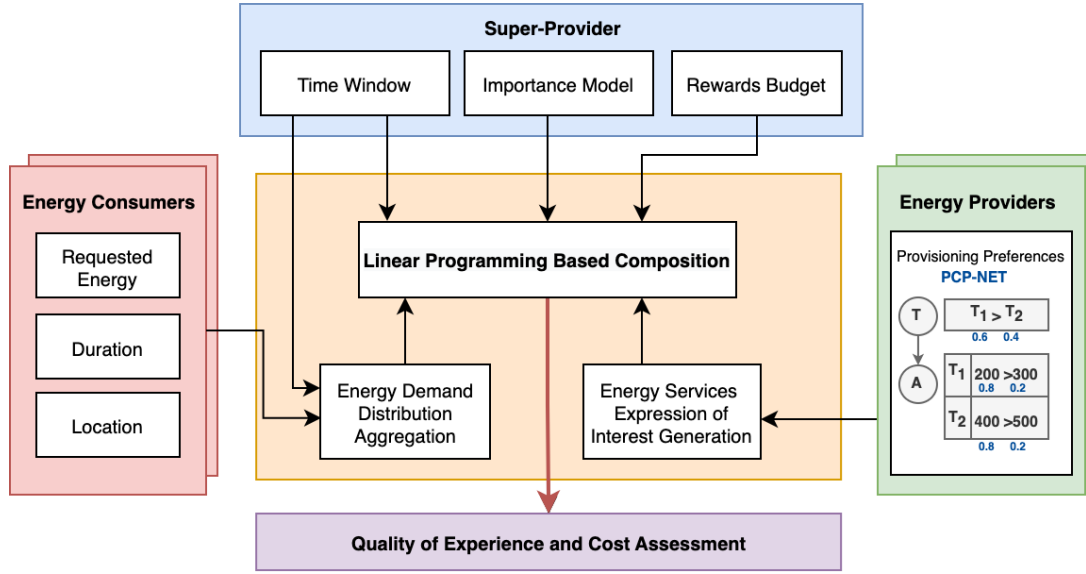


FIGURE 11.3. Quality of experience aware service composition framework

- Providers' energy size is fixed during composition.
- Providers and consumers have fixed locations.
- The microcell importance model IM , rewards budget RB , and energy demand distribution ED are given as input, and computing them is out of the scope of this work.
- The energy demand distribution ED is deterministic.
- There is no energy loss in sharing.
- A trust and secure framework is used to preserve the security and privacy of the IoT devices [174].

11.3 Energy Service Composition Framework

We introduce our energy service composition framework for allocating energy services to provide consumers with the best quality of experience QoE with respect to the super-provider reward budget (RB) and importance model IM (see Fig. 11.3). The framework aims to maximize QoE by minimizing the difference between the required energy and allocated services while adhering to the reward budget. The framework

is divided into three phases: (1) providers' energy service EoI generation, (2) energy service composition, and (3) QoE and cost assessment. We discuss each phase in detail in the following Subsections 11.3.1 - 11.3.3.

11.3.1 Energy Services Expression of Interest (EoI) Generation

In the first phase, the framework generates each provider's set of Expressions of Interest (EoI) to offer energy service based on their probabilistic conditional preference network, *PCP-Net*. *PCP-Net* ranks and generates a description of the services, their price per energy unit, and availability. As aforementioned, providers may have a set of provisioning preferences, i.e., they can come at several time slots or offer different services. However, they may prefer certain times over others, e.g., a provider may prefer to offer their service in the morning rather than in the evening. Intuitively, assigning providers to a *highly* preferred time slot requires *fewer* rewards. Thus, ranking providers' provisioning preferences may help the super-provider reduce the rewards amount.

A CP-Net-based provisioning model was proposed to capture provider provisioning preferences in Chapter 10; the model structures provider preferences using Conditional Preference Networks (CP-NeT). CP-Net enables ranking the provider's qualitative preferences compactly and intuitively [173]. However, the model cannot express providers' preferences in fine-grained details [181][176]. Specifically, the model cannot indicate the probability distributions or uncertain preferences for preference ordering [176]. Therefore, we extend the proposed model using a Probabilistic Conditional Preference Network (PCP-Net) to express provider preferences [176]. PCP-Net addresses this limitation by allowing the presentation of probabilistic distributions over preference orderings, making it possible to model and reason with preferences in uncertain situations about the preferences' strength or direction.

A PCP-Net is a direct acyclic graph formally representing probabilistic preferences with dependent relations among multiple attributes [181]. Assume a set of attributes

$V = \{X_1, \dots, X_n\}$ over which a user has preferences. A PCP-Net over V consists of a directed graph (DG), whose variables are annotated with a probabilistic conditional preference table ($PCPT(X_i)$) associated with a probability distribution p over the set of orders $\succ_{X_i|U}$: p for each $X_i \in V$. The directed graph G represents the dependency between the attributes $X_1, \dots, X_n$. A child node in a dependency graph depends on a set of direct parent nodes, $Pa(X_i)$. The child node is connected by an arc from $Pa(X_i)$ to X_i in the dependency graph. Parent attributes affect user preferences for the value of X_i . Each node X_i in the dependency graph has $Pa(X_i)$ except for the root nodes. The conditional probabilistic preference table $PCPT(X_i)$ is defined for a finite set of domains $D(X_i)$. Each $PCPT(X_i)$ associates a preference order between the values of $D(X_i)$ and each instantiation u of X_i 's parents $Pa(X_i) = U$. Moreover, $PCPT(X_i)$ provides a conditional probability for each u , given its parents' set of order $Pa(X_i)$. Therefore, given a PCP-Net and an outcome o , the probability of o is computed by multiplying the probabilities of the corresponding set of instantiation u as follows:

$$p(o) = P(u_0) * \prod_{i=1}^n p(u_i | Pa(X_i)_{u_{i-1}}), \forall u_i \in o \quad (11.6)$$

where $P(u_0)$ is the probability of the root parent and $p(u_i | Pa(X_i)_{u_{i-1}})$ is the conditional probability of u_i given the u_{i-1} of its parents. For better illustration, we can generate a tree $Ptree$ of the PCP-Net's attributes V , instances U , and probabilities $Prob$ by combining the directed graph (DG) and probabilistic conditional preference table ($PCPT(X_i)$). The tree may be used to compute the probability of each outcome by traversing the tree from the leaf node to the root node, visiting the nodes that combine the given outcome o (see Fig. 11.4(C)).

In our context, we present each provider's provisioning preferences using a PCP-Net. For instance, the provider in Fig. 11.4 prefers to offer their service at T_1 rather than T_2 . Therefore, time (T) is the most important attribute in the dependency graph of PCP , followed by the amount of energy (A). The arc from "T" to "A" implies that the provider's preferences on the amount of provided energy depend on the time selection. Time preference is expressed as $T_1 > T_2$ in the probabilistic conditional preference

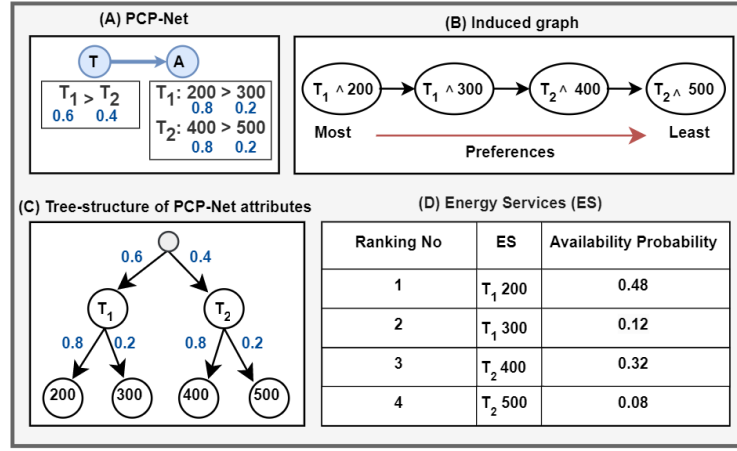


FIGURE 11.4. Provider's PCP-Net-based provisioning preferences

table (*PCPT*) of *PCP* with T_1 having a probability of 0.6 and T_2 having a probability of 0.4. This implies that the provider has a 60% preference for the first time slot, T_1 , to a 40% preference for the second time slot, T_2 . The choice of time determines the choice of energy amount in *PCP*. For example, if the provider is assigned to the second time slot T_2 , the provider prefers to offer 400 energy units rather than 500, according to *PCP*. Similarly, each amount of energy has a probability preference. For example, the provider has an 80% preference to offer 400 energy units, given that they were assigned to a time slot T_2 , i.e., $p(400 | T_2) = 0.8$. Therefore, given the outcome $o = T_2 \wedge 400$, the probability of o using Eq. 11.6 is $p(o) = p(T_2) * p(400 | T_2) = 0.4 * 0.8 = 0.32$. Therefore, for each provider, we generate their ranked provisioning preferences as a set of EoIs to offer energy services S_i using a tuple of $\langle PCP_i, PCPT_i \rangle$ where PCP_i is the PCP-Net of provider P_i and $PCPT_i$ is the probabilistic conditional preference table of provider P_i . PTP_i is computed by generating an induced graph, G , from PCP_i and $PCPT_i$ [173]. A preference outcome is a combination of the values of all the attributes of a PCP-Net [173]. For example, $\{T_1 \wedge 200\}$ and $\{T_1 \wedge 300\}$ are the two preference outcomes for *PCP* denoted by $o1$ and $o2$. According to the value of the attributes in *PCPT*, it can be shown that $o1 > o2$ or $o1$ dominates $o2$ with $o1$ having a probability of 48 % and $o2$ having a probability of 12%. The dominant relationship between the two preference outcomes is the pre-order between them. Fig. 11.4 depicts the induced graph of a PCP-Net with all preference outcomes and their probabilities.

Algorithm 11.1 ES Advertisement Generation and Ranking**Input:** P **Output:** $\mathcal{S}$

```

1: for  $p_i$  in  $P$  do
2:    $\mathcal{O} \leftarrow \text{Generate\_Induced\_Graph}(p_i.PCP - Net)$ 
3:    $ptree \leftarrow \text{Generate\_tree\_Graph}(p_i.PCP - Net)$ 
4:   for  $o_j$  in  $\mathcal{O}$  do
5:      $p(o_j) \leftarrow ptree(u_0)$ 
6:     for  $u_k$  in  $\mathcal{O}_l$  do
7:        $p(o_j) \leftarrow p(o_j) * ptree(u_k|u_{k-1})$ 
8:      $\mathcal{S}(p_i).\text{add\_ES}(o_j, p(o_j))$ 
9: return  $\mathcal{S}$ 

```

As mentioned previously, outcome preferences are used to define the provider's set of EoIs to offer energy services ES using Definitions 1 and 2, where each outcome o is used to define the service. These services will be used as an expression of interest from the provider to offer their services in the microcell.

Algorithm 11.1 presents the steps for generating energy services' EoI for each provider. Lines 1-8 traverse each provider's $PCP-Net$ to generate its energy services EoI. Line 2 generates the induced graph for each provider using their $PCP-Net$. Line 3 generates a tree that represents the order, dependency, and probability of each instance u by using the directed graph (DG) and probabilistic conditional preference table ($PCPT(X_i)$) of the provider's $PCP-Net$. Line 4 computes for each outcome its probability using the generated tree and Eq. 11.6.

11.3.2 Energy Services Composition

This phase aims to compose the most certain energy services to maximize QoE and adhere to the reward budget. Our composition approach utilizes flexible providers, i.e., providers with multiple provisioning preferences—to assign them to the most important time slots if they are within the budget. For example, if a provider offers services in two slots, the approach assigns the service to the most important time slot if it is within the budget. If the time slot does not require the service, the service will be assigned to the

next time slot. Moreover, the approach uses providers' *PCP-Net* to select services with a higher probability of availability. Finally, we used a linear programming technique to efficiently compose energy services [182]. In Subsection 11.3.2.1, we present the formulation of the problem as a linear programming problem.

11.3.2.1 MILP Formalization of Energy Service Composition

We formulate the problem of QoE-based energy provisioning as a Mixed-Integer Linear Programming (MILP) [183]. Typically, a problem solved using linear programming is defined by its objective functions and constraints. In the following, we will detail the instance of the general optimization problem outlined in Section 11.2.4 as a MILP formulation:

Decision variables

Let x_{ij} be a binary variable, where $x_{ij} = 1$ if a provider i is assigned to time slot j , and $x_{ij} = 0$ otherwise. The decision variable is represented as follows:

$$x_{ij} \in \{0, 1\}, \quad \forall i \in P, \quad j \in W \quad (11.7)$$

Objective function and constraints

Maximize:

$$\sum_{i=1}^n \sum_{j=1}^m x_{ij} \times P.S.a_{ij} \times P.S.ap_{ij} \quad (11.8)$$

Subject to:

$$x_{ij} \in \{0, 1\}, \quad \forall i \in P \ \& \ \forall j \in W \quad (11.9)$$

$$\sum_{j=1}^m x_{ij} = 1 \quad \forall j \in W, \quad \forall i \in P \quad (11.10)$$

$$0 \leq \sum_{j=1}^m \sum_{i=1}^n x_{ij} \times P.S.a_{ij} \times P.S.ap_{ij} \leq W.d_j, \quad \forall j \in W \quad (11.11)$$

$$x_{ij} \times P.S.ru_{ij} \leq x_{ij} \times RB.ru_j \forall i \in P \ \& \ \forall j \in W \quad (11.12)$$

$$0 \leq \sum_{i=1}^n x_{ij} \times P.S.a_{ij} \times \min(P.S.ru_{ij}, RB.ru_j) \leq RB.b_j, \quad (11.13)$$

$$\forall j \in W$$

$$0 \leq \sum_{j=1}^m \sum_{i=1}^n x_{ij} \times P.S.a_{ij} \times \min(P.S.ru_{ij}, RB.ru_j) \leq TB \quad (11.14)$$

The objective of the super-provider is to minimize the difference between the energy demand and allocated series to maximize the QoE. This may be achieved by maximizing the total allocated energy across all slots. Equation 11.8 represents the objective function, which is to maximize the total energy allocated across all providers-time slot pairings. Note that the service is multiplied by the probability of provider availability $P.S.ap_{ij}$, based on the provider preferences presented in Section 11.3.1. Multiplying the allocated energy by $P.S.ap_{ij}$ aims to select the providers with the highest probability. There is a set of constraints to formulating the problem in our context: Constraint 11.9 ensures that the decision variables $x[i, j]$ are binary, indicating whether provider i is assigned to slot j , i.e., $x=1$, or not, i.e., $x=0$. Constraint 11.10 ensures that each provider i is assigned exactly to one slot j . Constraint 11.11 ensures that the total energy allocated to each slot j is greater than zero and does not exceed the slot's demand. Note that the service is multiplied by the probability of provider availability, $P.S.ap_{ij}$, to ensure the selection of providers with the highest probability. Constraint 11.12 ensures that the unit price of the energy provided by provider i in slot j does not exceed the reserved unit price for that slot. Note that the super-provider pays the minimum price between the service unit price and its reserved unit price. Constraint 11.13 ensures that the total cost of the energy allocated to each slot j does not exceed the allocated budget for that slot. Constraint 11.14 ensures that the total cost of energy allocated across all slots does not exceed the total budget.

11.3.3 Energy Services Provisioning Composition

We present our proposed energy service composition approach. The goal is to compose certain energy services to maximize QoE and adhere to the reward budget. Our composition approach utilizes flexible providers, i.e., providers with multiple provisioning preferences, to assign them to the most important time slots if it is within the budget. We use our MILP formulation and several solvers to compose the optimal services.

Algorithm 11.2 presents the energy service composition approach. First, the algorithm generates each provider their EoI to offer their services $\mathcal{S}$ based on their provisioning preferences (*PCP-Net*) (Line 1). The search space is reduced by excluding services with a price per unit higher than the reserved price per unit, as determined by the super-provider budget model (Line 2). We define the MILP problem objective function to maximize the allocation of energy services. This minimizes the difference between the required and allocated energy, thereby maximizing QoE (Line 3). The decision variables for our problem are specified as the allocated energy services and time slot demand (Line 4). The objective function is defined to maximize the allocation of services (Line 5). The problem constraints are defined in Lines 6-16. Within the context of our problem, the constraints are as follows: (1) Providers' constraints: a provider should be assigned to a single time slot (Lines 6-7), (2) Energy constraints: the allocated energy for each time slot should be greater than zero and should not exceed its energy demand (Lines 8-9), and (4) reward budget constraints: (A) the unit price of the energy services in a slot does not exceed the reserved unit price for that slot (Lines 10-12); (B) The reward cost is greater than zero and less than the allocated budget for the corresponding time slot (Lines 13-14); and (C) The total reward cost is less than the total budget for the entire time window W (Lines 15-16). Line 17 composes the services by solving the linear programming problem. The linear programming problem can be solved using different methods, such as the graphical method, the simplex method, or tools such as R and an open solver [182]. Once the allocation of the service is completed, a quality of experience function (Line 18) and a cost function (Line 19) are invoked to assess the resulting composition using Eqs 11.3 and 11.4, respectively.

Algorithm 11.2 Energy Services Provisioning Composition**Input:** W, IM, ED, RB, P **Output:** $energy_comp, QoE, C$

```

1:  $\mathcal{S} \leftarrow \text{ES\_Ad\_Rank\_Generation}(P)$ 
2:  $\mathcal{S} \leftarrow \text{ES\_Filter}(RB, \mathcal{S})$ 
   Define the problem
3:  $problem \leftarrow \text{Create\_LP\_Problem}(\text{Maximize})$ 
   Define the decision variables
4:  $x \leftarrow \text{Create\_LP\_Variables}(x, [(i, j) \text{ for } i \text{ in } P$ 
    $\text{for } j \text{ in } W], \text{"Binary"})$ 
   Define the object function
5:  $obj \leftarrow obj + lpSum(x[i][j] * P.S[i][j].a * P.S[i][j].ap \text{ for } i \text{ in } P \text{ for } j \text{ in } W)$ 
   Define the constraints
   Define providers constraints
6: for  $i \text{ in } P$  do
7:    $const \leftarrow const + lpSum([x[i, j] \text{ for } j \text{ in } W]) \leq 1$ 
   Define energy constraints
8: for  $j \text{ in } W$  do
9:    $const \leftarrow const + lpSum([x[i, j] * P.S[i][j].a * P.S[i][j].ap \text{ for } i \text{ in } P]) \leq$ 
    $W[j].d$ 
   Define total rewards budget constraints
10: for  $i \text{ in } P$  do
11:   for  $j \text{ in } W$  do
12:      $const \leftarrow x[i][j] * P.S[i][j].ru \leq x[i][j] * RB[j].ru$ 
13:    $const \leftarrow \sum_{i=1}^n x[i][j] * P.S[i][j].a * \min(P.S[i][j].ru, RB[j].ru) \geq 0$ 
14:    $const \leftarrow \sum_{i=1}^n x[i][j] * P.S[i][j].a * \min(P.S[i][j].ru, RB[j].ru) \leq RB[j].b$ 
15:    $const \leftarrow \sum_{i=1}^n x[i][j] * P.S[i][j].a * \min(P.S[i][j].ru, RB[j].ru) \geq 0$ 
16:    $const \leftarrow \sum_{i=1}^n x[i][j] * P.S[i][j].a * \min(P.S[i][j].ru, RB[j].ru) \leq \text{sum}(RB)$ 
   Solve the LP problem using a solver model such as GLPK
17:  $energy\_comp \leftarrow \text{prob.solve}(\text{solver\_Mdl})$ 
18:  $QoE \leftarrow \text{Compute\_QoE}(W, IM, ED, energy\_comp)$ 
19:  $C \leftarrow \text{Compute\_Cost}(W, IM, ED, energy\_comp)$ 
20: return  $energy\_comp, QoE, C$ 

```

The composition of selected services, quality of experience, and cost will be returned in Line 20.

11.4 Experiments Results

We evaluated our proposed energy service composition (*EC*) using six solvers: GLPK, Gurobi, CPLEX, XPRESS, and SCIP[183]. An EC-solver name denotes each approach; for example, EC-GLPK. We compared our composition approach using different solvers with a baseline traditional resource allocation algorithm, namely, first come first served allocation (*Greedy1*) and a priority-based allocation algorithm (*Greedy2*) [95]. In *Greedy1*, time slots and services are processed based on their start time. In *Greedy2*, time slots are processed based on the budget in descending order, and services are sorted based on rewards and energy. We also compared our approach with the *Demand-based* method proposed by [10]. In the Demand-based algorithm, slots and services are sorted based on energy, and the allocation starts with the least flexible providers. Additionally, we compared our approach with the *Importance-based* method proposed in Chapter 10. In an importance-based algorithm, slots are sorted based on importance, and services are sorted based on energy. However, we modified the assumptions in demand-based and importance-based approaches to account for the fact that flexible providers may not be available for all time slots. We also attempted to implement a brute force approach; nonetheless, its substantial computational and memory requirements made effective execution impossible. The high resource demand rendered this method impractical for our specific problem setting. In our composition problem, energy services are considered resources, and the super-provider's importance model defines priority. We evaluated the effectiveness of each approach by measuring the quality of experience, cost, and energy utilization. Furthermore, we evaluated the efficiency of each method by comparing their execution times.

11.4.1 Dataset Description

We used a real dataset collected by an app developed in [6]. The app monitors the wireless energy-sharing process using coils connected to two smartphones. The consumer determines the granularity of the monitoring time. The app allows users to

TABLE 11.1. Statistics of the experiment's environment

Variables	Value
Total Energy Requests for coffee shop 1 in April	16830
Energy Services	10000
Duration of All Energy Services	5 - 30 minutes
Duration of Energy Requests	5 - 30 minutes
Provided Energy	5 - 100 mAh
Requested Energy	5 - 100 mAh
Number of time slots	8
Providers' preferences	1 - 4

request energy services from nearby smartphones by size, e.g., 1,000 mAh, or by time, e.g., to charge for the next 5 minutes. The dataset consists of energy transfer records between a provider (smartphone) and a consumer (smartphone). The records include the provider ID, consumer ID, transaction date, time, energy service amount, request amount, and transfer duration. We used the energy dataset to generate QoS parameters for the energy services and requests. For example, the amount of wireless charging transfer in mAh defines the amount of requested/provided energy. Additionally, the energy dataset records of the wireless charging transfer duration were used to define the end time of each request/service.

We augmented the energy sharing dataset to mimic the crowd's behavior within microcells by leveraging a dataset published by IBM for a coffee shop chain with three branches in New York City¹. The dataset consists of transaction records of customer purchases in each coffee shop over one month. Each coffee shop has 560 transactional records per day and 16,500 transaction records in total. We used the IBM dataset to simulate the spatio-temporal features of energy services and requests. Our experiment employed the consumer ID, transaction date, time, location, and coffee shop ID from each record in the dataset to define the spatio-temporal features of energy services and requests, e.g., the start and location of an energy service or a request. We randomly generated provider preferences. Table 11.1 presents the statistics of the experiment.

¹<https://ibm.co/2O7IvxJ>

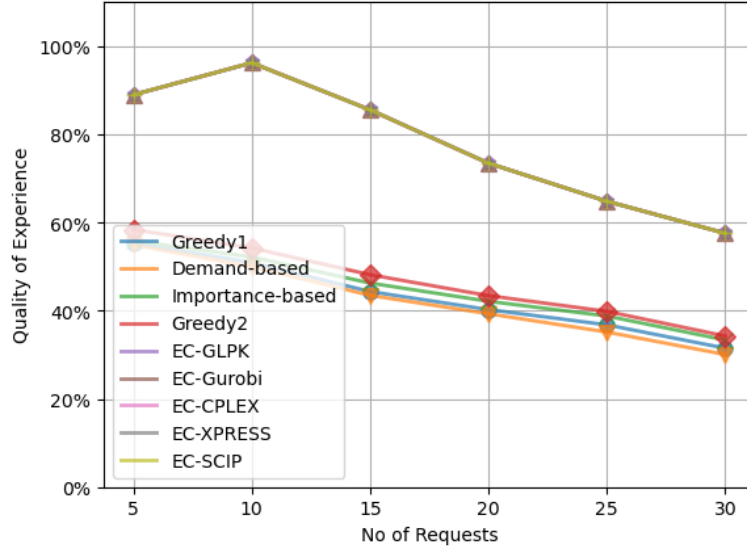


FIGURE 11.5. The average of QoE vs. number of requests

11.4.2 Evaluation of the Composition Framework

We ran four experiments to determine the *effectiveness* and *efficiency* of the proposed approach using different solvers. The experiments evaluated the approaches in terms of their quality of experience, cost, energy utilization, and computation cost. We ran the approaches in different settings by fixing the number of services and gradually changing the number of requests over the time interval W . We repeated the experiments 3,000 times and considered the average value for each approach.

The first experiment compares the proposed approach's QoE with those of the other four approaches. As previously mentioned, QoE represents consumers' satisfaction over time with respect to the super-provider's importance model. Therefore, a high QoE of a composition indicates a higher level of satisfaction for consumers from a super-provider's perspective. QoE was computed using Eq. 11.3. Fig. 11.5 presents the average QoE in the microcell for each approach; the x-axis represents the number of energy requests in the microcell. The number of services in this experiment was a randomly selected set of 15 services out of 340 services for each run. In Fig. 11.5, our proposed approach using each solver outperformed all other approaches. This was because our approach used linear programming solvers to compose the services.

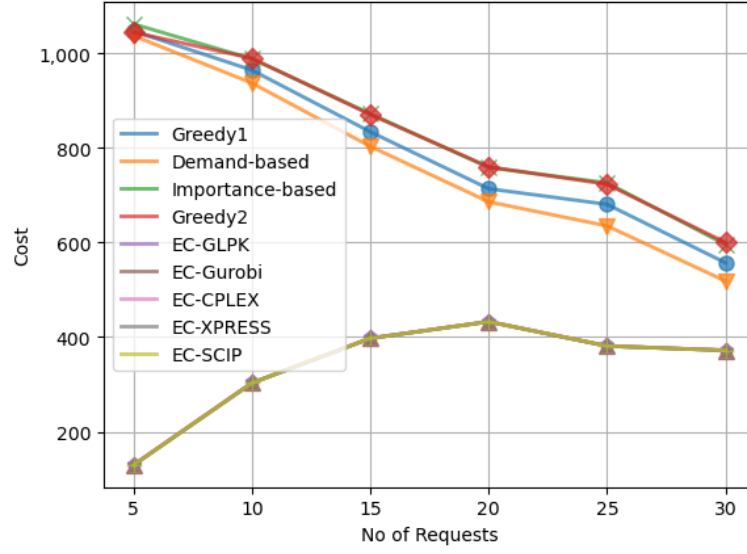


FIGURE 11.6. The average of cost vs. number of requests

In general, solvers surpass resource allocation algorithms because of their guaranteed optimality [183]. Typically, solvers can find an optimal solution to an optimization problem. In contrast, priority-based allocation may not always lead to an optimal solution. Another observation is a decrease in QoE with the increase in requests for all composition approaches. The increase in energy demand can explain this observation. If the demand rises with fixed service availability, many consumers' needs will not be met, resulting in a decrease in QoE .

The second experiment evaluated the cost $\mathcal{C}$ of allocating services using the resulting composition of each approach. Recall that the cost is the sum of the rewards paid by the super-provider to encourage providers to offer their services. The cost was computed using Eq. 11.4. In Fig. 11.6, the cost decreases for almost all approaches as the number of requests increases. An increase in demand leads to better-aligning service allocations with provider preferences. As mentioned in Chapter 10, aligning with providers' preferences will likely reduce costs, as providers will be less resistant to offering their services. Another observation is the low cost of the proposed approach using each solver compared with the other approaches. This is because the used linear programming solvers continually search for the optimal solution; if two services are available, the one with the lowest reward will be selected. On the other hand, all

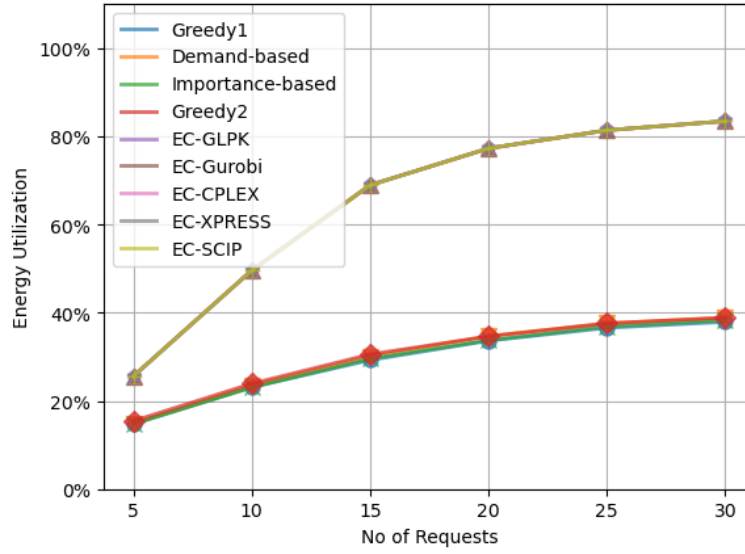


FIGURE 11.7. The average of energy utilization vs. number of requests

other resource allocation approaches select any service that meets the super-provider constraints.

The third experiment compared the average Energy Utilization (EU) for each composition (see Fig. 11.7). EU is computed as the ratio of the amount of used energy services to the total amount of available services. A high EU value indicates more satisfied consumers. In Fig. 11.7, the EU for the proposed composition using each solver is the highest. This is because the used solvers continually search for the optimal solution, that is, select the most certain services. In addition, EU increases for all approaches as the ratio of energy services to requests decreases. This is because a low ratio of services to requests means there are more requests than available services, which means that all services will be used. However, when the ratio is 100, the number of requests is equal to the number of services, which increases the probability of missing services due to time slot preferences.

11.5 Conclusion

In this chapter, we proposed an efficient QoE-aware energy service-provisioning framework. A new PCP-Net model was proposed to capture providers' fine-grained provisioning preferences and availability. A QoE-driven composition approach was proposed to effectively allocate energy services with *uncertain* availability while considering the super-provider constraints. Our approach uses linear programming to formulate a QoE-based energy service-provisioning problem. We used several linear programming solvers to compose services that maximize the QoE by leveraging providers' flexible preferences. We conducted a set of experiments to assess the effectiveness and efficiency of the proposed framework. The future direction is to consider the probability of change in the microcell energy demand.

Context-aware Trustworthy Energy Service Composition

12.1 Introduction

A key challenge in QoE-based energy service provisioning is to *assess providers' commitment* to sharing energy [13][14]. For example, a provider may terminate the energy transfer by leaving the transfer range. They may also stop the delivery of energy due to excessive device usage. Such service disruptions may reduce the expected amount of shared energy, impacting the consumer's QoE. In such cases, real-time service replacement is not always guaranteed [49]. Therefore, a *trust* management framework is necessary to evaluate the uncertainty in a provider's commitment.

Trust refers to the belief that providers will adhere to agreements and maintain the quality of service as advertised [14][13]. In crowdsourced energy environments, existing trust frameworks are not applicable due to energy scarcity and the fluctuating behavior of IoT users [50]. Energy scarcity implies providers tend to offer less energy than advertised [49]. While IoT users' commitment may fluctuate due to mobility and usage patterns of IoT users [14][50], using existing trust metrics results in low trust scores for most providers. Therefore, a new trust assessment framework is needed to better utilize the scarce energy services [13][14]. Adjusting the trust assessment based on providers' history and the context of service provisioning can lead to more accurate results. For example, existing trust frameworks would give a low score to a provider who consistently offers only 50 mAh, regardless of the advertised services. However,

if the super-provider only needs 50 mAh, assessing the provider based on this specific context would result in a higher trust score, making them a suitable candidate for allocation.

We propose a context-aware trust management framework to accurately assess and efficiently allocate trustworthy providers. The framework assesses providers' trustworthiness based on the super-provider constraints. The providers' trust scores are then used to *select* and *compose* the best set of energy services to fulfill the super-provider requirements. Our framework enables utilizing providers that may appear untrustworthy but can make valuable contributions while maintaining commitment. In addition, we propose a heuristic-based approach that ensures a higher QoE. Our approach composes additional services, based on the trust scores of providers, as a backup in the case of service cancellation. The contributions of this chapter are:

- A novel trust-aware framework to compose energy services and ensure QoE.
- A context-aware trust assessment model to fairly evaluate energy providers.
- A context model for defining super-provider constraints in evaluating provider trust.
- A heuristic-based approach to compose trustworthy energy services.

12.2 Motivating Scenario

We present a scenario where a microcell owner (super-provider) aims to enhance consumers' *experience* by utilizing crowdsourced energy services (see Fig. 12.1(A)). IoT users may act as energy providers or consumers. All local energy services and requests are submitted to the *edge* and managed by the super-provider (see Fig. 12.1(B)). We assume that the super-provider has prior knowledge of the *energy demand distribution* in the microcell over a given time window. Additionally, the super-provider adopts a QoE-based approach, such as [10][11], to select optimal providers to fulfill the microcell's energy demand. The super-provider also uses rewards to encourage providers'

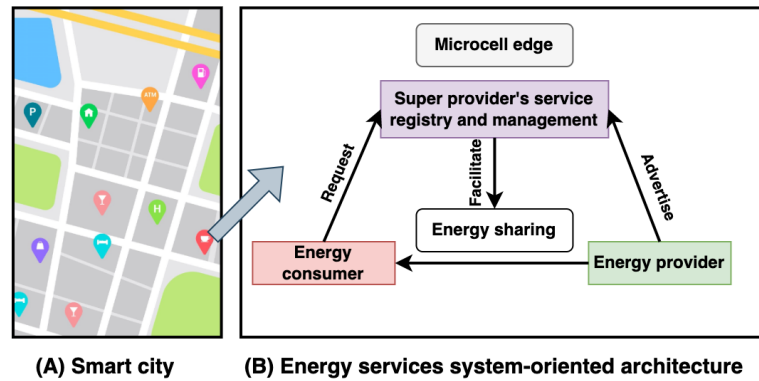


FIGURE 12.1. IoT energy services environment

participation [11][1], but rewards do not guarantee their commitment [1]. The *uncertainty* in providers' behavior may adversely affect consumers' QoE if a service fails and no other nearby services are available [49]. Thus, a trust framework is required to evaluate providers' commitment. While there are several trust frameworks for IoT services, they do not apply to energy services as they do not accommodate the nature of energy services. An inaccurate trust assessment may lead to misjudgment of a service. This could lead to either not using it due to a low score or using an untrustworthy service that disappoints consumers. Given energy scarcity, it is paramount to correctly assess trust to efficiently utilize all available services. Furthermore, super-providers typically may vary in their service expectations and requirements. These expectations may result in different trustworthiness scores for the same provider. Hence, finding the right trust assessment that meets super-provider expectations is challenging.

Fig. 12.2 shows a provider's advertised service and history. Let us assume we limit the trust metrics to the provider's commitment to fulfill an energy request. Using a traditional trust framework where a provider is assessed based on all its available history, the provider trust score will be 67% without an estimation of the provided amount. However, using a more context-aware trust framework, where a super-provider defines their minimum requirements as a threshold, may result in a different score. For example, if a super-provider has a 60 mAh threshold, they will look at the provider's history and how often they fulfill 60 mAh requests. Based on this, they rated the provider's service to 60 mAh with a 93% trust score. In another example, if the threshold

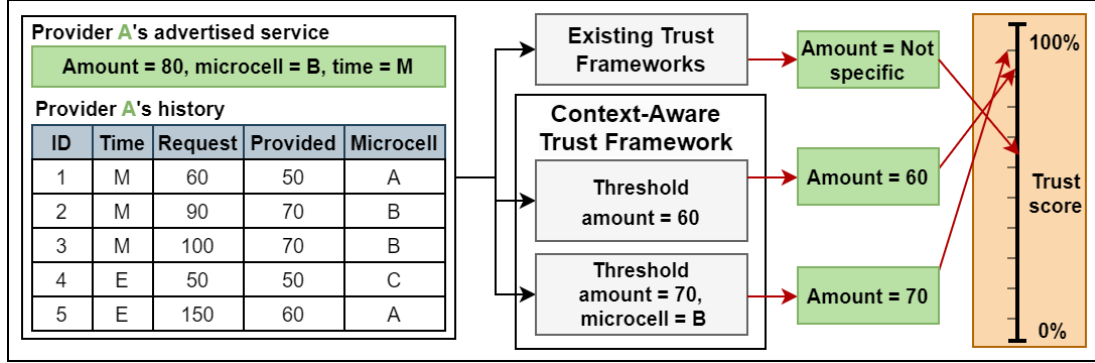


FIGURE 12.2. Example of how different trust assessments yield different trust scores

is set to evaluate the rate of providing 70 mAh services in microcell B, the trust framework will consider the provider's history within microcell B to understand their performance in that area. As a result, the super-provider will rate the provider's services as 70 mAh with a 100% trust score. Allocating the *limited* available energy amid the *uncertainty* of providers' commitment presents a critical challenge for the efficient and QoE-aware provisioning of IoT energy services [13]. We propose a context-aware trust-management framework for energy services. Our framework assesses provider trust based on the expectations and requirements of super-providers, which may lead to varying trust scores for the same service. The framework composes the most trustworthy energy services, ensuring consumer QoE and leveraging super-providers' expectations to assess and adhere to their standards. Moreover, our composition approach accounts for the natural untrustworthiness in the energy crowdsourcing environment by selecting additional services as a backup. The selection of these services relies on the trustworthiness of the available services

12.3 System Model

We focus on assessing the trust of energy services in a microcell during a single time slot t . We use the below definitions to formalize the problem.

12.3.1 Energy Service Model

DEFINITION 1. **Energy Service (S).** We adopt the definition of EaaS introduced in [25]. An energy service S is defined as a tuple of $\langle s_{id}, p_{id}, F, Q \rangle$, where:

- s_{id} is a unique service identifier,
- p_{id} is a unique provider identifier,
- F is the function of delivering wireless energy,
- Q is a tuple of $\langle q_1, q_2, \dots, q_n \rangle$, where each q_i denotes the Quality of Service (QoS), e.g., energy amount and geographical location.

DEFINITION 2. **Energy Quality of Service (QoS).** QoS attributes enable users to differentiate between energy services [39]. We extend the definition of QoS attributes from [23]. QoS is presented as a tuple of $\langle a, l, d, b \rangle$ [23], where:

- a is the amount of energy offered by the provider,
- l is the geographical location of the provider,
- d is the service's time interval $\langle st, et \rangle$,
- tr is the provider's trust score to offer their service,

DEFINITION 3. **Energy Service Provider (P).** P is an IoT device with spare energy to be shared as a service. P is defined as a tuple of $\langle p_{id}, \mathcal{H} \rangle$, where:

- p_{id} is a unique provider identifier,
- $\mathcal{H}$ is the set of historical records of the provider's previous provisioning. We assume that all providers have a history, and dealing with newcomers is not the focus of this work. We also assume that the edge will retrieve the history of providers from the cloud (see Fig. 12.1(B)). $\mathcal{H}$ is a tuple of $\langle h_1, h_2, \dots, h_m \rangle$, where each h_i represents the record of previous energy transfer as $\langle s, de, m, t \rangle$ where:
 - S is the advertised energy service,
 - de is the amount of delivered energy,
 - m is the microcell where the energy sharing occurred,

- t is the time duration $\langle st, et \rangle$ of the energy sharing.

12.3.2 Super-Provider Preferences

DEFINITION 4. **Energy Demand Distribution (ED).** *The super-provider uses the traffic history of the microcell to define the Energy Demand distribution (ED).*

We adopt the definition of ED from [10]. ED is the predicted consumers' energy demand distribution in the service time window W . Therefore, the demand for a single time slot is defined as a tuple $\langle d, t \rangle$ where d is the aggregation of the predicted energy requests at time interval t . The prediction of energy requests may be obtained using prediction techniques applied to the historical records $\mathcal{H}$ for all IoT users who visited the microcell at time interval t [178][179]. Energy requests may be aggregated according to their spatio-temporal features [49]. Given a set of predicted energy requests, the super-provider aggregates the energy requests using the composition approach proposed by [23]. The approach considers the time interval of each request to define a composite energy request that includes all available requests. The super-provider sums the requested energy by all the available requests at that time interval using:

$$d(t) = \sum_{j=0}^m f(\mathcal{H}_j, t) \quad (12.1)$$

where d is the aggregated energy demand at time slot t , $\mathcal{H}_j$ is the provider's j history of energy requests at time t , m is the number of consumers who have visited the microcell at t in the past and f is the prediction function.

DEFINITION 5. **Quality of Experience (QoE).** *QoE measures the aggregated satisfaction of energy consumers in a microcell at time slot t .*

$$QoE = \sum_{i=1}^n S_{i.a} / \sum_{j=1}^m R_{j.re} \quad (12.2)$$

where n is the number of selected services, a is energy service i 's amount, and $\sum_{j=1}^m R_j.re$ is the aggregated energy demand at time slot t . In this study, we assume the energy demand is deterministic and is equal to $Ed.d$ (see Eq. 12.1), m is the number of aggregated requests, and re is energy request size.

12.3.3 Problem Definition

Assume a super-provider in a microcell m who wants to fulfill their consumer's expected energy demand $ED.d$ at a time slot t . The super-provider has a set of n candidate providers $\mathcal{P} = \{p_1, p_2, \dots, p_n\}$ who expressed their interest in offering their service $\mathcal{S} = \{s_1, s_2, \dots, s_n\}$ at that time and location. Each provider has a history $\mathcal{H} = \langle h_1, h_2, \dots, h_m \rangle$ of sharing energy. The super-provider uses a context-aware trust model to evaluate the trustworthiness of the providers. The super-provider aims to ensure QoE by maximizing the fulfillment of energy demand. This will be achieved by allocating the most trustworthy providers that will likely commit to their advertised service. *We reformulate the service provision problem as a time-constrained optimization problem as follows:*

- Maximize QoE as $\sum_{i=1}^n S_i.a * \mathcal{P}_{Trust_i} / \sum_{j=1}^m R_j.re$

Subject to:

- $Ed.d > 0$,
- $S_i.d \subset t$ for each $S_i \in \mathcal{S}$.

where QoE is computed based on the allocated energy multiplied by the corresponding provider trust score, $\mathcal{P}_{Trust_i}$ is provider i trust score, $S_j.d$ is the time interval $\langle st, et \rangle$ a provider of S_i may offer their energy, t is the duration of a time interval, and $Ed.d$ is the aggregated energy demand at time slot t .

We use the following assumptions to formulate the problem:

- Providers' energy size is fixed during composition.

- Providers and consumers are static during energy sharing.
- The super-provider context model is given as input, and determining the constraints is out of the scope of this study.
- There is no energy loss in sharing. As the technology matures, we anticipate that the devices will share more energy, and the sharing energy loss will become minimal [24][13].
- A reward system incentivizes providers to offer their service [1].
- All providers have a history of energy requests collected in a phase before deploying our proposed framework.
- The energy demand distribution ED is deterministic.
- A secure framework has been implemented to preserve the integrity and privacy of the IoT devices [168].

12.4 Trust Assessment Model

We define the provider trust level assessment to determine the trustworthiness score ($\mathcal{P}_{Trust}$) of an energy service. The trust level assessment considers providers' behavior and history in delivering energy. We compute the trustworthiness of a provider using the following attributes:

- **Success Rate** Given a provider, we define its *success rate* ($\mathcal{SR}_P$) as the ratio of *completed energy services* to the total number of *initiated services* by that provider. Completed energy services refer to services where providers deliver all the advertised energy to their consumers. Providers-initiated services include all services, regardless of whether they have been completed. We compute $\mathcal{SR}_P$ of provider P as follows:

$$\mathcal{SR}_P = \frac{|\{S \in \mathcal{E}_P \mid S \text{ is completed}\}|}{|\mathcal{E}_P|} \quad (12.3)$$

where $\mathcal{E}_P$ is all services initiated by provider P and $|\cdot|$ is the cardinality of the set.

- **Delivery Size** The Delivery Size rate ($\mathcal{DS}_P$) measures the rate of energy delivery attempts based on the size of the service. $\mathcal{DS}_P$ may be calculated as the ratio of the sum of the sizes of successful energy delivery attempts to the sum of the sizes of all energy delivery attempts by a provider. $\mathcal{DS}_P$ is computed using:

$$\mathcal{DS}_P = \frac{\sum_{i=1}^n h_i.de}{\sum_{i=1}^n S_i.a} \quad (12.4)$$

where n represents the number of previous services delivered by the provider, $h_i.de$ represents the actual delivered energy retrieved from the history record h_i , and $S_i.a$ represents the advertised amount of the i -th energy service.

- **Timeliness Score** The TimeLiness score, denoted as $\mathcal{TL}_P$, quantifies the punctuality of a provider in delivering energy services. Disconnections in the transfer process may sometimes occur due to the provider's indoor movement, leading to delays in energy transfer [50]. Such delays, causing consumers to wait additional time for their services, can adversely affect their QoE. Consequently, the timeliness score is an essential attribute of the trust model. $\mathcal{TL}_P$ is calculated using the following formula:

$$\mathcal{TL}_P = \begin{cases} 1 & \text{if } \sum_{i=1}^n (h_i.t.et - S_i.et) \leq 0 \\ \frac{1}{\sum_{i=1}^n (h_i.t.et - S_i.et)/n} & \text{otherwise} \end{cases} \quad (12.5)$$

where n represents the number of previous energy services delivered by the provider, $h_i.t.et$ represents the actual end time of delivering a service i retrieved from the history record h_i of the provider, and $S_i.et$ represents the advertised end time of the i -th energy service.

- **Impact Score:** As previously stated, a provider's service cancellation may impact the consumers' QoE. The impact of canceling service on the consumers may be used to evaluate the providers' trustworthiness. We compute the service failure's impact $\mathcal{F}_S$ based on the number of affected consumers. We define the failure impact of an energy service by the number of impacted consumers due to its cancellation. We compute the failure impact $\mathcal{F}_S$ of a

canceled energy service as follows:

$$\mathcal{F}_S = \frac{|\{C \in c \mid C \text{ is receiving from } S\}|}{|C|} \quad (12.6)$$

where C is the set of all consumers in the microcell within the duration of the service $[S_{st} - S_{et}]$ and $|\cdot|$ is the cardinality of a set.

We define the provider impact as the cumulative effect of all their services on consumers. Typically, a trustworthy provider will have minimal impact on consumers when a failure occurs. In other words, the provider's impact is calculated as the complementary value to the failure impact. Therefore, we compute the provider impact score $\mathcal{I}_P$ as follows:

$$\mathcal{I}_P = \left(\sum_{i=1}^n (1 - \mathcal{F}_{S_i}) \right) / n \quad (12.7)$$

where n represents the number of previous energy services delivered by the provider.

- **Mobility Pattern:** The mobility patterns of IoT devices may be computed based on the spatial and temporal regularity in their owners' movements [184] [39]. Energy providers' mobility patterns may influence their trustworthiness. For example, if a provider's mobility pattern shows their staying time at a microcell is 5 minutes, and they advertised a service that requires 30 minutes, this might be an indication of the probability of not committing to the request. Therefore, we consider the staying duration pattern as an attribute in the trust-level assessment. We use the history of the provider's previous services to compute their duration pattern. The number of previous services should be greater than a predefined threshold α to be considered. We compute the staying duration pattern $\mathcal{SD}_P$ as:

$$\mathcal{SD}_P = \left(\sum_{i=1}^k (S_{i.et} - S_{i.st}) \right) / k \mid k > \alpha \quad (12.8)$$

where $S_{i.st}$ is the provider's previous service i start time, $S_{i.et}$ is the provider's previous service i end time, and k is the number of previous services.

We compute the provider Duration trust factor $\mathcal{D}_P$ using the staying duration pattern $\mathcal{SD}_P$ as follows:

$$\mathcal{D}_P = \begin{cases} 1 & \text{if } (S.et - S.st) \leq \mathcal{SD}_P \\ \frac{\mathcal{SD}_C}{(S.et - S.st)} & \text{otherwise} \end{cases} \quad (12.9)$$

where $S.st$ is the provider's current service start time, $S.et$ is the provider's current service end time.

Providers' Trustworthiness

The provider trust score ($\mathcal{P}_{Trust}$) combines all the aforementioned trust attributes with weights assigned based on their importance to the specific use case. ($\mathcal{P}_{Trust}$) is calculated using the following formula:

$$\mathcal{P}_{Trust} = \sum_{J \in \mathcal{I}} w_J \times \mathcal{L}_P \quad (12.10)$$

where $\mathcal{L} = \{SR, TL, DS, I, D\}$ and w represents the weight assigned to each trust attribute based on its importance, with the constraint that $\sum w_i = 1$ to normalize the impact of each factor on the overall trustworthiness score. Multiple methods exist for calculating the weight of each attribute, depending on the preference of the super-provider [40]. We assign all w to 0.2 for equal impact on the trust score.

12.5 Energy Service Composition Framework

We introduce our energy service composition framework for allocating trustworthy energy services to provide consumers with the best QoE (see Fig. 12.3). The framework ensures QoE by allocating trustworthy services based on the super-provider context. The framework is divided into three phases: (1) Context-aware trust assessment, (2) Trust-based energy service composition, and (3) QoE assessment. In this subsection follows, we discuss each phase in detail:

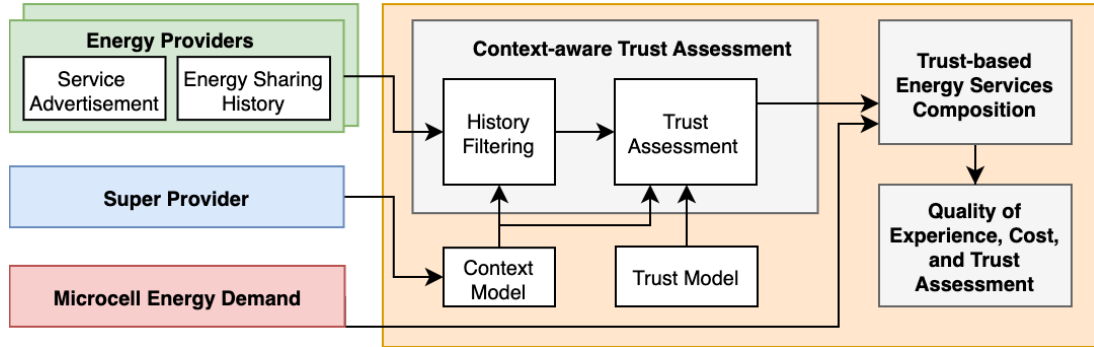


FIGURE 12.3. Trust-based energy service composition framework

12.5.1 Context-Aware Trust Assessment

In this phase, the proposed framework evaluates the historical performance of each provider to assess their trustworthiness. This framework uses a trust model to evaluate a provider's trust. It also utilizes a context model that represents the constraints set by the super-provider regarding the attributes and data used for the trust assessment. As mentioned, the perspective of assessing trust may result in different trust scores (see Fig. 12.4). We identify two main factors that impact the trust score in an energy crowdsourcing environment: (1) the history used and (2) the super-provider perspective of expected performance. For instance, a super-provider may assess a provider on what they advertised to deliver, or based on a certain required amount. Therefore, we propose a context-aware trust assessment model (see Fig. 12.4). Our model consists of two main components: (1) history constraints and (2) trust assessment constraints. This model is then used in two steps in the framework: history filtering and adaptive trust assessment (see Fig. 12.3). In the following subsections, we will present the context model and discuss each step of the framework in detail.

12.5.1.1 Context Model

As mentioned, the context model establishes the constraints set by the super-provider concerning the attributes and data employed to evaluate a provider's trustworthiness. The context model consists of two sets of constraints. The first set pertains to the provider's history, addressing aspects such as considering their entire history or focusing

solely on their behavior within a specific microcell. The second type defines the super-provider's constraints on trust assessment attributes, which may involve using all trust model attributes or only a subset, adjusting the weights of trust model attributes, or evaluating particular attributes.

- **History Constraints:** As previously mentioned, trust assessment can employ the entire history of a provider or only a portion of it, depending on the super-provider history constraints. For instance, a super-provider might assess trust based solely on the provider's history within a specific microcell. Another example would be evaluating the provider's energy-sharing history for larger requests, such as those exceeding 100 units. Therefore, we formulate the history constraints as a constraint satisfaction problem (CSP) [185]. Consequently, the super-provider history constraints are represented as a triple $\langle X, D, C \rangle$, where X is a set of variables, D is a set of corresponding domains of values, and C is the super-provider's constraint set. The formulation of the super-provider history constraints is as follows:

$X : y_i$: A variable that equals 1 if provider i meets the constraints, and 0 otherwise.

D : The domain of y_i is binary, i.e., 0, 1.

C : Location constraint (c_L): $\forall S \in P.\mathcal{H} : c_L = 1 \Rightarrow S.l = L$, where L is a specific microcell.

Time constraint (c_T): $\forall S \in P.\mathcal{H} : c_T = 1 \Rightarrow S.d = D$, where D is a specific time interval.

Energy constraint (c_E): $\forall S \in P.\mathcal{H} : c_E = 1 \Rightarrow S.a \geq A$, where A is a specific energy service size.
(12.11)

The goal is to find an assignment for the variable y_i that satisfies all the constraints, effectively identifying the providers' history that meets the super-provider's specific constraints. Our CSP formulation is designed to be highly flexible and adaptable to various super-provider preferences. By allowing super-providers to select any combination of constraints, we provide a tailored trust assessment that meets their requirements.

This flexibility enables our model to cater to diverse scenarios and applications, ensuring that trust assessment remains relevant and informative in a wide range of contexts.

- **Trust Assessment Constraints:** The super-provider's trust constraints are related to the trust assessment attributes and the overall trust score. These constraints are formulated as a tuple of $\langle W, \alpha, ae \rangle$ where:
 - w represents the weight assigned to each trust attribute based on its importance, with the constraint that $\sum w_i = 1$ to normalize the impact of each factor on the overall trustworthiness score,
 - α is a threshold for the provider trust score. α is determined based on the super-provider preference. The super-provider might adopt a greedy approach, aiming to maximize energy without considering the providers' trustworthiness. Conversely, they could be risk-averse, prioritizing trust over the amount of provided energy. Alternatively, the super-provider may take a neutral stance, striking a balance between trustworthiness and the energy provided,
 - ae is the adjusted expectations of a super-provider. Super-providers may have different expectations for assessing provider trustworthiness based on their microcell needs. For instance, a provider with a low trust score may be considered trustworthy by a super-provider if they need 50 units, and the provider has historically delivered that amount. Super-providers can adjust their expectations according to their requirements, with various parameters such as delivery time and failure impact. In this work, we focus on adjusting expectations concerning energy. Consequently, we propose three ways to adjust provider expectations in terms of energy (see Fig. 12.4):
 - * **Advertised amount:** In this setting, a super-provider will assess providers based on the amount of delivered energy against the amount they advertised using Eq. 12.4,
 - * **Capped amount:** In this setting, a super-provider will assess providers based on the amount of delivered energy against the super-provider's needs (*expectedAmount*). For instance, assessing them on delivering 30 units regardless of their service advertisement.

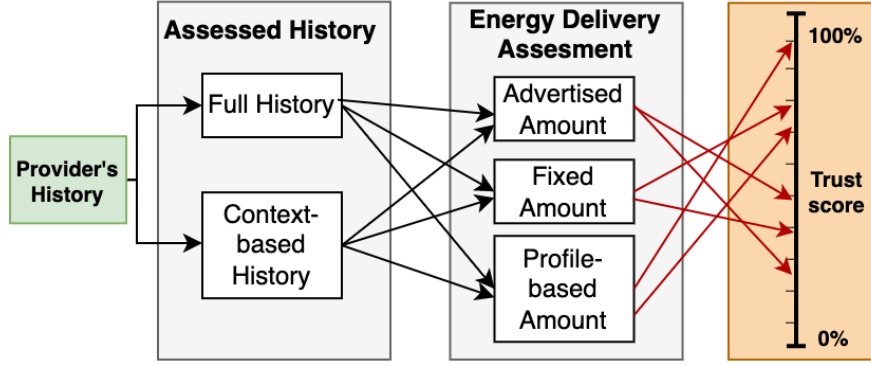


FIGURE 12.4. Context-aware trust assessment model

The needed amount will be fixed for all providers and may depend on the energy demand of the microcell, the available providers, and their trust score. For instance, if a super-provider needs 100 mAh and they have five providers, and their trust score is low, then a possible solution is to assess those providers by looking into their history and the rate for delivering 20 units,

- * Customized amount: In this setting, the super-provider assesses providers using an energy size extracted from their pattern of delivered energy from their history. The reason for adjusting the energy is that sometimes IoT users may overestimate what they can deliver [49]. The customized amount may be computed using statistical values such as mean, mode, and median, and it will be unique to each provider based on their profile.

If the super-provider wants to assess the providers on delivering a fixed amount or a profile-based amount as *expectedAmount* regardless of their service advertisement. In such a scenario, the denominator in Eq. 12.4 will be *expectedAmount*, and the DS_P score will be 1 if the amount is provided is larger than *expectedAmount*.

12.5.1.2 History Filtering:

As previously mentioned, the first step of the trust assessment involves filtering the providers' history. We employ the history constraints of the context model as a filter to refine the providers' history. If a provider lacks sufficient history after applying the constraints, we can gather their history from microcells with similar contexts to our microcell. However, this approach is beyond the scope of this work. In cases where there is insufficient history (i.e., the number of historical records is less than a predefined threshold), we use the provider's original history for assessment.

12.5.1.3 Trust Assessment:

This step uses the proposed trust model to assess the trust of the providers after refining their history in the previous step. The input for this step is the trust constraints from the context model. Note that any combination of history and energy constraints may be applied based on the super-provider requirements (see Fig. 12.4). For example, a super-provider may need to use the full history and a fixed amount assessment. The output is each provider's trust score $\mathcal{P}_{trust}$, calculated using Eq. 12.10 and the weights defined in the context model W . $\mathcal{P}_{trust}$ is then employed to identify the most trustworthy providers that exceed the super-provider's trust score threshold α .

12.5.2 Trust-based Energy Services Composition

This phase aims to compose the most trustworthy energy services to ensure QoE. Given a set of providers and their trust score, we may utilize any priority-based spatio-temporal composition [24][10][1] or other resource allocation algorithms for service provisioning such as Max-Min, knapsack, and genetic algorithms. However, due to the scarcity of energy, a super-provider may end up having a pool of providers with lower trust scores. Intuitively, a super-provider may allocate extra providers as a backup. Hence, we propose a heuristic approach that selects additional services to ensure sufficient energy services. Our approach reduces providers' services based on their trust

scores and identifies complementary services as backups if the original providers do not fulfill their commitments. There are other ways to over-provision by increasing the amount of energy demand based on the available providers and their trust score. We leave it for future work to estimate the optimal increase in energy demand using other techniques.

12.5.3 Assessing Quality of Experience and Cost

The super-provider assesses the QoE of each proposed composition in this phase. The QoE is computed using the model discussed in Subsection 12.3.3. The assessment of QoE gives an indicator of consumers' satisfaction in the microcell.

12.6 Evaluation

We investigated the effectiveness and efficiency of the proposed composition approaches based on a comprehensive set of experiments. This section presents a description of the dataset used in the experiments, followed by a description of the experiment setup and a discussion of the findings.

12.6.1 Dataset Description

We used a real dataset from an app developed in [6]. The app monitors the wireless energy-sharing process using coils connected to two smartphones. The consumer determines the granularity of the monitoring time. The app allows users to request energy from nearby smartphones by size, e.g., 1,000 mAh, or by time, e.g., to charge for 5 minutes. The dataset consists of energy transfer records between a provider (smartphone) and a consumer (smartphone). The records include provider ID, consumer ID, transaction date, time, energy service amount, request amount, and transfer duration. We used the energy dataset to generate QoS parameters for the energy services and requests. For example, the amount of wireless charging transfer in mAh defines the

amount of requested/advertised/provided energy. Additionally, the energy dataset records of the wireless charging transfer duration were used to define the end time of each request/service.

We augmented the energy sharing dataset to mimic the crowd's behavior within microcells by leveraging a dataset published by IBM for a coffee shop chain with three branches in New York City¹. The dataset consists of transaction records of customer purchases in each coffee shop over one month. On average, each coffee shop has 560 transactional records per day and 16,500 transaction records in total. We used the IBM dataset to simulate the spatio-temporal features of energy services and requests. Our experiment employs the consumer ID, transaction date, time, location, and coffee shop ID from each record in the dataset to define the spatio-temporal features of energy services and requests, e.g., the start and location of an energy service or a request. We randomly generate services cancellation to create untrustworthy providers. Table 12.1 presents the experiment parameters and statistics.

12.6.2 Evaluation of the Composition Framework

We compare the proposed heuristic-based composition approached with a baseline traditional resource allocation algorithms, namely first come first served allocation (*Greedy*), and a priority-based allocation algorithm (*Priority-based*) [95]. In *Greedy*, services are processed based on their start time regardless of their trust score. In *Priority-based*, services are processed based on the trust score. We also compared our approach with the knapsack-inspired service composition method (*knapsack-based*) proposed by [23]. The approach selects services that maximize the minimum trust value of the participating providers. We also attempted to implement a brute force approach; nonetheless, its substantial computational and memory requirements made effective execution impossible. We evaluate the effectiveness of each approach by measuring the quality of experience. Furthermore, we evaluate the efficiency of each

¹<https://ibm.co/2O7IvxJ>

TABLE 12.1. Experiments Variables

Variables	Values
Total energy service records for all coffee shops	49894
Energy providers	2248
Duration of all energy services	10 - 30 minutes
Energy services amount	150 - 300 mAh
Energy demand amount	500 - 2500 mAh
Time window	2 hours

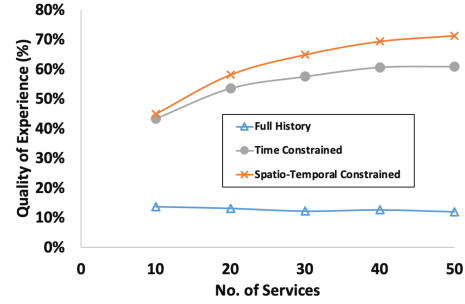


FIGURE 12.5. The average of QoE using different history

method by comparing their execution times. We ran the approaches in different settings by fixing the energy demand and gradually changing the number of services over the time interval T . We repeated the experiment 1,000 times at each point and considered the average value for each approach.

12.6.2.1 Quality of Experience Evaluation

The first experiment evaluated the impact of the context model's history constraints on QoE. As mentioned earlier, QoE reflects consumer satisfaction over time, and a high QoE for a composition indicates a greater degree of consumer satisfaction. Fig. 12.5 displays the average QoE using the *knapsack-based* approach with different constraints, i.e., full history, time constraints representing the history filtered based on the energy demand's time interval, and spatio-temporal constraints assessing trust using the history at the time interval and the given microcell location. We set the energy demand in this experiment to 1,000 mAh. Fig. 12.5 shows that using providers' full history yielded the lowest QoE compared to the context-aware filtered history. This is because assessing trust based on the full history results in many low-trust providers not being used. However, the context-aware trust assessment focuses on their behavior in the microcell. This led to better assessment and provided more services to allocate, ensuring a higher QoE.

The second experiment examined the impact of the context model's trust assessment constraints on QoE. Fig. 12.6 displays the average QoE using the *knapsack-based*

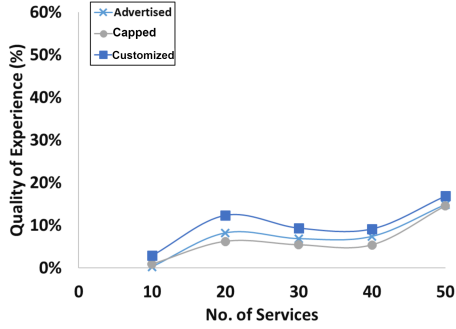


FIGURE 12.6. The average of QoE using different trust assessments

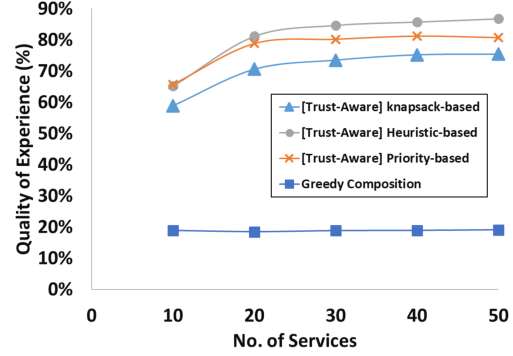


FIGURE 12.7. The average of QoE in trustworthy environment

approach with different expected energy assessments: "advertised" assesses providers based on the service amount advertised in their history record, "capped" assesses them based on a fixed amount determined by the super-provider (e.g., 50 units), and "customized" is a profile-based assessment. In the customized assessment, we evaluated providers based on the median of their historical records. Fig. 12.6 shows a slight QoE improvement when assessing providers based on the median; further experiments are needed to understand the consistency in providers' patterns and the sufficient number of records for profiling a user. We plan to explore the provider's patterns in the future.

The third experiment evaluated the effectiveness of our proposed heuristic-based composition. We compared our approach with the aforementioned approaches in three environment settings: a trustworthy environment where most of the provider's trust score is high, i.e., above 80% (see Fig. 12.7); a neutral environment where providers' trustworthiness follows a random distribution (see Fig. 12.8); and an untrustworthy environment where most of the provider's trust score is low, i.e., below 20% (see Fig. 12.9). Our proposed approach performed better than the rest due to its over-provisioning strategy. However, the over-provisioning comes with a higher cost of rewards (see Fig. 12.10). We computed the cost as the price per energy unit multiplied by the amount of provided energy [1]. Another observation is that QoE increases for all approaches except greedy in all settings. This is intuitive as, with the increase in services, there is more energy to fulfill the energy demand and thereby increase QoE.

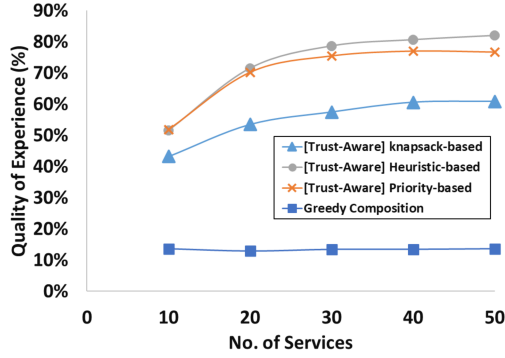


FIGURE 12.8. The average of QoE in neutral environment

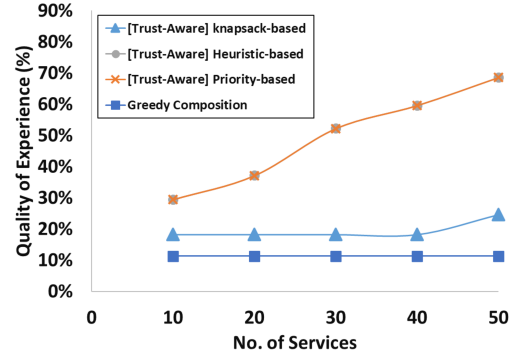


FIGURE 12.9. The average of QoE in untrustworthy environment

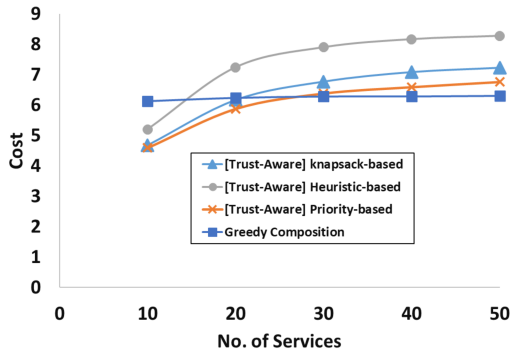


FIGURE 12.10. The average of cost

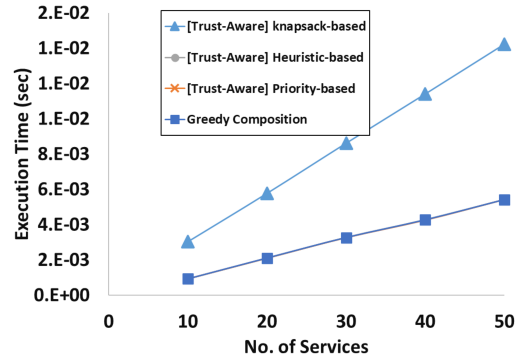


FIGURE 12.11. The average of computation cost

12.6.2.2 Computation Efficiency Evaluation

In the fourth experiment, we assessed the computational cost of all approaches. The execution time for all approaches increases with the increase in services' availability (see Fig. 12.11). This is due to the increase in processing time to assign these services.

Lessons Learned:

From the experiments conducted, we learned that context-aware trust assessment is crucial in achieving higher QoE for the energy-sharing ecosystem. Additionally, filtering history based on the microcell context and adjusting expectations based on the

super-provider's needs resulted in better trust assessment and improved QoE. Furthermore, our heuristic-based composition outperformed other approaches in various settings, demonstrating the effectiveness of over-provisioning strategies; however, it comes with additional costs in terms of rewards paid by the super-provider. Lastly, the execution time for all approaches increased with the availability of services, highlighting the importance of optimizing computational costs.

12.7 Conclusion

We proposed an IoT energy service provisioning trust-based framework to ensure consumers' Quality of Experience (QoE). A key aspect of this framework is the novel context-aware trust assessment model. The model effectively evaluates the trustworthiness of providers while adapting to the dynamic nature of energy service providers and the super-provider's need to maintain QoE. This model assesses providers' trustworthiness in various contexts, considering their behavior and energy provisioning history. Moreover, a trust-adaptive composition technique is proposed for optimal energy allocation, ensuring efficient energy service provisioning. We conducted a set of experiments to assess the performance of the proposed framework. Experiments showed that filtering history based on the microcell context and adjusting expectations based on the super-provider's needs resulted in better trust assessment and improved QoE. The future direction is to consider the probability of change in the energy demand.

Conclusion and Future Work

This thesis maps out several strategies to leverage the service paradigm to utilize IoT energy services in smart cities. First, we thoroughly explored the topic of Crowdsourcing energy services, drawing insights from three distinct fields: the crowdsourcing of IoT services, wireless energy crowdcharging, and the selection and composition of services. We dedicated time to reviewing and analyzing the relevant studies within these areas. We also presented the existing literature in the area of quality of experience. Then, we propose an incentive model to address providers' resistance to participate in Chapter 3. We also proposed an incentive-based energy requests allocation approach.

Chapter 4 presents a reliability model to evaluate energy service requests. The proposed model is designed based on consumers' mobility patterns and the history of energy requests. Adaptive and priority composition approaches are presented to allocate the most reliable energy requests. Selecting reliable consumers ensures the maximum incentives for providers.

In Chapter 5, we presented a platform demonstrating the wireless energy transfer between smartphones. Moreover, the platform collects data from the wireless energy transfer process. Additionally, we presented a wireless energy estimation framework that predicts the actual received energy. We also proposed an activity-based profiling framework to cluster users based on their activity. These clusters will be used in the future to estimate harvested kinetic energy. In addition, we conducted a study on the impact of incentives on consistent activity reporting.

Chapter 6 explores the impact of wireless energy services on the delivery time of drone swarm delivery. We propose a novel framework for swarm-based drone delivery with in-flight energy sharing. A *support drone* is used to in-flight charge other drones in the swarm. We also proposed Priority-based and Fairness-based energy composition approaches to utilize the available energy.

In Chapter 7, we propose a flow-based energy service composition framework to match *multiple* IoT energy services to cater to *multiple* energy requests. A modified Maximum Flow matching algorithm is proposed to efficiently provision energy services to fulfill multiple energy requests. Moreover, we extend our proposed approach to considering the *partial* provisions of services when request and services *partially overlap* in terms of time.

Chapter 8 proposed a *recommendation-based* approach to proactively distribute energy providers to balance energy distribution in a smart city's microcells. Our approach utilizes the mobility patterns of providers as a tool to identify their spatio-temporal provisioning preferences. The proposed approach leverages the provider's preferences to *reduce the cost of incentives* while effectively distributing them.

In Chapter 9, a novel *Quality of Experience (QoE)* model is developed to assess consumers' satisfaction with energy services. We also proposed two QoE-based service composition approaches to ensure the QoE for consumers. Both approaches leverage the providers' *flexibility* and *shareable* nature of energy services to allocate services and optimize the QoE. A set of extensive experiments is conducted to evaluate the proposed approaches' efficiency and effectiveness.

Chapter 10 extends the previous QOE framework by proposing a *CP-Net-based* provisioning preferences model. The model captures providers' provisioning preferences to rank them. We also presented preference-aware approaches to ensure the QoE for consumers. The approaches try to align the providers' preferences with the energy demand. Moreover, the Heuristic-based approach minimizes the cost of service provisioning.

In Chapter 11, we extend the previous preference-aware QoE-based framework by considering the constraints of provisioning energy services. We propose a novel *PCP-Net-based preference model* to capture providers' fine-grained provisioning preferences and availability. The model is used to assess the certainty of the provider's provisioning commitment. A new Mixed-Integer Linear Programming (MILP) formulation for QoE-based energy service provisioning was presented. Several solvers were used to find the optimal service composition.

Chapter 12 proposed a *trust assessment* model to assess energy providers. The proposed model evaluates providers' trustworthiness in various contexts, considering their behavior and energy provisioning history. Additionally, a trust-adaptive composition technique is presented for optimal energy allocation.

In conclusion, we envision a sustainable ecosystem that allows on-the-go wireless energy crowdsourcing to recharge IoT devices in smart cities. We addressed challenges from both the provider's and super-provider's perspectives. In particular, we investigate composing energy requests for a provider in a crowdsourced IoT environment. In that respect, we addressed provider resistance and consumers' reliability. Moreover, we present several demos to prove the feasibility of deploying energy services. Additionally, we studied the impact of energy services composition in drone swarm delivery. From the super-provider's perspective, we focused on efficient strategies to allocate energy services that maximize energy utilization and Quality of Experience (QoE). We also proposed using energy services as a tool to ensure consumers' quality of experience and address related challenges.

13.1 Open Challenges and Future Directions

To unlock the full potential of crowdsourced energy services, multiple challenges must be tackled. These challenges can be classified into three main categories enabling technologies, human-in-the-Loop, and system deployment. The following subsections describe the challenges encountered in each category and propose possible solutions.

13.1.1 Enabling Technologies:

The deployment of a crowdsourcing energy services ecosystem highly relies on the current Wireless Power Transfer (WPT) technologies. Although WPT technologies are advancing rapidly, technological constraints impede the ecosystem from being fully deployed [186]. Challenges posed by technology include energy loss, miniature energy delivery, and restricted power transmission range. Each of these constraints is discussed further below.

13.1.1.1 Energy Loss

Using the recently developed wireless power transfer technologies, consumers may spend more energy to receive the requested energy [5]. The consumed energy may include the energy required for service discovery, energy delivery, and connection establishment between the consumer and the provider [69]. The huge energy loss impedes the energy services ecosystem from fulfilling its purpose of charging nearby IoT devices. Therefore, minimizing energy loss is a challenge yet to be addressed.

13.1.1.2 Miniature Delivered Energy

Even though there is rapid development in wireless power transfer technologies, the amount of delivered energy is insufficient compare to that which is currently required [32]. The small amount of delivered energy may be justified by the energy loss in delivering the energy service as aforementioned. Hence, delivering a considerable amount of energy wirelessly over a distance is still a challenge that hinders the full deployment of the energy services ecosystem.

13.1.1.3 Limited Power Transfer Range

Near-field magnetic coupling is an efficient technology to transfer energy wirelessly [32]. However, the magnetic coupling has a short power transfer distance. The limited transfer distance restricts the spatial freedom we envision as part of this ecosystem.

13.1.2 Human-in-the-Loop

Crowdsourcing IoT energy services is a human-centered application that depends on their participation behavior. Modeling human behavior is challenging as it requires modeling complex behavioral, psychological, and physiological characteristics of human nature [187]. In our environment, the human factor has several influences on the ecosystem. First, the deployment of the ecosystem depends on the IoT users' willingness to participate in the energy-sharing process [1]. Second, the mobility of IoT users in this dynamic environment could impede the delivery of energy services [50]. Third, energy services are offered by IoT devices that are simultaneously in use by their owners. Therefore, the consistent wireless energy delivery from one IoT device to another depends on the device owners' usage frequency. Finally, the spatial and temporal preferences of IoT users differ, which affects the availability of energy services. In the following, we discuss each of the aforementioned challenges.

13.1.2.1 Provisioning Resistance

Individuals are likely motivated to engage in the energy services ecosystem since they can easily access nearby energy resources. However, since energy is a scarce resource, they may be reluctant to share it [1]. Service resistance is the reluctance to provide services due to limited resources [188]. In the energy services context, service resistance refers to the unwillingness to provide energy to IoT devices [1]. A provider's resistance is influenced by the provider's available energy and the size of the requested energy. As a result, one of the challenges is to predict providers' resistance to offering their energy service and devise a correct incentive to overcome it. Since humans get impacted differently by different incentives, determining the effective incentive for energy providers is a challenge that still needs to be addressed. While we proposed an incentive model in this thesis, further analysis and experiments on real energy services experience are yet to be addressed and validated.

13.1.2.2 Modeling Human Mobility

IoT devices change their locations regularly according to the mobility of their owners. The mobility of the crowd has multiple impacts on the crowdsourcing ecosystem. First, the mobility of the crowd across microcells determines the availability of energy services and requests. Second, the IoT users' mobility affects the connectivity between IoT devices. Highly moving IoT devices within a microcell may result in disconnecting the wireless energy transfer between them. A disconnection in the delivery of the energy transfer may result in its failure [2][50]. Therefore, the mobility of IoT users should be modeled and analyzed to manage the energy services in the ecosystem better. The mobility model of IoT users should be studied and investigated on a micro level as indoor mobility and a macro level as mobility patterns across microcells. The indoor mobility models should reflect the users' movements within a confined space [189]. Indoor mobility models will enable the ecosystem to filter services and requests from a spatial perspective within the power transfer range. Moreover, it will allow the ecosystem to predict users' movement and their impact on disconnecting a service delivery. On the other side, the outdoor mobility pattern of an IoT user may be used to proactively plan for them when to request energy [39]. Furthermore, it enables us to understand the crowd movement and predict the undersupplied areas [190]. Thus, mobility models are needed to unlock the full potential of the ecosystem.

13.1.2.3 Modeling Usage Patterns

The energy usage pattern of providers may impact the consistency of provisioning energy services [49]. An energy provider with highly dynamic usage of their device impacts the delivered energy service quality (QoS). Providers may offer their spare energy while simultaneously using it for their devices. If a provider is heavily using their device, they may end up consuming their advertised service while sharing it. Thus, energy services may fluctuate due to the providers' usage. As a result, IoT devices' energy usage behavior must be modeled to ensure consistent provision of energy services [189][190].

13.1.2.4 Determining Services Availability

Matching energy services and requests in a crowdsourced environment is challenging due to the uncertain availability of energy services and requests [191]. Predicting the availability of energy services is challenging as it relies on detecting the mobility patterns of IoT users and their energy usage patterns. Defining the mobility patterns mainly focuses on analyzing the spatio-temporal attributes and potential regularities hidden in individual movement trajectories [192]. However, the existing literature to determine mobility patterns has low accuracy due to the *flexible nature* of human movement [190]. Thus, the challenge of detecting human mobility patterns hinders the prediction of service availability.

13.1.3 Deployment of Energy Services Ecosystem

The deployment of the energy crowdsourcing ecosystem raises several challenges. In what follows, we discuss each challenge and its impact on the ecosystem.

13.1.3.1 Reliability

Considering reliability in the energy ecosystem encourages the participation of energy users in crowdsourcing energy services [2][23]. The reliability of an energy provider refers to the probability that an energy service will be successfully delivered with the advertised Quality of Service (QoS) attributes. The reliability of the energy provider is impacted by their usage pattern and mobility. As aforementioned, a provider that heavily uses their device might consume their advertised energy. Also, a highly mobile provider will cause frequent disconnections in the energy transfer between the devices; similarly, the reliability of an energy consumer is impacted by their mobility patterns. Like providers, a highly mobile consumer will cause frequent disconnections in the energy transfer between the devices. While we addressed the reliability of consumers, the reliability of providers should be addressed as the factors that impact it are different, such as provider usage.

13.1.3.2 Security

The goal of IoT device security is to prevent and protect against IoT attacks and service failures [193]. Attacks on IoT devices may compromise the privacy and confidentiality of users, infrastructures, data, and IoT devices [194]. Furthermore, IoT attacks such as denial-of-service attacks may impede the delivery of services to IoT Users. Securing IoT systems is a critical challenge because users may not adopt many IoT systems if they are not secure [195]. Existing security architectures and protocols are difficult to integrate with IoT devices due to their limited computing power and storage size [196].

13.1.3.3 Privacy

The global digital data generated by IoT devices is expected to reach 180 Zettabytes by 2020 [197]. As a result, protecting IoT data and the privacy of users who generate or consume data has become a major concern in research and industry [198]. Privacy refers to “The claim of individuals, groups, or institutions to determine for themselves when, how and to what extent information about them is communicated to others” [199]. Maintaining privacy in IoT ecosystems is challenging, as it conflicts with the need to use IoT data to achieve their functions. The data utilization may invade the privacy of IoT users [200]. For example, IoT data are required to enhance the energy crowdsourcing process by profiling the energy providers and consumers. The collected IoT data may include the IoT users’ mobility behavior and their energy consumption model. Collecting such personal information might violate the privacy of IoT users. Therefore, maintaining a balance between IoT data protection and use is a continuing challenge.

Bibliography

- [1] Amani Abusafia, Athman Bouguettaya and Sajib Mistry. ‘Incentive-based selection and composition of iot energy services’. In: *2020 IEEE International Conference on Services Computing (SCC)*. IEEE. 2020, pp. 304–311.
- [2] Amani Abusafia and Athman Bouguettaya. ‘Reliability model for incentive-driven iot energy services’. In: *MobiQuitous 2020-17th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*. 2020, pp. 196–205.
- [3] Jiajie Li et al. ‘Activity-based Profiling for Energy Harvesting Estimation’. In: *The International Conference on Information Processing in Sensor Networks*. 2023, pp. 326–327.
- [4] Ruochun Wang et al. ‘The Nudging Effect on Tracking Activity’. In: *Adjunct Proceedings of the 2021 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2021 ACM International Symposium on Wearable Computers*. 2022.
- [5] Jessica Yao et al. ‘Wireless iot energy sharing platform’. In: *2022 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*. IEEE. 2022, pp. 118–120.
- [6] Pengwei Yang et al. ‘Towards peer-to-peer sharing of wireless energy services’. In: *Service-Oriented Computing–ICSOC 2022 Workshops: ASOCA, AIPA, FMCIoT, WESOACS 2022, Sevilla, Spain, November 29–December 2, 2022 Proceedings*. Vol. 13821. Springer Nature. 2023, p. 388.
- [7] Pengwei Yang et al. ‘Monitoring efficiency of iot wireless charging’. In: *2022 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*. 2023.

- [8] Balsam Alkouz, Athman Bouguettaya and Sajib Mistry. ‘Swarm-based drone-as-a-service (sdaas) for delivery’. In: *2020 IEEE International Conference on Web Services (ICWS)*. IEEE. 2020, pp. 441–448.
- [9] Amani Abusafia, Abdallah Lakhdari and Athman Bouguettaya. ‘Flow-based energy services composition’. In: *IEEE Transactions on Services Computing* (2023).
- [10] Amani Abusafia, Athman Bouguettaya and Abdallah Lakhdari. ‘Maximizing consumer satisfaction of IoT energy services’. In: *Service-Oriented Computing: 20th International Conference, ICSOC 2022, Seville, Spain, November 29–December 2, 2022, Proceedings*. Springer. 2022, pp. 395–412.
- [11] Amani Abusafia, Athman Bouguettaya and Abdallah Lakhdari. ‘Quality of experience optimization in iot energy services’. In: *2022 IEEE International Conference on Web Services (ICWS)*. IEEE. 2022, pp. 91–96.
- [12] Amani Abusafia et al. ‘Context-Aware Trustworthy IoT Energy Services Provisioning’. In: *International Conference on Service-Oriented Computing*. Springer. 2023, pp. 167–185.
- [13] Amani Abusafia, Abdallah Lakhdari and Athman Bouguettaya. ‘Service-Based Wireless Energy Crowdsourcing’. In: *Service-Oriented Computing: 20th International Conference, ICSOC 2022, Seville, Spain, November 29–December 2, 2022, Proceedings*. Springer. 2022, pp. 653–668.
- [14] Abdallah Lakhdari, Amani Abusafia and Athman Bouguettaya. ‘Crowdsharing wireless energy services’. In: *2020 IEEE 6th International Conference on Collaboration and Internet Computing (CIC)*. IEEE. 2020, pp. 18–24.
- [15] Andrew Whitmore, Anurag Agarwal and Li Da Xu. ‘The Internet of Things—A survey of topics and trends’. In: *Information systems frontiers* 17.2 (2015), pp. 261–274.
- [16] Sébastien Ziegler et al. ‘Internet of Things and crowd sourcing-a paradigm change for the research on the Internet of Things’. In: *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*. IEEE. USA: IEEE, 2015, pp. 395–399.

- [17] IHS Markit. 'The Internet of Things: a movement, not a market'. In: *IHS Market* 1.1 (2017), p. 1.
- [18] Athman Bouguettaya et al. 'A service computing manifesto: the next 10 years'. In: *Communications of the ACM* 60.4 (2017), pp. 64–72.
- [19] Athman Bouguettaya et al. 'An internet of things service roadmap'. In: *Communications of the ACM* 64.9 (2021), pp. 86–95.
- [20] Karim Habak et al. 'Femto clouds: Leveraging mobile devices to provide cloud service at the edge'. In: *2015 IEEE 8th international conference on cloud computing*. IEEE. 2015, pp. 9–16.
- [21] Mohammed Bahutair, Athman Bouguettaya and Azadeh Ghari Neiat. 'Multi-Use Trust in Crowdsourced IoT Services'. In: *IEEE Transactions on Services Computing* 16.2 (2022), pp. 1268–1281.
- [22] Sean Dieter Tebje Kelly, Nagender Kumar Suryadevara and Subhas Chandra Mukhopadhyay. 'Towards the implementation of IoT for environmental condition monitoring in homes'. In: *IEEE sensors journal* 13.10 (2013), pp. 3846–3853.
- [23] Abdallah Lakhdari et al. 'Composing energy services in a crowdsourced iot environment'. In: *IEEE Transactions on Services Computing* 15.3 (2020), pp. 1280–1294.
- [24] Abdallah Lakhdari and Athman Bouguettaya. 'Fairness-Aware Crowdsourcing of IoT Energy Services'. In: *International Conference on Service-Oriented Computing*. Springer. 2021, pp. 351–367.
- [25] Abdallah Lakhdari, Athman Bouguettaya and Azadeh Ghari Neiat. 'Crowdsourcing energy as a service'. In: *Service-Oriented Computing: 16th International Conference, ICSOC 2018, Hangzhou, China, November 12-15, 2018, Proceedings 16*. Springer. 2018, pp. 342–351.
- [26] Suranga Seneviratne et al. 'A survey of wearable devices and challenges'. In: *IEEE Communications Surveys & Tutorials* 19.4 (2017), pp. 2573–2620.
- [27] Young-Man Choi, Moon Gu Lee and Yongho Jeon. 'Wearable biomechanical energy harvesting technologies'. In: *Energies* 10.10 (2017), p. 1483.

- [28] José Manjarrés et al. ‘Enhancing Kinetic Energy Harvesting-based Human Activity Recognition with Deep Learning and Data Augmentation’. In: *IEEE Internet of Things Journal* 9.10 (2021), pp. 7545–7558.
- [29] Maria Gorlatova et al. ‘Movers and shakers: Kinetic energy harvesting for the internet of things’. In: *IEEE Journal on Selected Areas in Communications* 33.8 (2015), pp. 1624–1639.
- [30] Theofanis P Raptis. ‘Online social network information can influence wireless crowd charging’. In: *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*. IEEE. 2019, pp. 481–486.
- [31] Eyuphan Bulut et al. ‘Is crowdcharging possible?’ In: *27th International Conference on Computer Communication and Networks (ICCCN)*. IEEE. China: IEEE, 2018, pp. 1–9.
- [32] Hao Feng et al. ‘Advances in high-power wireless charging systems: Overview and design considerations’. In: *Transactions on Transportation Electrification* 6.3 (2020), pp. 886–919.
- [33] Jessica Dolcourt. *Over-the-air wireless charging will come to smartphones*. 2019. URL: <https://tinyurl.com/yxj8sfzd>.
- [34] Andre Kurs et al. ‘Wireless power transfer via strongly coupled magnetic resonances’. In: *science* 317.5834 (2007), pp. 83–86.
- [35] Kazuya Sakai et al. ‘Towards Wireless Power Transfer in Mobile Social Networks’. In: *IEEE Transactions on Network Science and Engineering* 9.3 (2021), pp. 1091–1103.
- [36] Vikram Iyer et al. ‘Charging a smartphone across a room using lasers’. In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1.4 (2018), pp. 1–21.
- [37] James C Lin. ‘Wireless power transfer for mobile applications, and health effect’. In: *IEEE Antennas and Propagation Magazine* 55.2 (2013), pp. 250–253.
- [38] EN Baikova et al. ‘Study on electromagnetic emissions from wireless energy transfer’. In: *2016 IEEE International Power Electronics and Motion Control Conference (PEMC)*. IEEE. 2016, pp. 492–497.

- [39] Abdallah Lakhdari and Athman Bouguettaya. 'Proactive composition of mobile iot energy services'. In: *2021 IEEE International Conference on Web Services (ICWS)*. IEEE. 2021, pp. 192–197.
- [40] Mohammed Bahutair, Athman Bouguettaya and Azadeh Ghari Neiat. 'Multi-perspective trust management framework for crowdsourced IoT services'. In: *IEEE Transactions on Services Computing* 15.4 (2021), pp. 2396–2409.
- [41] Pradip Kumar Sharma, Neeraj Kumar and Jong Hyuk Park. 'Blockchain Technology Toward Green IoT: Opportunities and Challenges'. In: *IEEE Network* (2020).
- [42] Rushan Arshad et al. 'Green IoT: An investigation on energy saving practices for 2020 and beyond'. In: *IEEE Access* 5 (2017), pp. 15667–15681.
- [43] Faisal Karim Shaikh, Sherali Zeadally and Ernesto Exposito. 'Enabling technologies for green internet of things'. In: *IEEE Systems Journal* 11.2 (2015), pp. 983–994.
- [44] Ali Eslami Varjovi and Shahram Babaie. 'Green Internet of Things (GIoT): Vision, applications and research challenges'. In: *Sustainable Computing: Informatics and Systems* 28 (2020), p. 100448.
- [45] Sudeep Pasricha et al. 'A survey on energy management for mobile and IoT devices'. In: *IEEE Design & Test* (2020).
- [46] Theofanis P Raptis. 'When wireless crowd charging meets online social networks: A vision for socially motivated energy sharing'. In: *Online Social Networks and Media* 16 (2020), p. 100069.
- [47] Bruno Scrosati and Jürgen Garche. 'Lithium batteries: Status, prospects and future'. In: *Journal of power sources* 195.9 (2010), pp. 2419–2430.
- [48] Aashish Dhungana et al. 'Peer-to-peer energy sharing in mobile networks: Applications, challenges, and open problems'. In: *Ad Hoc Networks* 97 (2020), p. 102029.
- [49] Abdallah Lakhdari et al. 'Elastic composition of crowdsourced iot energy services'. In: *MobiQuitous 2020-17th EAI International Conference on Mobile*

- and Ubiquitous Systems: Computing, Networking and Services*. 2020, pp. 308–317.
- [50] Abdallah Lakhdari and Athman Bouguettaya. ‘Fluid composition of intermittent iot energy services’. In: *2020 IEEE International Conference on Services Computing (SCC)*. IEEE. 2020, pp. 329–336.
- [51] Tiffany Chao et al. *C-flow: visualizing foot traffic and profit data to make informative decisions*. Tech. rep. Technical Report, University of Maryland, 2013.
- [52] Christopher C Muller and Robert H Woods. ‘An expanded restaurant typology’. In: *Cornell Hotel and Restaurant Administration Quarterly* 35.3 (1994), pp. 27–37.
- [53] Conor Muldoon, Michael O’Grady and Gregory O’Hare. ‘A survey of incentive engineering for crowdsourcing’. In: *The Knowledge Engineering Review* 33 (2018), e2.
- [54] Andrea Capponi et al. ‘A survey on mobile crowdsensing systems: Challenges, solutions, and opportunities’. In: *IEEE communications surveys & tutorials* 21.3 (2019), pp. 2419–2465.
- [55] Fen Hou, Yingying Pei and Jingyi Sun. *Mobile crowd sensing: incentive mechanism design*. Springer, 2019.
- [56] Fazlullah Khan et al. ‘Mobile crowdsensing: A survey on privacy-preservation, task management, assignment models, and incentives mechanisms’. In: *Future Generation Computer Systems* 100 (2019), pp. 456–472.
- [57] Guang Yang et al. ‘Promoting cooperation by the social incentive mechanism in mobile crowdsensing’. In: *IEEE Communications Magazine* 55.3 (2017), pp. 86–92.
- [58] Aikaterini Katmada, Anna Satsiou and Ioannis Kompatsiaris. ‘Incentive mechanisms for crowdsourcing platforms’. In: *International Conference on Internet Science*. Springer. 2016, pp. 3–18.

- [59] Zibin Zheng and Michael R Lyu. ‘Personalized reliability prediction of web services’. In: *ACM Transactions on Software Engineering and Methodology (TOSEM)* 22.2 (2013), pp. 1–25.
- [60] Jian An et al. ‘A crowdsourcing assignment model based on mobile crowd sensing in the Internet of Things’. In: *IEEE Internet of Things Journal* 2.5 (2015), pp. 358–369.
- [61] Manoop Talasila, Reza Curtmola and Cristian Borcea. ‘Mobile crowd sensing’. In: *Handbook of Sensor Networking: Advanced Technologies and Applications*. CRC Press Boca Raton, 2015, pp. 3–1.
- [62] Lav R Varshney. ‘Privacy and reliability in crowdsourcing service delivery’. In: *2012 Annual SRII Global Conference*. IEEE. USA: IEEE, 2012, pp. 55–60.
- [63] Talal Halabi and Mohammad Zulkernine. ‘Reliability-Driven Task Assignment in Vehicular Crowdsourcing: A Matching Game’. In: *2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*. IEEE. USA: IEEE, 2019, pp. 78–85.
- [64] Burak Kantarci and Hussein T Mouftah. ‘Mobility-aware trustworthy crowdsourcing in cloud-centric internet of things’. In: *2014 Symposium on Computers and Communications (ISCC)*. IEEE. 2014, pp. 1–6.
- [65] Zhen Cao, Jürgen Fitschen and Panagiotis Papadimitriou. ‘Social Wi-Fi: Hot-spot sharing with online friends’. In: *2015 IEEE 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. IEEE. 2015, pp. 2132–2137.
- [66] Sibel Adali et al. ‘Measuring behavioral trust in social networks’. In: *2010 IEEE international conference on intelligence and security informatics*. IEEE. 2010, pp. 150–152.
- [67] Mohammed Bahutair, Athman Bougeuttaya and Azadeh Ghari Neiat. ‘Adaptive trust: Usage-based trust in crowdsourced iot services’. In: *2019 IEEE international conference on web services (ICWS)*. IEEE. 2019, pp. 172–179.

- [68] Mo Huang, Yan Lu and Rui P Martins. ‘A reconfigurable bidirectional wireless power transceiver for battery-to-battery wireless charging’. In: *IEEE Transactions on Power Electronics* 34.8 (2018), pp. 7745–7753.
- [69] Woongsoo Na et al. ‘Energy-efficient mobile charging for wireless power transfer in Internet of Things networks’. In: *IEEE Internet of Things journal* 5.1 (2017), pp. 79–92.
- [70] Douglas Bell and Michael A Leabman. *Systems and methods for real or near real time wireless communications between a wireless power transmitter and a wireless power receiver*. US Patent App. 10/211,685. 2019.
- [71] Aashish Dhungana, Tomasz Arodz and Eyuphan Bulut. ‘Exploiting peer-to-peer wireless energy sharing for mobile charging relief’. In: *Ad Hoc Networks* 91 (2019), p. 101882.
- [72] Sotiris Nikolettseas, Theofanis P Raptis and Christoforos Raptopoulos. ‘Wireless charging for weighted energy balance in populations of mobile peers’. In: *Ad Hoc Networks* 60 (2017), pp. 1–10.
- [73] Adelina Madhja et al. ‘Peer-to-peer energy-aware tree network formation’. In: *Proceedings of the 16th ACM International Symposium on Mobility Management and Wireless Access*. UK: ACM, 2018, pp. 1–8.
- [74] Aashish Dhungana and Eyuphan Bulut. ‘Energy sharing based content delivery in mobile social networks’. In: *IEEE 20th International Symposium on "A World of Wireless, Mobile and Multimedia Networks"(WoWMoM)*. IEEE. USA: IEEE, 2019, pp. 1–9.
- [75] Shuo Chen et al. ‘Mobile wireless charging and sensing by drones’. In: *14th Annual International Conference on Mobile Systems, Applications, and Services Companion*. 2016.
- [76] Fatih Yucel, Kemal Akkaya and Eyuphan Bulut. ‘Efficient and privacy preserving supplier matching for electric vehicle charging’. In: *Ad Hoc Networks* 90 (2019), p. 101730.

- [77] Dinh Hoa Nguyen. ‘Electric vehicle–wireless charging-discharging lane decentralized peer-to-peer energy trading’. In: *IEEE Access* 8 (2020), pp. 179616–179625.
- [78] Zugang Li, Shi Chen and Buxiang Zhou. ‘Electric vehicle peer-to-peer energy trading model based on smes and blockchain’. In: *IEEE Transactions on Applied Superconductivity* 31.8 (2021), pp. 1–4.
- [79] Rui Zhang et al. ‘Collaborative interactive wireless charging in a cyclic mobispace’. In: *2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)*. IEEE. 2018, pp. 1–10.
- [80] Liangzhao Zeng et al. ‘QoS-aware middleware for web services composition’. In: *IEEE Transactions on software engineering* 30.5 (2004).
- [81] Abderrahmane Maaradji et al. ‘Social-based web services discovery and composition for step-by-step mashup completion’. In: *2011 IEEE International Conference on Web Services*. IEEE. 2011, pp. 700–701.
- [82] Mengyu Sun et al. ‘Energy-efficient IoT service composition for concurrent timed applications’. In: *Future Generation Computer Systems* 100 (2019), pp. 1017–1030.
- [83] Deze Zeng, Lin Gu and Hong Yao. ‘Towards energy efficient service composition in green energy powered Cyber–Physical Fog Systems’. In: *Future Generation Computer Systems* 105 (2020), pp. 757–765.
- [84] Feng Li et al. ‘Service composition and scheduling in cloud-based simulation environment’. In: *Proceedings of the Model-driven Approaches for Simulation Engineering Symposium*. Society for Computer Simulation International. 2018, p. 2.
- [85] Shamsollah Ghanbari and Mohamed Othman. ‘A priority based job scheduling algorithm in cloud computing’. In: *Procedia Engineering* 50.0 (2012), pp. 778–785.
- [86] Stephen S Yau and Ho G An. ‘Adaptive resource allocation for service-based systems’. In: *Proceedings of the First Asia-Pacific Symposium on Internetware*. ACM. 2009, p. 3.

- [87] ChiSung Bae and Dong-Ho Cho. ‘Fairness-aware adaptive resource allocation scheme in multihop OFDMA systems’. In: *IEEE communications letters* 11.2 (2007), pp. 134–136.
- [88] Kaneez Fizza et al. ‘QoE in IoT: a vision, survey and future directions’. In: *Discover Internet of Things* 1 (2021), pp. 1–14.
- [89] Yan Gong et al. ‘Model-based approach to measuring quality of experience’. In: *2009 first international conference on emerging network intelligence*. IEEE. 2009, pp. 29–32.
- [90] Steven Schmidt et al. ‘Assessing interactive gaming quality of experience using a crowdsourcing approach’. In: *2020 Twelfth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE. 2020, pp. 1–6.
- [91] Pengfei Wang, Matteo Varvello and Aleksandar Kuzmanovic. ‘Kaleidoscope: A crowdsourcing testing tool for web quality of experience’. In: *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. IEEE. 2019, pp. 1971–1982.
- [92] George Margetis et al. ‘iEat: An interactive table for restaurant customers’ experience enhancement’. In: *HCI International 2013-Posters’ Extended Abstracts: International Conference, HCI International 2013, Las Vegas, NV, USA, July 21-26, 2013, Proceedings, Part II* 15. Springer. 2013, pp. 666–670.
- [93] *Charging time based on the capacity of the battery*. URL: https://batteryuniversity.com/learn/article/charging_lithium_ion_batteries.
- [94] Xinglin Zhang et al. ‘Incentives for mobile crowd sensing: A survey’. In: *IEEE Communications Surveys & Tutorials* 18.1 (2015), pp. 54–67.
- [95] Robert Kruse and CL Tondo. *Data structures and program design in C*. Pearson, 2007.
- [96] Azadeh Ghari Neiat, Athman Bouguettaya and Sajib Mistry. ‘Incentive-based crowdsourcing of hotspot services’. In: *ACM Transactions on Internet Technology (TOIT)* 19.1 (2019), pp. 1–24.
- [97] T Heatherton, Dianne M Tice et al. ‘Losing control: How and why people fail at self-regulation’. In: *San Diego: Academic* (1994).

- [98] Xiao Lu and others Wang. ‘Wireless charging technologies: Fundamentals, standards, and network applications’. In: *IEEE communications surveys & tutorials* 18.2 (2015), pp. 1413–1452.
- [99] Huoran Li, Xuanzhe Liu and Qiaozhu Mei. ‘Predicting smartphone battery life based on comprehensive and real-time usage data’. In: *arXiv preprint arXiv:1801.04069* (2018).
- [100] Martin Ester et al. ‘A density-based algorithm for discovering clusters in large spatial databases with noise.’ In: *kdd*. Vol. 96. 34. 1996, pp. 226–231.
- [101] Mete Çelik, Filiz Dadaşer-Çelik and Ahmet Şakir Dokuz. ‘Anomaly detection in temperature data using DBSCAN algorithm’. In: *2011 international symposium on innovations in intelligent systems and applications*. IEEE. 2011, pp. 91–95.
- [102] Tianqi Chen and Carlos Guestrin. ‘Xgboost: A scalable tree boosting system’. In: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. 2016, pp. 785–794.
- [103] Fabian Pedregosa et al. ‘Scikit-learn: Machine learning in Python’. In: *the Journal of machine Learning research* 12 (2011), pp. 2825–2830.
- [104] Muhammad Moid Sandhu. ‘Phd forum abstract: Energy harvesting based sensing for the batteryless iot’. In: *IPSN*. IEEE. 2020, pp. 373–374.
- [105] Sara Khalifa et al. ‘Harke: Human activity recognition from kinetic energy harvesting data in wearable devices’. In: *IEEE Transactions on Mobile Computing* 17.6 (2017), pp. 1353–1368.
- [106] Sara Khalifa et al. ‘Energy-harvesting wearables for activity-aware services’. In: *IEEE internet computing* 19.5 (2015), pp. 8–16.
- [107] Jianmo Ni, Larry Muhlstein and Julian McAuley. ‘Modeling heart rate and activity data for personalized fitness recommendation’. In: *The World Wide Web Conference*. 2019, pp. 1343–1353.
- [108] Daniel A Epstein et al. ‘Reconsidering the device in the drawer: lapses as a design opportunity in personal informatics’. In: *Proceedings of the 2016 ACM*

- international joint conference on pervasive and ubiquitous computing*. 2016, pp. 829–840.
- [109] Patrick C Shih et al. ‘Use and adoption challenges of wearable activity trackers’. In: *IConference 2015 proceedings* (2015).
 - [110] Ana Caraban et al. ‘23 ways to nudge: A review of technology-mediated nudging in human-computer interaction’. In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 2019, pp. 1–15.
 - [111] Jonathan Baarsch and M Emre Celebi. ‘Investigation of internal validity measures for K-means clustering’. In: *Proceedings of the international multicference of engineers and computer scientists*. Vol. 1. sn. 2012, pp. 14–16.
 - [112] Yuan-Chi Tseng, Hui-Yen Chang and Shih-Wei Yen. ‘The different effects of motivational messages and monetary incentives on fostering walking behavior’. In: *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*. 2018, pp. 1–6.
 - [113] Ali Hamdi et al. ‘Drone-as-a-service composition under uncertainty’. In: *IEEE Transactions on Services Computing* 15.5 (2021), pp. 2685–2698.
 - [114] Pablo Garcia-Aunon, Juan Jesús Roldán and Antonio Barrientos. ‘Monitoring traffic in future cities with aerial swarms: Developing and optimizing a behavior-based surveillance algorithm’. In: *Cognitive Systems Research* 54 (2019).
 - [115] Mauro S Innocente and Paolo Grasso. ‘Self-organising swarms of firefighting drones: Harnessing the power of collective intelligence in decentralised multi-robot systems’. In: *Journal of Computational Science* 34 (2019).
 - [116] Babar Shahzaad et al. ‘Composing drone-as-a-service (DAAS) for delivery’. In: *2019 IEEE International Conference on Web Services (ICWS)*. IEEE. 2019, pp. 28–32.
 - [117] Jalel Euchi. ‘Do drones have a realistic place in a pandemic fight for delivering medical supplies in healthcare systems problems?’ In: *Chinese Journal of Aeronautics* 34.2 (2021), pp. 182–190.
 - [118] Balsam Alkouz and Athman Bouguettaya. ‘Formation-based Selection of Drone Swarm Services’. In: (2020), pp. 386–394.

- [119] Katsuya Fujii, Keita Higuchi and Jun Rekimoto. ‘Endless flyer: a continuous flying drone with automatic battery replacement’. In: *2013 IEEE 10th international conference on ubiquitous intelligence and computing and 2013 IEEE 10th international conference on autonomic and trusted computing*. IEEE. 2013, pp. 216–223.
- [120] Scott Morton, Ruben D’Sa and Nikolaos Papanikolopoulos. ‘Solar powered UAV: Design and experiments’. In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2015.
- [121] Global Energy Transmission. *IN-FLIGHT WIRELESS CHARGING*. [http :
//getcorp.com/technology-overview/](http://getcorp.com/technology-overview/).
- [122] Ahmed Abed Benbuk, Nour Kouzayha and et al. ‘Leveraging UAVs for Passive RF Charging and Ultralowpower Wake-Up of Ground Sensors’. In: *IEEE Sensors Letters* 4.5 (2020).
- [123] Vishal Sharma et al. ‘LoRaWAN-based energy-efficient surveillance by drones for intelligent transportation systems’. In: *Energies* 11.3 (2018), p. 573.
- [124] Petr Skobelev et al. ‘Designing multi-agent swarm of uav for precise agriculture’. In: *Highlights of Practical Applications of Agents, Multi-Agent Systems, and Complexity: The PAAMS Collection: International Workshops of PAAMS 2018, Toledo, Spain, June 20–22, 2018, Proceedings 16*. Springer. 2018, pp. 47–59.
- [125] Australian Financial Review. *Google spreads wings as drone deliveries up 500%*. <https://www.afr.com/technology/google-spreads-wings-as-drone-deliveries-up-500pc-20200929-p560bw>. 2020.
- [126] Federal Aviation Administration (FAA) in USA. *Package Delivery by Drone (Part 135)*. [https://www.faa.gov/uas/advanced_operations/
package_delivery_drone](https://www.faa.gov/uas/advanced_operations/package_delivery_drone). 2023.
- [127] Stuart Layt. *Drone delivery company confident of taking flight in Queensland, Australia*. <https://www.brisbanetimes.com.au/national/>

- queensland/drone-delivery-company-confident-of-taking-flight-in-queensland-20190731-p52ckx.html. 2019.
- [128] Stephen Wilkerson et al. ‘Aerial swarms as asymmetric threats’. In: *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE. 2016, pp. 381–386.
 - [129] Weisen Shi et al. ‘Drone assisted vehicular networks: Architecture, challenges and opportunities’. In: *IEEE Network* 32.3 (2018), pp. 130–137.
 - [130] Markus Waibel, Bill Keays and Federico Augugliaro. *Drone shows: Creative potential and best practices*. Tech. rep. ETH Zurich, 2017.
 - [131] Boris Galkin, Jacek Kibilda and Luiz A DaSilva. ‘UAVs as mobile infrastructure: Addressing battery lifetime’. In: *IEEE Communications Magazine* 57.6 (2019).
 - [132] Hongbing Wang et al. ‘Web service selection with incomplete or inconsistent user preferences’. In: *Service-Oriented Computing*. Springer, 2009, pp. 83–98.
 - [133] KD Tripathi. *Essentials of medical pharmacology*. JP Medical Ltd, 2013.
 - [134] Raja Naeem Akram et al. ‘Security, Privacy and Safety Evaluation of Dynamic and Static Fleets of Drones’. In: 2017, arXiv–1708.
 - [135] Judy Scott and Carlton Scott. ‘Drone delivery models for healthcare’. In: *international conference on system sciences*. 2017.
 - [136] Dane Bamburry. ‘Drones: Designed for product delivery’. In: *Design Management Review* 26.1 (2015).
 - [137] Connie A Lin et al. ‘Drone delivery of medications: Review of the landscape and legal considerations’. In: *The Bulletin of the American Society of Hospital Pharmacists* 75.3 (2018).
 - [138] Mario GCA Cimino, Alessandro Lazzeri and Gigliola Vaglini. ‘Combining stigmergic and flocking behaviors to coordinate swarms of drones performing target search’. In: *2015 6th International Conference on Information, Intelligence, Systems and Applications (IISA)*. IEEE. 2015, pp. 1–6.

- [139] Elisa Casella et al. 'Mapping coral reefs using consumer-grade drones and structure from motion photogrammetry techniques'. In: *Coral Reefs* 36 (2017), pp. 269–275.
- [140] UM Rao Mogili and BBVL Deepak. 'Review on application of drone systems in precision agriculture'. In: *Procedia computer science* 133 (2018).
- [141] Khin Thida San, Eun Young Lee and Yoon Seok Chang. 'The delivery assignment solution for swarms of UAVs dealing with multi-dimensional chromosome representation of genetic algorithm'. In: *IEEE 7th Annual UEMCON*. 2016.
- [142] Kaya Kuru et al. 'Analysis and optimization of unmanned aerial vehicle swarms in logistics: An intelligent delivery platform'. In: *IEEE Access* 7 (2019).
- [143] Piotr Kardasz et al. 'Drones and possibilities of their using'. In: *Journal of Civil & Env. Engineering* 6.3 (2016).
- [144] Ruben D'Sa et al. 'SUAV: Q-An improved design for a transformable solar-powered UAV'. In: *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2016, pp. 1609–1615.
- [145] Darren Quick. *CyPhy Works' UAVs use ground-based power to stay aloft indefinitely*. May 2015. URL: <https://newatlas.com/cyphy-works-microfilament-uavs/25331/>.
- [146] Christine M Haissig. 'Military formation flight as a model for increased capacity in civilian airspace'. In: *The 23rd Digital Avionics Systems Conference (IEEE Cat. No. 04CH37576)*. Vol. 1. IEEE. 2004, pp. 1–C.
- [147] C Cutts and J Speakman. 'Energy savings in formation flight of pink-footed geese'. In: *Journal of Experimental Biology* 189 (1994).
- [148] Henri Weimerskirch et al. 'Energy saving in flight formation'. In: *Nature* 413.6857 (2001).
- [149] Simone Milani and Alvisio Memo. 'Impact of drone swarm formations in 3D scene reconstruction'. In: *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE. 2016, pp. 2598–2602.

- [150] A Mirzaeinia et al. 'Energy conservation of V-shaped swarming fixed-wing drones through position reconfiguration'. In: *Aerospace Science and Technology* 94 (2019), p. 105398.
- [151] Hamidreza Arasteh et al. 'Tot-based smart cities: A survey'. In: *2016 IEEE 16th international conference on environment and electrical engineering (EEEIC)*. IEEE. 2016, pp. 1–6.
- [152] Martin L Shooman. *Reliability of computer systems and networks*. Wiley Online Library, 2002.
- [153] David A Patterson, Garth Gibson and Randy H Katz. 'A case for redundant arrays of inexpensive disks (RAID)'. In: *Proceedings of the 1988 ACM SIGMOD international conference on Management of data*. 1988, pp. 109–116.
- [154] Edsger W Dijkstra et al. 'A note on two problems in connexion with graphs'. In: *Numerische mathematik* 1.1 (1959).
- [155] Tien-Wen Sung, Linyun Sun and Kuo-Chi Chang. 'Multi-hop route planning based on environment information for path-following UAVs'. In: *Proceedings of the International Conference on Artificial Intelligence and Computer Vision (AICV2020)*. Springer. 2020, pp. 831–839.
- [156] Alireza Karduni, Amirhassan Kermanshah and Sybil Derrible. 'A protocol to convert spatial polyline data to network formats and applications to world urban road networks'. In: *Scientific data* 3.1 (2016).
- [157] Liane Margarida Rockenbach Tarouco et al. 'Internet of Things in healthcare: Interoperability and security issues'. In: *2012 IEEE international conference on communications (ICC)*. IEEE. 2012, pp. 6121–6125.
- [158] George T Heineman, Gary Pollice and Stanley Selkow. *Algorithms in a nutshell: A practical guide*. " O'Reilly Media, Inc.", 2016.
- [159] Dogay Altinel and Gunes Karabulut Kurt. In: *IEEE Internet of Things Journal* 6.6 (2019), pp. 10846–10854.
- [160] Jon Kleinberg and Eva Tardos. *Algorithm design*. Pearson, 2006.

- [161] Bozidar Radunovic and Jean-Yves Le Boudec. 'A unified framework for max-min and min-max fairness with applications'. In: *IEEE/ACM Transactions on networking* 15.5 (2007), pp. 1073–1083.
- [162] Fuat Basık et al. 'Fair task allocation in crowdsourced delivery'. In: *IEEE Transactions on Services Computing* 14.4 (2018), pp. 1040–1053.
- [163] Claude E Shannon. 'A mathematical theory of communication'. In: *Bell system technical journal* 27.3 (1948), pp. 379–423.
- [164] Ke Huang et al. 'Predicting mobile application usage using contextual information'. In: *proceedings of the 2012 ACM conference on ubiquitous computing*. 2012, pp. 1059–1065.
- [165] Azadeh Ghari Neiat and Athman Bouguettaya. *Crowdsourcing of sensor cloud services*. USA: Springer, 2018.
- [166] Dingqi Yang et al. 'Modeling user activity preference by leveraging user spatial temporal characteristics in LBSNs'. In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 45.1 (2014), pp. 129–142.
- [167] Shuiguang Deng et al. 'Constraints-driven service composition in mobile cloud computing'. In: *2016 IEEE International Conference on Web Services (ICWS)*. IEEE. 2016, pp. 228–235.
- [168] Jiayu Zhang et al. 'Who is charging my phone? Identifying wireless chargers via fingerprinting'. In: *IEEE Internet of Things Journal* 8.4 (2020), pp. 2992–2999.
- [169] Jian Pei et al. 'Mining sequential patterns by pattern-growth: The prefixspan approach'. In: *IEEE Transactions on knowledge and data engineering* 16.11 (2004), pp. 1424–1440.
- [170] Kalyanmoy Deb et al. 'A fast and elitist multiobjective genetic algorithm: NSGA-II'. In: *IEEE transactions on evolutionary computation* 6.2 (2002), pp. 182–197.
- [171] Sebastian Möller and Alexander Raake. *Quality of experience: advanced concepts, applications and methods*. Springer, 2014.

- [172] Paul W Farris et al. *Marketing metrics: The definitive guide to measuring marketing performance*. Pearson Education, 2010.
- [173] Sheik Mohammad Mostakim Fattah, Athman Bouguettaya and Sajib Mistry. ‘A cp-net based qualitative composition approach for an iaas provider’. In: *International Conference on Web Information Systems Engineering*. Springer. 2018, pp. 151–166.
- [174] Mohammed Bahutair and Athman Bouguettaya. ‘Blockchain-based Trust Information Storage in Crowdsourced IoT Services’. In: *2021 IEEE International Conference on Web Services (ICWS)*. IEEE. 2021, pp. 608–617.
- [175] Alessio Faccia and Vishal Pandey. ‘Business planning and big data, budget modelling upgrade through data science’. In: *2021 the 6th International Conference on Information Systems Engineering*. 2021, pp. 21–25.
- [176] Sleh El Fidha and Nahla Ben Amor. ‘Probabilistic weighted CP-nets’. In: *Intelligent Decision Technologies: Proceedings of the 7th KES International Conference on Intelligent Decision Technologies (KES-IDT 2015)* 7. Springer. 2015, pp. 159–169.
- [177] Shuai Yuan et al. ‘An empirical study of reserve price optimisation in real-time bidding’. In: *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2014, pp. 1897–1906.
- [178] Pavel Mach and Zdenek Becvar. ‘Mobile edge computing: A survey on architecture and computation offloading’. In: *IEEE communications surveys & tutorials* 19.3 (2017), pp. 1628–1656.
- [179] Soheila Abrishami and Piyush Kumar. ‘Using real-world store data for foot traffic forecasting’. In: *Big Data*. IEEE. 2018, pp. 1885–1890.
- [180] Nick Danenberg et al. ‘Advertising budgeting: a reinvestigation of the evidence on brand size and spend’. In: *Journal of Advertising* 45.1 (2016), pp. 139–146.
- [181] Damien Bigot et al. ‘Probabilistic conditional preference networks’. In: *arXiv preprint arXiv:1309.6817* (2013).
- [182] Robert J Vanderbei et al. *Linear programming*. Springer, 2020.

- [183] Mokhtar S Bazaraa, John J Jarvis and Hanif D Sherali. *Linear programming and network flows*. John Wiley & Sons, 2011.
- [184] Marta C Gonzalez, Cesar A Hidalgo and Albert-Laszlo Barabasi. ‘Understanding individual human mobility patterns’. In: *nature* 453.7196 (2008), pp. 779–782.
- [185] Vipin Kumar. ‘Algorithms for constraint-satisfaction problems: A survey’. In: *AI magazine* 13.1 (1992), pp. 32–32.
- [186] *Wireless Charging Market Size, Share and Growth: Analysis - 2027*. URL: <https://www.alliedmarketresearch.com/wireless-charging-market>.
- [187] Flávia C Delicato and Paulo F Pires. ‘Challenges in developing collaborative IoT systems’. In: *2020 IEEE 6th International Conference on Collaboration and Internet Computing (CIC)*. IEEE. 2020, pp. 25–33.
- [188] Shuiguang Deng et al. ‘Toward mobile service computing: opportunities and challenges’. In: *IEEE Cloud Computing* 3.4 (2016), pp. 32–41.
- [189] Riccardo Menon et al. ‘DIY-IPS: Towards an Off-the-Shelf Accurate Indoor Positioning System’. In: *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*. 2022, pp. 778–780.
- [190] Yisheng Alison Zheng et al. ‘IMAP: Individual huMAN mobility Patterns visualizing platform’. In: *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*. 2022, pp. 797–799.
- [191] Amani Abusafia, Athman Bouguettaya and Abdallah Lakhdari. In: *2022 IEEE International Conference on Web Services (ICWS)*. IEEE. 2022, pp. 91–96.
- [192] Jinzhong Wang et al. ‘Urban human mobility: Data-driven modeling and prediction’. In: *Acm Sigkdd Explorations Newsletter* 21.1 (2019), pp. 1–19.
- [193] Yang Lu and Li Da Xu. ‘Internet of things (iot) cybersecurity research: A review of current research topics’. In: *IEEE Internet of Things Journal* 6.2 (2018), pp. 2103–2115.
- [194] Wan Haslina Hassan et al. ‘Current research on Internet of Things (IoT) security: A survey’. In: *Computer networks* 148 (2019), pp. 283–294.

- [195] Madhusanka Liyanage et al. *IoT Security: Advances in Authentication*. John Wiley & Sons, 2020.
- [196] Jamie Payton et al. ‘Envisioning an Information Assurance and Performance Infrastructure for the Internet of Things’. In: *2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC)*. IEEE. 2018, pp. 266–271.
- [197] Michael Kanellos. *152,000 Smart Devices Every Minute In 2025: IDC Outlines The Future of Smart Things*. Mar. 2016. URL: <https://tinyurl.com/y4w6u2bj>.
- [198] Diego Mendez Mena, Ioannis Papapanagiotou and Baijian Yang. ‘Internet of things: Survey on security’. In: *Information Security Journal: A Global Perspective* 27.3 (2018), pp. 162–182.
- [199] Alan F Westin. ‘Privacy and freedom Atheneum’. In: *New York* 7 (1967), pp. 431–453.
- [200] Kewei Sha et al. ‘On security challenges and open issues in Internet of Things’. In: *Future Generation Computer Systems* 83 (2018), pp. 326–337.