Chapter 2

Self-Reduction

2.1 Basic Definitions

Given a rule governing the transformation of a problem instance $\mathcal{I}$, we can obtain a related instance $\mathcal{I}'$ by *self-reduction* using this operation. We will assume that a set of self-reduction operations $\mathcal{A}$ is associated with a set $\mathrm{dom}(\mathcal{A})$ of instances on which $\mathcal{A}$ operates, and that $\mathrm{dom}(\mathcal{A})$ is closed under all of the operations in $\mathcal{A}$. If $\mathcal{A}$ is a set of self-reduction operations for set covering, for example, $\mathrm{dom}(\mathcal{A})$ might be the set of all set systems.

Given a sequence of self-reductions $\alpha = \alpha_1\alpha_2\ldots\alpha_k$ operating on an initial instance $\mathcal{I}$, φ_i^α will denote the operation used by α_i and $\mathcal{I}_i^\alpha$ will denote the instance obtained after the ith self-reduction of α. For convenience, the instance $\mathcal{I}_{|\alpha|}^\alpha$ will be denoted by $\mathcal{I}^\alpha$. If ξ is a single self-reduction, its associated operation will be denoted by φ^ξ. When we say "a sequence of self-reductions α from $\mathcal{A}$", we mean a sequence of self-reductions α such that $\varphi_i^\alpha \in \mathcal{A}$ for all i.

The remainder of this section and the following one is adapted from work in term re-writing. Our notation and terminology follows that of [2].

We will write $\mathcal{I} \to_\mathcal{A} \mathcal{I}'$ to denote the relation "$\mathcal{I}'$ can be obtained from $\mathcal{I}$ by a self-reduction from $\mathcal{A}$". The reflexive, transitive closure of the relation $\to_\mathcal{A}$ will be denoted by $\xrightarrow{*}_\mathcal{A}$, that is, $\mathcal{I} \xrightarrow{*}_\mathcal{A} \mathcal{I}^\alpha$ for any (possibly empty) sequence of self-reductions α from $\mathcal{A}$.

Definition 2.1.1 An instance $\mathcal{I} \in \text{dom}(\mathcal{A})$ is *irreducible* with respect to $\mathcal{A}$ if there is no instance $\mathcal{I}' \in \text{dom}(\mathcal{A})$ such that $\mathcal{I} \rightarrow_{\mathcal{A}} \mathcal{I}'$.

Definition 2.1.2 $\mathcal{A}$ is *terminating* if, for all instances $\mathcal{I} \in \text{dom}(\mathcal{A})$, all self-reduction sequences from $\mathcal{A}$ acting on $\mathcal{I}$ are of finite length.

Definition 2.1.3 Two instances $\mathcal{I}_1$ and $\mathcal{I}_2$ in $\text{dom}(\mathcal{A})$ are *joinable* with respect to $\mathcal{A}$ if there exists an instance $\mathcal{J} \in \text{dom}(\mathcal{A})$ such that $\mathcal{I}_1 \xrightarrow{*} \mathcal{J}$ and $\mathcal{I}_2 \xrightarrow{*} \mathcal{J}$.

If instances $\mathcal{I}_1$ and $\mathcal{I}_2$ are joinable with respect to $\mathcal{A}$, we will write $\mathcal{I}_1 \downarrow_{\mathcal{A}} \mathcal{I}_2$. As a simple example, consider manipulating algebraic expressions. Let $f(x) = (x+1)(x+2)$ and let $g(x) = x(x+3)+2$. Using the distributive law as a "reduction" we can obtain $f_1^\alpha(x) = (x+1)x+(x+1)2$. With two more uses of the distributive law, we get $f_3^\alpha(x) = x^2+x+2x+2$, and finally $f_4^\alpha(x) = x^2+3x+2$. Reducing $g(x)$ similarly, we can obtain $g_1^\beta(x) = x^2 + 3x + 2 = f_4^\alpha(x)$ and so $f(x) \downarrow_{\mathcal{A}} g(x)$, where $\mathcal{A}$ is the set of algebraic manipulations.

2.2 Confluence

Definition 2.2.1 $\mathcal{A}$ is said to be *confluent* if for all instances $\mathcal{I} \in \text{dom}(\mathcal{A})$ and all sequences of self-reductions α and β from $\mathcal{A}$ acting on $\mathcal{I}$, $\mathcal{I}^\alpha \downarrow_{\mathcal{A}} \mathcal{I}^\beta$.

That is to say, if $\mathcal{I}$ can be reduced to two different instances by some operations in $\mathcal{A}$, then these two instances can themselves be reduced to a common instance. Considering again $f(x) = (x+1)(x+2)$, we can obtain either $f_1^\alpha(x) = (x+1)x + (x+1)2$ or $f_1^\beta = x(x+2) + 1(x+2)$, and continuing the "reduction" in the obvious ways we find $f^\alpha(x) = f^\beta(x) = x^2+3x+2$. Furthermore, it is easy to see that the set of algebraic manipulations is confluent.

Confluence leads to the following simple result, which will be important in our study of self-reductions in this thesis.

Theorem 2.1 A terminating set of self-reduction operations $\mathcal{A}$ is confluent if and only if, for all instances $\mathcal{I} \in \mathrm{dom}(\mathcal{A})$, there is exactly one irreducible instance $\mathcal{J} \in \mathrm{dom}(\mathcal{A})$ such that $\mathcal{I} \xrightarrow{*}_{\mathcal{A}} \mathcal{J}$.

Checking confluence does not, in general, appear to be easy. Fortunately, we have some results that will make checking confluence much easier for the self-reductions used in this thesis.

Definition 2.2.2 $\mathcal{A}$ is said to be *locally confluent* if for all instances $\mathcal{I} \in \mathrm{dom}(\mathcal{A})$ and self-reductions ξ, ζ from $\mathcal{A}$ acting on $\mathcal{I}$, $\mathcal{I}^{\xi} \downarrow_{\mathcal{A}} \mathcal{I}^{\zeta}$.

If a set of self-reduction operations is confluent, obviously it is also locally confluent. Checking local confluence is somewhat easier than checking confluence, and fortunately we have *Newman's Lemma* [52]:

Theorem 2.2 If $\mathcal{A}$ is terminating and locally confluent, $\mathcal{A}$ is confluent.

Proof. The proof is by well-founded induction. For the inductive hypothesis, suppose that $\mathcal{I}^{\tau\gamma} \downarrow_{\mathcal{A}} \mathcal{I}^{\tau\delta}$ for all non-empty reduction sequences τ from $\mathcal{A}$ and all reduction sequences γ and δ from $\mathcal{A}$.

Let α and β be sequences of reductions from $\mathcal{A}$ on an instance $\mathcal{I}$. Obviously, $\mathcal{I}^{\alpha} \downarrow_{\mathcal{A}} \mathcal{I}^{\beta}$ if either α or β is empty.

Otherwise, there exists a $\mathcal{J}$ such that $\mathcal{I}_1^{\alpha} \xrightarrow{*}_{\mathcal{A}} \mathcal{J}$ and $\mathcal{I}_1^{\beta} \xrightarrow{*} \mathcal{J}$, since $\mathcal{A}$ is locally confluent. By the inductive hypothesis, $\mathcal{J} \downarrow_{\mathcal{A}} \mathcal{I}^{\alpha}$, and hence there exists a $\mathcal{K}$ such that $\mathcal{J} \xrightarrow{*}_{\mathcal{A}} \mathcal{K}$ and $\mathcal{I}^{\alpha} \xrightarrow{*}_{\mathcal{A}} \mathcal{K}$. Similarly, there exists an $\mathcal{L}$ such that $\mathcal{K} \xrightarrow{*}_{\mathcal{A}} \mathcal{L}$ and $\mathcal{I}^{\beta} \xrightarrow{*}_{\mathcal{A}} \mathcal{L}$. Since $\mathcal{I}^{\alpha} \xrightarrow{*}_{\mathcal{A}} \mathcal{K}$, we see that $\mathcal{I}^{\alpha} \xrightarrow{*}_{\mathcal{A}} \mathcal{L}$. Hence, $\mathcal{I}^{\alpha} \downarrow_{\mathcal{A}} \mathcal{I}^{\beta}$, as required. □

The following theorem will also enable us to check confluence of large sets more easily.

Theorem 2.3 Let $\mathcal{A}$ be a terminating set of self-reduction operations. If for all (not necessarily distinct) $\varphi, \psi \in \mathcal{A}$, $\{\varphi, \psi\}$ is locally confluent, then $\mathcal{A}$ is confluent.

Proof. Let ξ and ζ be reductions from $\mathcal{A}$ acting on an instance $\mathcal{I}$. As $\{\varphi^{\xi}, \varphi^{\zeta}\}$ is locally confluent, there exist sequences α and β from $\{\varphi^{\xi}, \varphi^{\zeta}\}$ (and hence, from $\mathcal{A}$) such that $\mathcal{I}^{\xi\alpha}$ is isomorphic to $\mathcal{I}^{\zeta\beta}$, showing that $\mathcal{I}^{\xi} \downarrow_{\mathcal{A}} \mathcal{I}^{\zeta}$. Since $\mathcal{A}$ is terminating, Theorem 2.2 shows the result. □

In the remainder of this work, we will omit $\mathcal{A}$ from the notation where the set of self-reduction operations is clear from context.

2.3 Self-Reduction and Combinatorial Optimisation

This thesis is concerned with self-reductions that can be used to facilitate solutions to combinatorial optimisation problems.

Consider the satisfiability of an arbitrary Boolean expression, and consider an input expression $f = (x_1 \wedge x_2) \vee (\bar{x}_1 \wedge x_2)$. By considering an application of the distributive law as the first self-reduction of a sequence α, we obtain $f_1^\alpha = (x_1 \vee \bar{x}_1) \wedge x_2$ and, by using the obvious identities, $f_2^\alpha = T \vee x_2$ and $f_3^\alpha = x_2$.

By choosing self-reductions in an appropriate manner, such as the above, f is true if and only if f_3^α is true. However, f_3^α is a simpler expression, requiring only one substitution to test for satisfiability whereas f required four.

In general, we will choose self-reduction rules such that the solution for $\mathcal{I}$ is easily recoverable from, if not the same as, the solution for an $\mathcal{I}'$ such that $\mathcal{I} \to \mathcal{I}'$. At each stage of the self-reduction sequence α, an instruction will be pushed onto a stack allowing a solution for $\mathcal{I}_i^\alpha$ to be recovered from a solution for $\mathcal{I}_{i+1}^\alpha$. Given a solution for a reduced instance $\mathcal{I}^\alpha$, a solution for $\mathcal{I}$ can then be recovered by popping instructions from the stack and applying them until the stack is empty.

Such self-reduction rules are described for a variety of problems in the following chapters of this thesis.

2.4 Decomposition

2.4.1 The Simplicial Decomposition

The simplicial decomposition first appeared in a theorem of Wagner in 1937 [58]. Other authors [29, 60, 57], apparently working independently of Wagner, have also used terms like "decomposition by clique separators". The present work shall follow the terminology of Diestel [18].

A graph is decomposed into two factors by use of a *clique separator*, that is, a clique whose removal causes the graph to become disconnected. Each factor contains one of the components of the separated graph, plus the separator, as shown in Fig. 2.1. Problems can be solved independently on each factor, since the factors are not connected to each other except through the separator. The special structure of the separator permits the solutions to be joined together in a straightforward way. Tarjan [57] describes how to solve the graph colouring, maximum clique and maximum independent set problems in this way. If the separator is a single vertex (a *cut vertex*), it easy to to see that the Steiner problem can be solved separately on each factor by making the cut vertex special.

Though it was known as early as 1962 [59] that certain properties of the graph could be computed by examining the graph's prime factors (that is, the indecomposable factors at the leaves of the tree), it appears that Gavril [29] in 1977 was the first to really consider using the decomposition as a computational tool. Gavril's algorithm for computing the decomposition, however, worked only for certain classes of graphs. An algorithm to decompose arbitrary graphs was proposed by Whitesides [60] in 1981. A faster algorithm, and more comprehensive consideration of its computational uses, was given by Tarjan [57] in 1985. This algorithm was further improved by Leimer [44] in 1993. A comprehensive survey of the decomposition, including both its computational and theoretical uses, is given by Diestel [18].

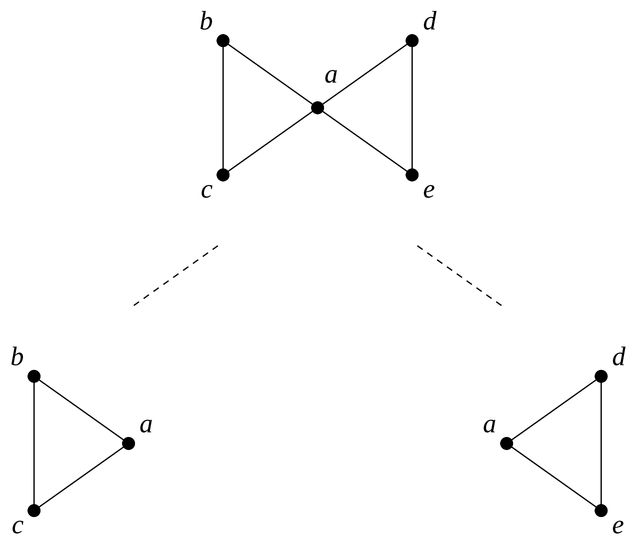

Figure 2.1: A simplicial decomposition of a graph using $\{a\}$ as a clique separator.

2.4.2 The Substitution Decomposition

The substitution decomposition first appeared as a theorem of Gallai in 1967 [26]. It has been the unfortunate result of a frenzy of research in the area that the decomposition and associated concepts have as many names as there are researchers. The present work shall follow the terminology of Möhring and Radermacher [50].

An *autonomous set* of a graph is a subset of the graph's vertices such that if any vertex in the autonomous set is adjacent to a vertex outside the autonomous set, then every vertex in the autonomous set is also adjacent to this outside vertex. A graph is decomposed by partitioning it into its maximal autonomous sets, as shown in Fig. 2.2.

To compute a property of the graph, the property is first computed on all of its factors. A *quotient graph* is then formed by creating a vertex for every autonomous set, with weight equal to the size of the solution for that set. Two vertices are joined by an edge if and only if their corresponding autonomous

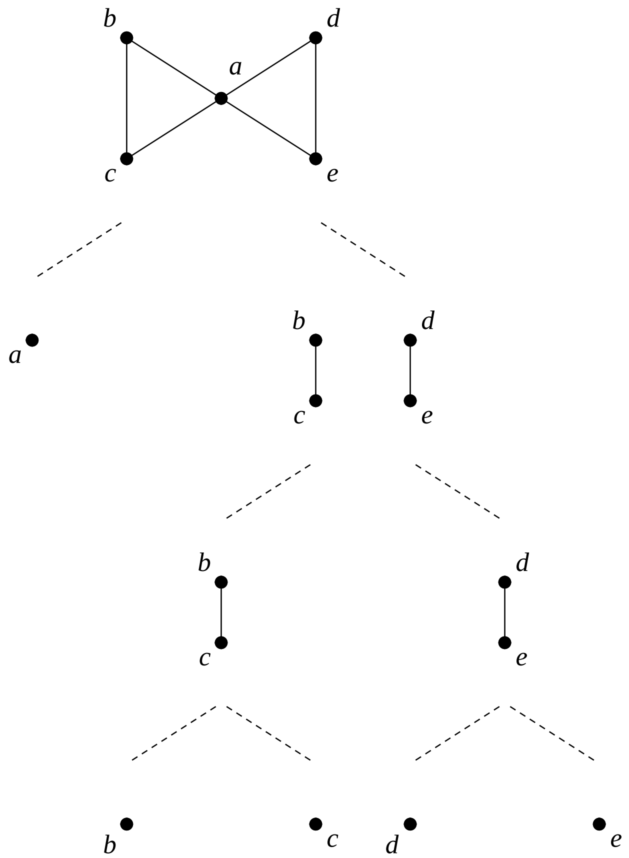

Figure 2.2: The substitution decomposition of a graph. The autonomous sets of the original graph are $\{a\}$ and $\{b, c, d, e\}$.

sets are connected in the original graph. The problem is solved on the quotient graph, and a solution for the original graph is obtained by substituting the autonomous solutions into the quotient solution.

James, Stanton and Cowan [37] proposed an algorithm for computing the decomposition in 1972, and more efficient algorithms were given by Habib and Maurer in 1979 [31] and by Buer and Möhring in 1984 [9]. The latter also gives a survey of problems on discrete structures that can be solved using the decomposition.

The authoritative work on the decomposition is the 1984 work of Möhring and Radermacher [50], who describe the decomposition for a variety of discrete structures including graphs. They demonstrate how to solve the graph colouring, maximum clique and maximum independent set problems on graphs. Solution of the minimum dominating set problem is similar.

Still more efficient algorithms followed in 1989 [51] and 1992 [56]. Finally, two algorithms of optimal efficiency (that is, running in time directly proportional to the size of the graph) were proposed in 1994, one by Cournier and Habib [13] and another by McConnell and Spinrad [48].

Unfortunately, these later algorithms are extremely complicated and all but impossible for a programmer to implement to meet their theoretical time bounds. A simpler optimal algorithm was proposed by Dahlhaus, Gustedt and McConnell in 1997 [17] based on an earlier parallel algorithm of Dahlhaus [16], but this algorithm is still extremely complex.