Available at WWW.MATHEMATICSWEB.ORG POWERED BY SCIENCE @ DIRECT

Journal of Discrete Algorithms 1 (2003) 77–88

JOURNAL OF DISCRETE ALGORITHMS

www.elsevier.com/locate/jda

On reductions for the Steiner Problem in Graphs

Jeffrey H. Kingston [a], Nicholas Paul Sheppard [b,*]

[a] *School of Information Technologies, The University of Sydney, Sydney, Australia 2006*
[b] *School of Information Technology and Computer Science, The University of Wollongong, Australia 2522*

Received 19 October 1999

Abstract

Several authors have demonstrated how reductions can be used to improve the efficiency with which the Steiner Problem in Graphs can be solved. Previous reduction algorithms have been largely ad hoc in nature. This paper uses a theory of confluence to show that, in many cases, all maximal reduction sequences are equivalent, gaining insights into the design of reduction algorithms that obtain a maximum degree of reduction.

Keywords: Reduction; Steiner problem

1. Introduction

The Steiner Problem in Graphs is a well-known problem in network theory that models various problems requiring a network of minimum cost to be constructed between a given collection of points.

Formally, let G be a connected graph with edge-weights, and let $K(G)$ denote a subset of the vertex set of G known as the graph's *special vertices*. A *Steiner tree* for G is a subset T of the edges of G such that T is connected, and every vertex of $K(G)$ is adjacent to at least one edge of T. A *minimum Steiner tree* T for a graph G is a Steiner tree for G such that there is no Steiner tree U for G such that the sum of weights of U is less than that of T.

Computation of the minimum Steiner tree on an arbitrary graph is NP-hard [7], and it is therefore unlikely that an efficient algorithm for solving the Steiner Problem in Graphs exists.

* Corresponding author.
E-mail address: nps@uow.edu.au (N.P. Sheppard).

1570-8667/$ – see front matter
doi:10.1016/S1570-8667(03)00008-X

In view of the computational expense of computing a solution to the problem, several authors have suggested a variety of comparatively inexpensive reduction tests to reduce the size of a Steiner problem before a normal solution algorithm is applied [2–4,6]. For example, edges that are too heavy to be part of a minimum Steiner tree can be deleted from the problem, as can non-special vertices that are too far away from the special vertices. Duin and Volgenant [4] and Duin [3] give empirical results showing that sophisticated reduction tests can have a substantial effect on the size of the graph, and can even solve the problem entirely in many cases (that is, remove all of the non-special vertices, reducing the Steiner Problem to the simple minimum spanning tree problem).

Having processed an input graph by a sequence of reductions, an expensive exact algorithm can be run more efficiently on the reduced graph than it could have been on the original, larger graph. Alternatively, an approximate algorithm may be able to obtain a better solution on the reduced (presumably easier) problem.

Obviously, it is desirable that an instance of the problem be reduced as much as possible. Developing more powerful reduction tests is one way of doing this, and Duin and Volgenant [4] give an exhaustive survey of old and new reduction tests. Duin [3] has very recently presented some improved and new tests.

In addition to developing more reduction tests, given a fixed set of reduction tests ("all reductions known to science", for example), it is desirable to achieve the maximum amount of reduction possible using this set. Previous reduction algorithms have applied tests in a more or less arbitrary fashion, at least insofar as maximality of reduction is concerned.

In this paper, we show how a theory of confluence can be used to identify sets of reductions for which all maximal sequences are equivalent. In this case, a straightforward greedy algorithm can be used to obtain the maximum amount of reduction. This theory is also used to obtain insights into potential algorithms for more difficult sets of reduction tests.

2. Notation

All of our graphs will be finite, simple and undirected. The vertex set of a graph G will be denoted $V(G)$ and edge set by $E(G)$. Each edge $uv \in E(G)$ is associated with a strictly positive weight $c(G, uv)$. If $uv \notin E(G)$, $c(G, uv) = \infty$. If X is a subset of $E(G)$, $c(G, X)$ denotes the sum of the weights of every edge in X. The neighbourhood of a vertex $v \in V(G)$ is denoted by $N(G, v)$.

In diagrams of graphs, all edges will have weight 1 unless otherwise labelled. Special vertices will be represented by filled circles, and non-special vertices by hollow circles.

Given a set $X \in V(G)$, the subgraph induced by the vertices in X will be denoted by $G|X$.

The *length* of a path P is the sum of the weights of the edges contained in P.

Given two vertices u and v, the *distance* $d(G, u, v)$ between u and v is the minimum length of all u–v paths in a graph G. The *distance graph* $D(G)$ is the complete undirected graph on $V(G)$ with edge uv having weight $d(G, u, v)$.

The *bottleneck length* of a path P in a graph G is the maximum weight of any edge on P. Given two vertices u and v, the *bottleneck distance* $b(G, u, v)$ between u and v is the minimum bottleneck length of all u–v paths in G.

A *special path* is a path in $D(G)$ such that all intermediate vertices (if any) are special. For two vertices u and v, the *special distance* $s(G, u, v)$ between u and v is the minimum bottleneck length over all special u–v paths in $D(G)$.

3. Graph reduction

A *reduction* on a graph G is a transformation of G into a related graph G' by application of some well-defined *reduction operation*. By defining appropriate operations, G' will have some relationship to G that can be exploited for computational or other purposes—for example, G' may be smaller than G but still have the same minimum Steiner tree as G.

Consider the portions of a graph shown in Fig. 1. Clearly, a minimum Steiner tree for the top graph will never pass through the non-special vertex, as the edges adjacent to this vertex could be replaced in any Steiner tree by the shorter edge connecting the two special vertices. Hence the graph can be transformed by deleting the non-special vertex and its adjacent edges without altering the minimum Steiner tree of the graph, and the bottom graph is equivalent to the top graph for computational purposes.

If ξ is a reduction that deletes a vertex, this vertex will be denoted by x^ξ. If ξ is a reduction that deletes or contracts an edge, this edge will be denoted by $x^\xi y^\xi$. If α is a sequence of reductions with an initial graph G, G_i^α will denote the graph obtained after applying the first i reductions of α. For convenience, G^α will denote the graph $G_{|\alpha|}^\alpha$.

All of the definitions and results in this section are known from the field of term re-writing, and apply equally to transformations of any structure, such as strings, set systems and Boolean expressions. For simplicity, however, the present discussion will be limited to graphs. The present work's terminology and notation follows that of [1].

Given a set of reduction operations $\mathcal{A}$, let $\to_{\mathcal{A}}$ be a relation such that $G \to_{\mathcal{A}} G'$ if and only if G' can be obtained from G by application of a reduction using an operation from $\mathcal{A}$. The transitive, reflexive closure of the relation $\to_{\mathcal{A}}$ will be denoted by $\xrightarrow{*}_{\mathcal{A}}$. That is, $G \xrightarrow{*}_{\mathcal{A}} H$ if and only if H can be obtained from G by a (possibly empty) sequence of reductions using operations from $\mathcal{A}$.

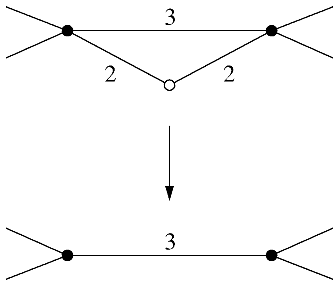

Fig. 1. A simple reduction on a Steiner problem.

Two graphs G and H are said to be *joinable* with respect to $\mathcal{A}$ if and only if there is a graph F such that $G \xrightarrow{*}_{\mathcal{A}} F$ and $H \xrightarrow{*}_{\mathcal{A}} F$. Such a relationship will be written as $G \downarrow_{\mathcal{A}} H$.

A set of reduction operations $\mathcal{A}$ is said be *confluent* if and only if, for all graphs G and reduction sequences α and β using operations from $\mathcal{A}$ on G, $G^{\alpha} \downarrow_{\mathcal{A}} G^{\beta}$. That is to say, if G can be reduced to two or more non-isomorphic graphs by reductions using operations from $\mathcal{A}$, these graphs can always be reduced to a common graph.

A set of reduction operations $\mathcal{A}$ is said to be *terminating* if and only if there is no graph G to which an infinitely long series of reductions using operations from $\mathcal{A}$ can be applied. All of the sets used in the present work are obviously terminating, since the reduction operations all strictly decrease the size of the (finite) graph.

The foregoing definitions lead to the following obvious result.

Theorem 3.1. *A terminating set of reduction operations $\mathcal{A}$ is confluent if and only if for every graph G, there is exactly one irreducible graph R such that $G \xrightarrow{*}_{\mathcal{A}} R$.*

Put another way, if $\mathcal{A}$ is confluent and terminating, then all maximal reduction sequences using operations from $\mathcal{A}$ acting on a graph G result in the same irreducible graph. It follows that, for such a set of operations, a greedy algorithm that simply applies all the reductions it can find until there are no more will find an optimum reduction sequence.

Showing that a set of reduction operations is confluent does not, in general, appear to be easy. A set of reduction operations $\mathcal{A}$ is said to be *locally confluent* if and only if, for every graph G and reductions ξ and ζ using operations from $\mathcal{A}$ on G, $G^{\xi} \downarrow_{\mathcal{A}} G^{\zeta}$. Checking local confluence is easier than showing confluence, and fortunately, we have the following theorem due to Newman [9]:

Theorem 3.2 (Newman's Lemma). *If a terminating set of reduction operations is locally confluent, then it is confluent.*

Proof. The proof is by well-founded induction. For the inductive hypothesis, suppose that $G^{\tau\gamma} \downarrow_{\mathcal{A}} G^{\tau\delta}$ 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 a graph G. Obviously, $G^{\alpha} \downarrow_{\mathcal{A}} G^{\beta}$ if either α or β is empty.

Otherwise, there exists an F such that $G_1^{\alpha} \xrightarrow{*}_{\mathcal{A}} F$ and $G_1^{\beta} \xrightarrow{*}_{\mathcal{A}} F$, since $\mathcal{A}$ is locally confluent. By the inductive hypothesis, $F \downarrow_{\mathcal{A}} G^{\alpha}$, and hence there exists an H such that $F \xrightarrow{*}_{\mathcal{A}} H$ and $G^{\alpha} \xrightarrow{*}_{\mathcal{A}} H$. Similarly, there exists a J such that $H \xrightarrow{*}_{\mathcal{A}} J$ and $G^{\beta} \xrightarrow{*}_{\mathcal{A}} J$. Since $G^{\alpha} \xrightarrow{*}_{\mathcal{A}} H$, we see that $G^{\alpha} \xrightarrow{*}_{\mathcal{A}} J$. Hence, $G^{\alpha} \downarrow_{\mathcal{A}} G^{\beta}$, as required. □

It is obvious that any confluent set of reduction operations is also locally confluent.

In the remainder of the paper, $\mathcal{A}$ will be omitted from the notation where the set is obvious from context.

4. Reductions for the Steiner problem in graphs

Many reductions for the Steiner problem in graphs have been described in the literature. Duin and Volgenant [4] give an extensive overview as well as suggesting some new and improved tests of their own. Very recently, Duin [3] has expanded the array of tests even further; unfortunately, Duin's work came to the attention of the present authors too late to be incorporated into the present work and the following list is based on the older paper. All of the results in this paper are for the reductions listed here.

Proofs for all of the theorems in this section are given in [4] or [5]. Our terminology follows [4].

4.1. Reachability (RT)

Theorem 4.1 [4]. *Let T be a Steiner tree on a graph G, and let x be a non-special vertex of G not adjacent to any edge of T. Let u_1, u_2 and u_∞ be the special vertices of G at, respectively, the least, second-least and most distance from x of all special vertices in G. If $d(G,x,u_1)+d(G,x,u_2)+d(G,x,u_\infty) \geqslant c(G,T)$, then there is a minimum Steiner tree for G in which x is either not part of the tree, or is adjacent to exactly two edges of the tree.*

From the proof of [5], it is easy to see that if x is adjacent to no more than two edges of T and satisfies the other conditions of Theorem 4.1, the same conclusion applies. We will use this slightly improved form of the test.

The problem is reduced by deleting x from the graph, along with all of its adjacent edges. For any distinct vertices $u, v \in N(G,x)$, an edge uv is added with weight $c(G,xu)+c(G,xv)$ if $c(G,xu)+c(G,xv) < c(G,uv)$; if the existing edge is lighter than the proposed edge, the edge is left unchanged. If one of the new edges appears in the minimum Steiner tree of the reduced graph, it is replaced by the two corresponding edges adjacent to x.

4.2. Cut reachability (CRT)

For a vertex u of a graph G, let $\hat{c}(G,u)$ denote $\min\{c(G,uv) \mid v \in N(G,u)\}$, and for a set $X \subseteq V(G)$ let $\hat{c}(G,X)$ denote $\sum_{x \in X} \hat{c}(G,x)$.

Theorem 4.2 [4]. *Let T be a Steiner tree on a graph G, and let x be a non-special vertex of G that is not adjacent to any edge of T. Let u_1 and u_2 be distinct special vertices of G such that $d(G,x,u_1)-\hat{c}(G,u_1)$ and $d(G,x,u_2)-\hat{c}(G,u_2)$ are the smallest and second-smallest, respectively. If $d(G,x,u_1)+d(G,x,u_2)+\hat{c}(G,K(G)-\{u_1,u_2\}) \geqslant c(G,T)$, then there exists a minimum Steiner tree for G not containing x.*

Such a vertex x can simply be deleted from the graph, along with all of its adjacent edges, to form a new graph with the same minimum Steiner tree.

Theorem 4.3 [4]. *Let T be a Steiner tree on a graph G, and let x be a non-special vertex of G that is not adjacent to any edge of T. Let u_1 and u_2 be distinct special vertices of G*

such that $d(G,x,u_1) - \hat{c}(G,u_1)$ and $d(G,x,u_2) - \hat{c}(G,u_2)$ are the smallest and second-smallest, respectively. If there is an edge $xy \in E(G)$ such that $d(G,x,u_1) + c(G,xy) + d(G,x,u_2) + \hat{c}(G,K(G) - \{u_1,u_2\}) \geqslant c(G,T)$, then there is a minimum Steiner tree for G not containing xy.

Such an edge xy can be deleted from the graph to form a new graph with the same minimum Steiner tree.

4.3. Nearest special vertices (NSV)

Theorem 4.4 [4]. *Let xy be an edge of a minimum spanning tree for G, and let u and v be special vertices such that at least one shortest u–v path passes through xy. If $b(G - xy, x, y) \geqslant d(G,u,v)$, then xy belongs to at least one minimum Steiner tree of G.*

If an edge xy of a graph G is identified as belonging to a minimum Steiner tree, G is reduced to the graph G' by contracting x and y into a single special vertex z. For each $u \in N(G',z) = N(G,x) \cup N(G,y) - \{x,y\}$: $c(G',zu) = \min\{c(G,xu), c(G,yu)\}$.

4.4. Smaller special distance (SD)

Theorem 4.5 [4]. *Let xy be an edge of G. If $s(G,x,y) < c(G,xy)$, then there is no minimum Steiner tree for G containing xy.*

4.5. Bottleneck degree (BDk)

Theorem 4.6 [4]. *Let x be a non-special vertex of G with $\deg(G,x) = k$, and let S be the complete undirected graph on $N(G,x)$ with edge $uv \in E(S)$ having weight $s(G,u,v)$. If, for every $Z \subseteq V(S)$, $|Z| \geqslant 3$, and minimum spanning tree T on $S|Z$, $c(S|Z,T) \leqslant \sum_{z \in Z} c(G,xz)$, then there is a minimum Steiner tree for G in which x is adjacent to no more than two edges.*

Of course, the number of minimum spanning trees constructed by this test is exponential in k. For this reason, Duin and Volgenant only apply the test for $k \leqslant 4$.

The problem is reduced to G' by deleting x from the graph, along with all of its adjacent edges. For any distinct vertices $u, v \in N(G,x)$, an edge uv is added with weight $c(G,xu) + c(G,xv)$ if $s(G,uv) \geqslant c(G,xu) + c(G,xv)$. If $uv \in E(G)$, the shorter of the two edges is chosen. If one of the new edges appears in the minimum Steiner tree of the reduced graph, it is replaced by the two corresponding edges adjacent to x.

4.6. Change edge-costs (CEC)

Theorem 4.7 [4]. *Let x be a non-special vertex of a graph G, and let xu and xv be distinct edges of G. Let G' be a graph the same as G except that $c(G',xu) = c(G,xu) + a$ and $c(G',xv) = c(G,xv) - a$ for some positive number a. If xu and xv are both in a minimum Steiner tree for G and G' when x is considered to be special, then a minimum Steiner tree on G' is also a minimum Steiner tree on G.*

Theorem 4.7 does not reduce the problem in itself. However, suppose that the NSV test fails on xu and xv in G, but succeeds in the modified graphs. Then the edge costs can be changed as in the theorem, and reduction can proceed in G'.

Duin and Volgenant's rule proceeds as follows:

Let x be a non-special vertex of G and xu and xv be distinct edges on a minimum spanning tree of G such that u is a special vertex. If the NSV test succeeds on xv on G with x considered special, then xu can have its weight increased by a and xv have its weight decreased by a for $a \leqslant c(G, xv)$ and $a < b(G - xu, x, u) - c(G, xu)$.

5. Maintaining the upper bound

The tree used by Theorems 4.1–4.3 can be obtained by an approximation algorithm for the Steiner Problem in Graphs; e.g., [5] gives a survey of such algorithms. We will assume that any approximate T is minimal, that is, that there is no Steiner tree T' such that $T' \subset T$. Any realistic approximation algorithm will have this property, and, in any case, it is trivial to compute a minimal tree from a non-minimal one.

When we consider the properties of sets containing the RT and CRT reductions, it is necessary to consider what happens to the approximate tree after a reduction. If a new tree were to be computed from scratch after every reduction, it seems unlikely that any confluence (or similar) results could be obtained at all, or, at least, none that were independent of the particular algorithm used for computing the tree. This method of maintaining an upper bound seems highly inefficient and unlikely to be desirable in practice, in any case.

Duin and Volgenant re-compute the upper bound only when the graph can no longer be reduced by the SD, BDk, NSV or CEC reductions. However, it seems profitable both in theory and practice to maintain the upper bound by modifying the tree in a straightforward manner for every reduction performed on the graph, since re-computing the tree from scratch is expensive.

It is trivial to maintain the tree for the RT and CRT reductions, and obviously the reduced tree will be no heavier than the original tree.

If an SD reduction deletes an edge xy from the tree, then the tree will be re-connected by inserting the portion of the minimum special x–y path that crosses between the two components of the tree. It is easy to see that the new upper bound will be strictly less than the old upper bound.

If a BDk reduction deletes a vertex x from the tree, the tree will be re-connected using the special distance spanning tree for all of the vertices adjacent to x. If, for tree-vertices $u, v \in N(G, x)$, uv is an edge of the reduced graph, this edge will be inserted into the tree. Otherwise, the tree will be re-connected as if for an SD reduction of uv. Since this spanning tree must be of less or equal weight to the edges adjacent to x, the upper bound cannot be increased by this procedure. Where x is not part of the tree but modifies an edge that is (that is, uv is an edge of the tree for $u, v \in N(G, x)$), this edge may be replaced by a less expensive edge created by the BDk reduction.

If an NSV reduction contracts an edge that is part of the tree, the tree will be contracted with it. If an NSV reduction causes a cycle to appear in the tree, the cycle will be broken by removing the heaviest edge on this cycle. Again, the new tree will be no heavier than the original tree.

6. Confluence

Theorem 6.1. *The set* $\{\mathrm{RT}\}$ *is confluent.*

Proof. Consider RT reductions ξ and ζ on a graph G. As RT reduction cannot decrease distances in the graph, and cannot increase the weight of the upper bound tree, x^ζ can be deleted from G^ξ by RT reduction to form a graph $G^{\xi\zeta}$. Similarly, a graph $G^{\zeta\xi}$ can be formed from G^ζ, and this graph is isomorphic of $G^{\xi\zeta}$, showing that $G^\xi \downarrow G^\zeta$. Hence $\{\mathrm{RT}\}$ is locally confluent, and confluence follows from Theorem 3.2 since the set is obviously terminating. □

Lemma 6.2. *Let G' be the graph obtained from G by application of a reduction from the set* $\{\mathrm{NSV}, \mathrm{SD}, \mathrm{BD}k, \mathrm{RT}\}$. *For any distinct $u, v \in V(G')$, $s(G', u, v) \leqslant s(G, u, v)$.*

Proof. Lemma 2 of [4] shows this for the NSV, SD and BDk cases. The proof for the RT case is similar to the BDk case. □

Corollary 6.3. *The set* $\{\mathrm{SD}\}$ *is confluent.*

Proof. The proof is similar to that of Theorem 6.1. □

Theorem 6.4. *The set* $\{\mathrm{RT}, \mathrm{SD}\}$ *is confluent.*

Proof. Let ξ and ζ be reductions on a graph G. If ξ and ζ are both RT reductions or both SD reductions, then $G^\xi \downarrow G^\zeta$ from Theorem 6.1 or Corollary 6.3. Otherwise, without loss of generality, let ξ be the RT reduction.

From Lemma 6.2, ζ is a legal reduction in G^ξ, yielding a graph $G^{\xi\zeta}$.

As ζ does not reduce distances in the graph and does not increase the weight of the tree, ξ is a legal reduction on G^ζ unless x^ξ is adjacent to more than two edges of the reduced upper bound tree. As x^ξ was adjacent to no more than two in the original tree, this would imply that ζ caused the tree to be re-connected by joining an edge to x^ξ. But any tree in which x^ξ is adjacent to more than two vertices must have weight at least as great as that of the original tree, contradicting the properties of an SD test performed on an edge of the tree. Hence ξ is a legal reduction in G^ζ, forming a graph $G^{\zeta\xi}$.

$G^{\xi\zeta}$ and $G^{\zeta\xi}$ are obviously isomorphic unless $x^\zeta y^\zeta$ is adjacent to x^ξ in G, that is, without loss of generality, $x^\xi = x^\zeta$. In this case, $G^{\xi\zeta}$ may have extra edges linking y^ζ and other neighbours of x^ξ. Consider such a neighbour z. The edge $y^\zeta z$ has weight $c(G, x^\xi z) + c(G, x^\xi y^\zeta)$. By traversing the minimum special x^ζ–y^ζ path starting at z, we see that $s(G^\xi, z, y^\zeta) \leqslant s(G, x^\zeta, y^\zeta) + c(G, x^\xi z)$. Since $x^\zeta y^\zeta$ was subject to SD in G, $s(G^\xi, z, y^\zeta) < c(G, x^\xi y^\zeta) + c(G, x^\xi, z)$ and hence $y^\zeta z$ is subject to SD deletion in G^ξ. By applying the SD reduction to all such edges, we can obtain a graph isomorphic to $G^{\zeta\xi}$.

Hence $G^{\xi} \downarrow G^{\zeta}$, showing local confluence. The set is obviously terminating, and so confluence follows from Theorem 3.2. □

Note, however, that Theorem 6.4 fails if Duin and Volgenant's original RT reduction is used. In this case, an RT vertex might lie on the minimum special path used to re-connect the tree after an SD reduction, as shown in Fig. 2. Heavy lines show the upper bound tree.

Unlike the reductions mentioned so far, the set {NSV} is not confluent, as demonstrated by Fig. 3. It is easy to check that both of the reduced graphs are irreducible under RT

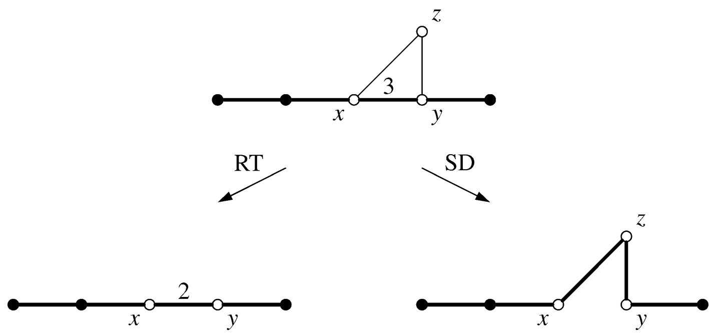

Fig. 2. Two non-isomorphic irreducible graphs produced by Duin and Volgenant's original RT and SD reductions.

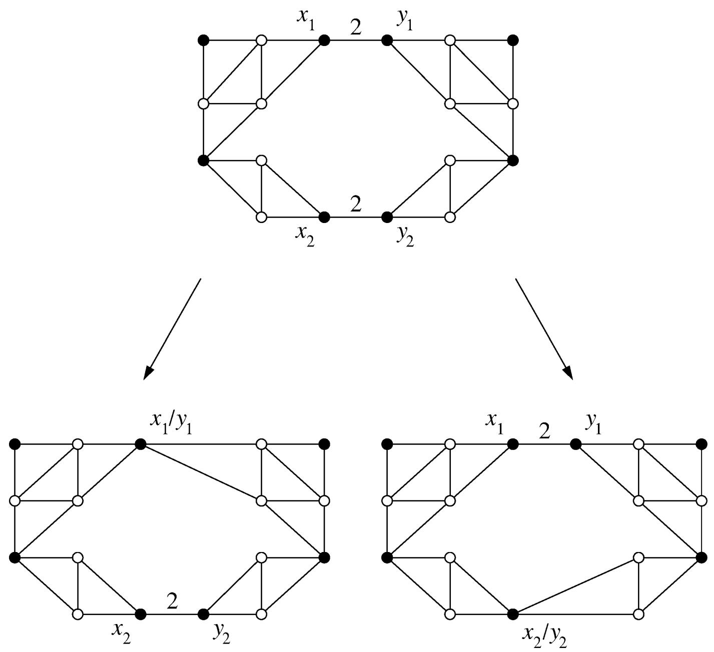

Fig. 3. Two non-isomorphic irreducible graphs produced by NSV reduction.

and SD, and so we immediately obtain the result that the set {RT, SD, NSV} and all of its subsets containing NSV are not confluent.

Note that the graph of Fig. 3 has several distinct minimum spanning trees, and that the two edges subject to NSV reduction lie on different spanning trees. If our graphs are restricted to the set of graphs with a unique minimum spanning tree, however, we obtain a different result.

Lemma 6.5. *If ξ and ζ are distinct* NSV *reductions on a graph G, and G has a unique minimum spanning tree, then $x^\zeta y^\zeta$ is subject to* NSV *reduction on G^ξ.*

Proof. It is obvious that $x^\zeta y^\zeta$ is on a minimum spanning tree of G^ξ, since it was on the minimum spanning tree of G. Let u^ζ and v^ζ be the special vertices whose shortest path passes through $x^\zeta y^\zeta$, and let z^ξ be the vertex obtained by contracting $x^\xi y^\xi$.

Suppose that $b(G^\xi - x^\zeta y^\zeta, x^\zeta, y^\zeta) < b(G - x^\zeta y^\zeta, x^\zeta, y^\zeta)$. Clearly, the new minimum bottleneck x^ζ–y^ζ path P must pass through z^ξ, and every edge of $P - x^\xi y^\xi$ must have length strictly less than $b(G - x^\zeta y^\zeta, x^\zeta y^\zeta)$ but $c(G, x^\xi y^\xi) > b(G - x^\zeta y^\zeta, x^\zeta y^\zeta)$. As $x^\zeta y^\zeta$ lies on the (unique) minimum spanning tree for G, $c(G, x^\zeta y^\zeta) < b(G - x^\zeta y^\zeta, x^\zeta y^\zeta)$. Consider the x^ξ–y^ξ path $P - x^\xi y^\xi \cup x^\zeta y^\zeta$, which, from the foregoing, must have a bottleneck length of strictly less than $b(G - x^\zeta y^\zeta, x^\zeta y^\zeta)$. Hence $b(G - x^\xi y^\xi, x^\xi y^\xi) < c(G, x^\xi y^\xi)$, contradicting the assumption that $x^\xi y^\xi$ lies on a minimum spanning tree. Hence $b(G^\xi - x^\zeta y^\zeta, x^\zeta, y^\zeta) \geqslant d(G^\xi, u^\zeta, v^\zeta)$, since the former cannot decrease and the latter cannot increase.

Suppose that no shortest u^ζ–v^ζ path in G^ξ passes through $x^\zeta y^\zeta$. As only paths containing $x^\xi y^\xi$ are affected by ξ, any new shortest path must pass through z^ξ. Every edge ab on such a path has $c(G^\xi, ab) \leqslant d(G^\xi, u^\zeta, v^\zeta) < d(G, u^\zeta, v^\zeta)$, since, by assumption, a shortest u^ζ–v^ζ path in G^ξ is shorter than in G. This implies the existence of an x^ζ–y^ζ path in G^ξ with bottleneck length strictly less than $d(G, u^\zeta, v^\zeta)$, which contradicts the foregoing proof of non-decreasing bottleneck distances. Hence a shortest u^ζ–v^ζ path in G^ξ must pass through $x^\zeta y^\zeta$.

Hence all of conditions for the NSV reduction ζ are satisfied in G^ξ, as required. □

Corollary 6.6. *The set* {NSV} *is confluent over the set of all graphs with unique minimum spanning trees.*

Proof. First, it is easy to see the set of all graphs with unique minimum spanning trees is closed under NSV reduction, as the minimum spanning tree contracts with the reduction. The proof is then similar to that of Theorem 6.1. □

If we assume that the minimum spanning tree used by a reduction algorithm is contracted with an NSV reduction (rather than being re-computed from scratch), Corollary 6.6 can be used to show that, given a minimum spanning tree on an arbitrary initial graph, the irreducible graph that results from a sequence of NSV reductions is unique. That is to say, selection of the minimum spanning tree on the original graph is the only choice of any consequence for NSV reduction.

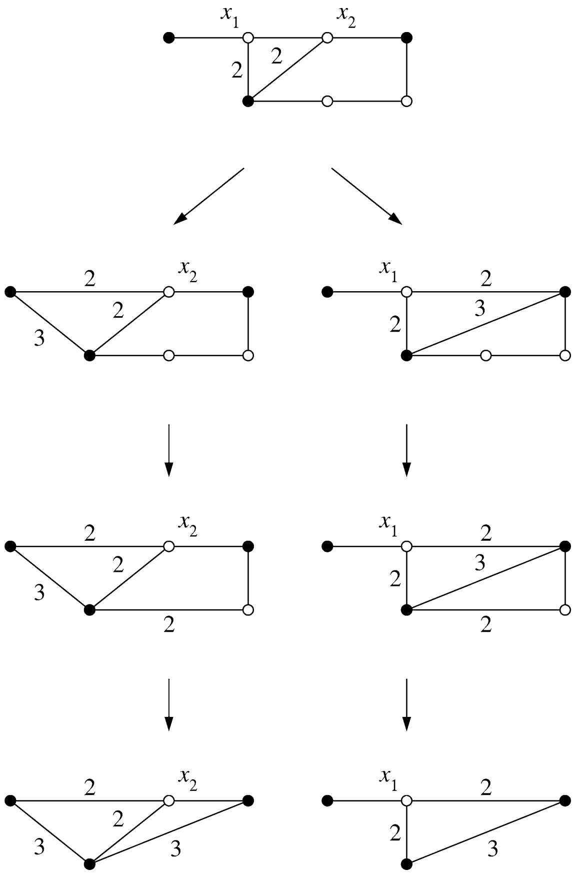

Fig. 4. Two non-isomorphic irreducible graphs produced by BDk reduction.

In an earlier version of this paper [8], it was conjectured that the set {BDk} is confluent. This is, however, not true, as Fig. 4 shows. As the reduced graphs are irreducible under RT, CRT and SD reduction, we immediately obtain the result that the set {RT, CRT, SD, BDk}, and its subsets containing BDk, are not confluent.

Observing that distances can only be increased, and the upper bound left unchanged, by a CRT reduction, it seems unlikely that a vertex subject to a CRT reduction could ever cease to be subject to CRT reduction by the action of other CRT reductions.

Conjecture 6.7. *The set* {CRT} *is confluent.*

At first glance, it may appear that Conjecture 6.7 is "obvious" by the same approach as used to prove Theorem 6.1. Unfortunately, since a CRT reduction may delete the minimum cost edge of a vertex u, altering the quantity $d(G,u,x) - \hat{c}(G,u)$ for a CRT vertex x, the vertices u_1 and u_2, and hence the conditions to be satisfied by Theorems 4.2 and 4.3, may be radically altered by the action of a CRT reduction.

Regardless of whether or not Conjecture 6.7 holds, it is easy to see from the proof of Theorems 4.2 and 4.3 that if a vertex or edge is subject to a CRT reduction, then there is always a minimum Steiner tree not containing this vertex for so long as only CRT reductions are performed on the graph. Hence, it is possible to "force confluence" of {CRT} by testing all edges and vertices in the graph and marking them all for deletion without checking the CRT conditions again.

7. Conclusion

This paper has considered reduction operations for the Steiner Problem in Graphs and shown that, for several sets of operations, all maximal reduction sequences are equivalent. For several other sets, examples have been given showing that different maximal reduction sequences can lead to different results. These observations have implications for obtaining the maximum possible amount of reduction for the Steiner Problem in Graphs.

We have made statements about every reduction test described by Duin and Volgenant, other than the CEC test. Nonetheless, our confluence results are incomplete and the new tests of [3] are yet to be examined. Little is known about obtaining maximum reduction with non-confluent sets of reductions. We hope to address these gaps in the near future.

References

[1] F. Baader, T. Nipkow, Term Rewriting and All That, Cambridge University Press, Cambridge, 1998.
[2] J.E. Beasley, An algorithm for the Steiner problem in graphs, Networks 14 (1984) 147–159.
[3] C.W. Duin, Preprocessing the Steiner problem in graphs, in: Advances in Steiner Trees, Kluwer, Dordrecht, 2000, pp. 175–233.
[4] C.W. Duin, A. Volgenant, Reduction tests for the Steiner problem in graphs, Networks 19 (1989) 549–567.
[5] F.K. Hwang, D.S. Richards, P. Winter, The Steiner Tree Problem, North-Holland, Amsterdam, 1992.
[6] A. Iwainsky, E. Canuto, O. Taraszow, A. Villa, Network decomposition for the optimization of connection structures, Networks 16 (1986) 205–235.
[7] R.M. Karp, Reducibility among combinatorial problems, in: Complexity of Computer Computations, Plenum, New York, 1972, pp. 83–103.
[8] J.H. Kingston, N.P. Sheppard, On reductions for the Steiner problem in graphs, in: Tenth Australasian Workshop on Combinatorial Algorithms Perth, WA, Curtin University Press, 1999, pp. 34–43.
[9] M.H.A. Newman, On theories with a combinatorial definition of 'equivalence', Ann. Math. 43 (1942) 223–243.