# Reduction Tests for the Prize-Collecting Steiner Problem

Eduardo Uchoa
Departamento de Engenharia de Produção, Universidade Federal Fluminense
Address: Rua Passso da Pátria 156 Bl.E sala 440, Niterói, Brazil, 24210-240.
Email: uchoa@producao.uff.br

January, 2005

**Abstract**

This article introduces a proper redefinition of the concept of bottleneck Steiner distance for the Prize-Collecting Steiner Problem. This allows the application of reduction tests known to be effective on Steiner Problem in Graphs in their full power. Computational experiments attest the effectiveness of the proposed tests.

**keywords** Steiner Problem, Network Design, Preprocessing, Combinatorial Optimization

## 1 Introduction

The *Prize-Collecting Steiner Problem* (PCSP) consists of given a connected graph $G = (V, E)$, positive edge costs $c(e)$ (also denoted by $c(u, v)$ when $e = (u, v)$), and non-negative vertex profits $p(i)$; find a subtree $(V', E')$ of $G$ maximizing $\sum_{e \in V'} p(i) - \sum_{e \in E'} c(e)$. Vertices with positive profits, those in the set $T = \{i \in V \mid p(i) > 0\}$, are called terminals. This problem is gaining much attention in the last years due to its practical applications on network design. The Steiner Problem in Graphs (SPG) can be viewed as the particular case of the PCSP where terminal profits are high enough to assure that all of them must be any optimal solution.

The practical efficiency of algorithms for the SPG increased dramatically in recent years. The sequence of results reported in [13, 15, 16, 19, 17, 14, 12] on the main benchmark instances from the literature [8] improve by two or three orders of magnitude upon the results from previous articles, like [7]. The strength of those new algorithms lies on a complex combination of reduction tests, primal heuristics, dual heuristics and branch-and-cut. Reduction tests are a key part of those combos.

The first reduction tests for the SPG were introduced by Beasley [2], Balakrishnan and Patel [1] and Voss [18]. They were generalized by Duin and Volgenant [5] in a work that established a set of reduction tests widely used in the 1990's decade. The most important such tests are based on the concept of bottleneck Steiner distances. More recently, Uchoa et al. [17] enhanced Duin and Volgenant's tests with the idea of expansion. This idea was further developed by Polzin and Vahdati [14]. The reduction tests with expansion still rely on Bottleneck Steiner distances.

Current algorithms for the PCSP are not as advanced as their SPG counterparts. This is statement is certainly true when one considers reduction tests. Recent algorithms like [3], [10], [9], and [11] apply weakened versions of Duin and Volgenant's tests obtained by replacing the bottleneck Steiner distances by standard distances. This weakening makes those tests much less effective.

This article proposes a new definition of bottleneck Steiner distances on the PCSP context. This redefinition allows the application of Duin and Volgenant's tests (and their enhancements with the idea of expansion) in their full power on the PCSP, just like they are now applied on the SPG. There is only one important difference. Computing exact bottleneck distances on the SPG can be done in polynomial time. On the other hand, it is shown that computing bottleneck distances on the PCSP is NP-hard. This point does not hinder the practical use of the new tests. Most current SPG codes only use fast and efficient heuristics to compute bottleneck distances. Similar heuristics also work on the PCSP, as shown by the computational experiments in the last section.

## 2 Reduction tests for the SPG

Reduction tests are procedures devised to transform an original instance into a smaller equivalent instance. An edge $e \in E$ is said to be *choosable* if there is at least one optimal solution containing $e$ and *redundant* if there is at least one optimal solution not containing $e$. Some reduction tests try to identify choosable and redundant edges. Once a choosable edge $e = (u, v)$ is identified, it can be forced into the solution and its endpoints $u$ and $v$ may be contracted. A redundant edge is simply deleted from the graph. In either case, the size of the instance is reduced. Other reduction tests lead to more complex graph transformations. Reduction tests may be successively applied to already reduced graphs, until no further reduction is possible. Some very simple tests are:

**Test 1 Non-terminal with degree 1 (NTD1)** - *A non-terminal vertex with degree 1 and its adjacent edge may be deleted.*

**Test 2 Non-terminal with degree 2 (NTD2)** - *A non-terminal vertex $u$ with degree 2 and its adjacent edges $(u, v)$ and $(u, w)$ may be replaced by a single edge $(v, w)$ with cost $c(u, v) + c(u, w)$.*

**Test 3 Terminal with degree 1 (TD1)** - *If $|T| \geq 2$, the edge adjacent to a terminal vertex with degree 1 is choosable.*

In order to introduce more complex tests, we review the definition of bottleneck Steiner distance on the SPG context. Let $u$ and $v$ be two distinct vertices in $V$. Let $\mathcal{P}(u, v)$ denote the set of all simple paths joining $u$ to $v$. The standard *distance* between vertices $u$ and $v$ is defined as

$$d(u, v) = \min\{c(P) \mid P \in \mathcal{P}(u, v)\}, \tag{1}$$

where $c(P)$ denotes the sum of the costs of the edges in path $P$. For $P \in \mathcal{P}(u, v)$, let $T(P)$ be $\{u, v\} \cup (T \cap P)$, i.e. the vertex-set formed by $u$, $v$ and the terminals in $P$. Two vertices $x$ and $y$ in $T(P)$ are said to be *consecutive* if the subpath from $x$ to $y$ in $P$ contains no other vertices in $T(P)$. The *Steiner distance* $SD(P)$ is the length of the longest subpath in $P$ joining two consecutive vertices in $T(P)$. The *bottleneck Steiner distance* between vertices $u$ and $v$ is defined as

$$B(u, v) = \min\{SD(P) \mid P \in \mathcal{P}(u, v)\}. \tag{2}$$

A nice interpretation for bottleneck distances (that appears, for instance in [4, 7]) is to consider $G$ as a road system, $T$ as petrol stations and a driver who wants to go from $u$ to $v$. Then $B(u, v)$ is the minimum distance he must be able to drive without refilling in order to reach his destination.

The bottleneck Steiner distance without passing through a given edge $e$ is defined as

$$B(u, v)^{-e} = \min\{SD(P) \mid P \in \mathcal{P}(u, v);\ e \notin P\}. \tag{3}$$

If $e$ disconnects $u$ and $v$, $B(u, v)^{-e} = \infty$.

The key concept of bottleneck distance allowed Duin and Volgenant [5] to propose strong reduction tests, generalizing some ideas which appeared earlier in the literature [2, 1, 18]. The most important such tests, in terms of graph reductions obtained, are the SD and the NTD-3 tests.

**Test 4 Special distance (SD)** - *Let $(u, v)$ be an edge in $E$. If $B(u, v)^{-(u,v)} \leq c(u, v)$, then edge $(u, v)$ is redundant.*

**Test 5 Non-terminal degree 3 (NTD-3)** - *Let $u$ be a non-terminal vertex with degree 3, adjacent to vertices $v$, $w$, and $z$. If*

$$\min\{B(v, w) + B(v, z), B(w, v) + B(w, z), B(z, v) + B(z, w)\} \leq c(u, v) + c(u, w) + c(u, z),$$

*then there is an optimal solution where the degree of $u$ is at most 2. Therefore, $u$ and its three adjacent edges can be replaced by the following three edges: $(v, w)$ with cost $c(u, v) + c(u, w)$, $(v, z)$ with cost $c(u, v) + c(u, z)$, and $(w, z)$ with cost $c(u, w) + c(u, z)$.*

The graph transformation given by the NTD-3 test is not much advantageous by itself, since the number of edges remains the same. But some of the newly created edges are quite likely to be immediately eliminated by the SD test. Duin and Volgenant actually introduced a general test NTD-$k$ for non-terminals with any degree $k$, based on bottleneck distances. They also proposed another test that does not use bottleneck distances.

**Test 6 Terminal distance (TDist)** - *Let $[W, \bar{W}]$ be a partition of the vertices in $V$ such that the subgraphs induced by $W$ and $\bar{W}$ are both connected, with $W \cap T \neq \emptyset$ and $\bar{W} \cap T \neq \emptyset$. Let $\delta(W)$ be the cut induced by this partition. If $\delta(W) = \{e\}$, then $e$ is choosable. If $|\delta(W)| \geq 2$, let $e = \text{argmin}_{e' \in \delta(W)} c(e')$ and $f = \text{argmin}_{f' \in \delta(W) \setminus \{e\}} c(f')$ be respectively a shortest and a second shortest edge in the cut. Suppose $e = (u, v)$ with $u \in W$ and $v \in \bar{W}$. If*

$$\min\{d(u, t_1) \,|\, t_1 \in T \cap W\} + c(e) + \min\{d(v, t_2) \,|\, t_2 \in T \cap \bar{W}\} \;\leq\; c(f),$$

*then $e$ is choosable.*

The new tests introduced in Uchoa et al. [17] are enhancements of the SD and the NTD-$k$ tests with the idea of expansion. Loosely speaking, this idea means probing the instance to dynamically build a chain of logical implications of the kind "if edge $e$ appear in some optimal solution $R$ then edge $f$ must also be in $R$" in order to prove that some graph reduction can be indeed performed. Such probing is heavily based on the concept of bottleneck distances.

# 3 Reduction tests for the PCSP

Consider a PCSP instance. Again, let $u$ and $v$ be two distinct vertices in $V$ and let $\mathcal{P}(u, v)$ denote the set of all simple paths joining $u$ to $v$. Consider a path $P \in \mathcal{P}(u, v)$. Define $P(x, y)$ as the subpath of $P$ between two given vertices $x$ and $y$ in $P$. Define the *Steiner distance* associated to this subpath as

$$SD(P(x, y)) = \sum_{e \in P(x,y)} c(e) - \sum_{i \in P(x,y) \setminus \{x,y\}} p(i),$$

i.e. the sum of the edge costs in this subpath, minus the sum of the profits of vertices in the interior of this subpath. The *Steiner distance* associated to the whole path $P$ is:

$$SD(P) = max_{x,y \in P} SD(P(x, y))$$

Finally, the *bottleneck Steiner distance* between vertices $u$ and $v$ is defined as

$$B(u, v) = \min\{SD(P) \,|\, P \in \mathcal{P}(u, v)\}. \tag{4}$$

The new definition of bottleneck distances for the PCSP can be interpreted as follows. Consider $G$ as a road system, $c(i,j)$ as the number of units of fuel needed to drive from $i$ to $j$ and the vertices as petrol stations that have only $p(i)$ units of fuel available. Suppose that a driver wants to go from $u$ to $v$, starting with a full tank. Then $B(u,v)$ is the minimum tank capacity (in units of fuel) necessary to reach $v$. In the SPG case, terminals can be viewed as petrol stations an unlimited quantity of fuel in stock, so the vehicle tank can be always completely refilled.

Figure 1 depicts part of a PCSP instance, where edge costs are indicated and vertex profits are the following, $p(a)=0, p(b)=5, p(c)=4, p(d)=0, p(e)=15, p(f)=3$. The Steiner distance associated to path $\{u,a,b,c,d,e,f,v\}$ is 10, the same of the subpath $\{b,c,d,e\}$. Going back to the vehicle analogy, we can say that a tank capacity of 10 units is enough to travel from $u$ to $v$ by that path. The vehicle starts at $u$ with a full tank, reaches $b$ with 5 units left, completes the tank with more 5 units, reaches $c$ with 2 units left, puts more 4 units, reaches $e$ with the tank empty, fill it with 10 units (from the 15 available), reaches $f$ with 5 units left, puts more 3 units and finally arrives at $v$ with 2 units left.

**Theorem 1** *Test SD with the new definition of bottleneck Steiner distance is valid for the PCSP.*

**Proof:** Suppose that $B(u,v)^{-(u,v)} \leq c(u,v)$. Let $P \in \mathcal{P}(u,v)$ be a path not using edge $(u,v)$ such that $SD(P) = B(u,v)$. Let $R$ be a solution tree using edge $(u,v)$. Removing $(u,v)$ from $R$ creates two subtrees. Let $R_u$ be the one containing vertex $u$ and $R_v$ be the other, containing $v$. Pick two vertices $x$ and $y$ from $P$ such that $x \in R_u$, $y \in R_v$ but no other vertices in $P(x,y)$ are in $R$. Since $SD(P(x,y)) \leq B(u,v)^{-(u,v)} \leq c(u,v)$, then $R' = R_u \cup R_v \cup P(x,y)$ is a solution tree at least as good as $R$. Therefore, there is an optimal solution that does not use $(u,v)$. ∎

Tests NTD-1 and NTD-2 are clearly valid for the PCSP.

**Theorem 2** *Test NTD-3 with the new definition of bottleneck Steiner distance is valid for the PCSP.*

**Proof:** Without loss of generality, suppose that $B(v,w) + B(v,z) \leq c(u,v) + c(u,w) + c(u,z)$, since the other two cases are similar. Let $P_1 \in \mathcal{P}(v,w)$ be a path such that $SD(P_1) = B(v,w)$ and $P_2 \in \mathcal{P}(v,z)$ be a path such that $SD(P_2) = B(v,z)$. Let $R$ be a solution tree using vertex $u$ with degree 3. Removing $u$ and its adjacent edges from $R$, we obtain the subtrees $R_v$, $R_w$, and $R_z$. Pick two vertices $x_1$ and $y_1$ from $P_1$ such that $x_1 \in R_w$, $y_1 \in R_v$ but no other vertices in the subpath $P_1(x_1,y_1)$ are in $R$. Similarly, pick two vertices $x_2$ and $y_2$ from $P_2$ such that $x_2 \in R_z$, $y_2 \in R_v$ but no other vertices in the subpath $P_2(x_2,y_2)$ are in $R$. Since $SD(P_1(x_1,y_1)) \leq B(v,w)$ and $SD(P_2(x_2,y_2)) \leq B(v,z)$, $SD(P_1(x_1,y_1)) + SD(P_2(x_2,y_2)) \leq c(u,v) + c(u,w) + c(u,z)$. Therefore $R' = R_v \cup R_w \cup R_z \cup P_1(x_1,y_1) \cup P_2(x_2,y_2)$ is a solution tree at least as good as $R$. Vertex $u$ has degree at most 2 in $R'$ (otherwise $B(v,w) + B(v,z) > c(u,v) + c(u,w) + c(u,z)$). ∎

The applicability of tests NTD-$k$ can be increased by considering as non-terminals not only vertices with zero profit. One can consider a vertex $u$ with positive profit as a "non-terminal" if it can be shown that $u$ would never appear as a leaf in some optimal solution. For instance, if $p(u)$ is less or equal to the cost of the cheapest edge adjacent to $u$. The graph transformations produced by the tests must be slightly changed when $p(u) > 0$. On test NTD-2, $(u,v)$ and $(u,w)$ must be replaced by an edge $(v,w)$ with cost $c(u,v) + c(u,w) - p(u)$. A similar change must be done on NTD-3. The TD-1 test is applied as follows: a terminal $u$ of degree 1 and its adjacent edge $(u,v)$ can be removed; if $p(u) - c(u,v) > 0$, this difference should be added to $p(v)$.

All the tests with expansion can also be applied on the PCSP with the new definitions of bottleneck distances and "non-terminals". We do not prove this here, but it is quite easy to adapt the proofs found in [17] to this new context.

The only SPG test that is not easily adapted to the PCSP is TDist. Its direct application depends on showing that both vertices $t_1$ and $t_2$ (as in the test definition) belong to some optimal solution. This can be hard, except on instances where some terminals have very large profits with respect to edge costs. However, the following weakening of the TDist test is valid:

**Test 7 Minimum Adjacency (MA)** - *Let $u$ and $v$ be two adjacent terminal vertices. If $\min\{p(u), p(v)\} - c(u, v) \leq 0$ and $c(u, v) = \min_{(u,t)\in E} c(u, t)$ , then $u$ and $v$ can be merged into one vertex of profit $p(u) + p(v) - c(u, v)$.*

The MA test was already used in [9]. This test does not need to assume that $u$ and $v$ belong to some optimal solution. The reasoning here is: if $u$ or $v$ belong to some optimal solution then $(u, v)$ also belong to some optimal solution. This may lead to other PCSP tests. For instance, if $\min\{p(u), p(v)\} - c(u, v) \leq 0$ and $(u, v)$ is a cut-edge of the graph, then $u$ and $v$ can be merged into a single vertex.

# 4 Computing Bottleneck Steiner Distances

A table with the exact bottleneck Steiner distances for all pairs of vertices in a SPG instance can be computed in $O(|V|^3)$ time [4]. Computing exact bottleneck Steiner distances on a PCSP instance can be much harder. Define this problem in a more formal way.

**Prize-Collecting Bottleneck Distance**
**Instance:** Graph $G = (V, E)$, positive integers $c$ associated to the edges, non-negative integers $p$ associated to vertices, vertices $u, v$ in $V$ and integer $b$.
**Question:** Is $B(u, v) \leq b$ ?

**Theorem 3** *Prize-Collecting Bottleneck Distance is NP-hard.*

**Proof:** The Hamiltonian path problem, find a simple path visiting all vertices in a graph, is widely known to be NP-hard. The following version of the problem, find a simple path between two given vertices visiting all the other vertices in a graph, is easily shown to be NP-hard too. This version is formally defined as follows.

**Hamiltonian Path**
**Instance:** Graph $G' = (V', E')$, vertices $u', v'$ in $V$.
**Question:** Is there a simple path from $u'$ to $v'$ visiting all the other vertices in $G'$ ?

Given an instance of Hamiltonian Path, produce an instance of Prize-Collecting Bottleneck Distance as follows. Graph $G = (V' \cup \{u, v\}, E' \cup \{(u, u'), (v, v')\})$. Costs are 1 for the edges in $E'$, $c(u, u') = c(v, v') = |V'|$. The profit of all vertices are 2, except for $u'$, $p(u') = 1$. Define $b$ as equal to $|V'|$. It can be seen that $B(u, v) = b$ on that instance if and only if there is a hamiltonian path from $u'$ to $v'$ on the original instance. ∎

The above theorem rules out the computation of exact bottleneck distances on reduction tests for the PCSP. One must use heuristics instead. Such kind of heuristics are widely used when applying reduction tests on SPG, since the $O(n^3)$ time for an exact computation is considered excessive. Those heuristics are fast and very effective, they yield upper bounds on the true bottleneck distances so tight that the amount of graph reduction obtained by the tests barely changes. This is possible because the tests (even those with expansion) almost always ask for distances between vertices that are very close in the graph. Only a few terminals in that neighborhood are likely to be relevant in that computation.

## 5 Computational Experiments

In order to evaluate the practical performance of the new tests, a preprocessing package containing tests NTD-1, NTD-2, TD-1, MA, SD (with expansion) and NTD-3 (with expansion) was implemented. The bottleneck distances $B(u, v)$ were heuristically computed by only considered paths in $\mathcal{P}(u, v)$ containing at most two terminal vertices. Two types of instances were used: the P instances proposed by Johnson et al. [6], and the C and D instances proposed by Canuto et al. [3]. Those instances appear on most recent literature on PCSP, including [9, 10, 11]. On all those works, the reductions obtained by applying the weakened versions of Duin and Volgenant's tests were similar, Tables 1–3 compare the proposed preprocessing with the one from Ljubić et al.[9]. Rows **Avg. Ratio** give the average ratio between the sizes of the reduced and original instances (in terms of vertices and edges).

The improvements obtained with the new tests are very significant. Some instances can even be solved by preprocessing alone. In those cases, the final reduced graph contains a single vertex with profit equal to the optimal solution value. Large instances with many terminals, like C20-A, C20-B, D20-A and D20-B, could be quickly solved in this way.

Summarizing, the results obtained on benchmark instances from the literature are quite satisfactory. As expected, on instances with more terminals, bottleneck Steiner distances are likely to be significantly smaller than standard distances, leading to larger reductions. It is worthy to mention that the practical applications mentioned in Canuto et al. [3], Lucena and Resende [10] (telecommunications network design) and in Ljubić et al. [9] (gas distribution) provide instances where almost all vertices have positive profits.

## References


[1] A. Balakrishnan and N. Patel, "Problem reduction methods and a tree generation algorithm for the Steiner network problem", *Networks* 17, 65–85, 1987.

[2] J. Beasley, "An algorithm for the Steiner problem in graphs", *Networks* 14, 147–159, 1984.

[3] S. Canuto, M. Resende and C. C. Ribeiro, "Local search with perturbations for the prize-collecting Steiner tree problem in graphs", *Networks* 38, 50–58, 2001.

[4] C. Duin, "Steiner's problem in graphs", PhD Thesis, University of Amsterdam, 1993.

[5] C. Duin and A. Volgenant, "Reduction tests for the Steiner problem in graphs", *Networks* 19, 549–567, 1989.

[6] D. Johnson, M. Minkoff and S. Phillips. "The prize-collecting Steiner tree problem: Theory and practice", Proceedings of 11th ACM-SIAM Symposium on Discrete Algorithms, 760-769, San Francisco, 2000.

[7] T. Koch and A. Martin, "Solving Steiner tree problems in graphs to optimality", *Networks* 32, 207–232, 1998.

[8] T. Koch, A. Martin and S. Voss, "SteinLib: An updated library on Steiner tree problems in graphs", Konrad-Zuse-Zentrum für Informationstechnik Berlin, ZIB-Report 00-37, 2000 (online document at `http://elib.zib.de/steinlib`).

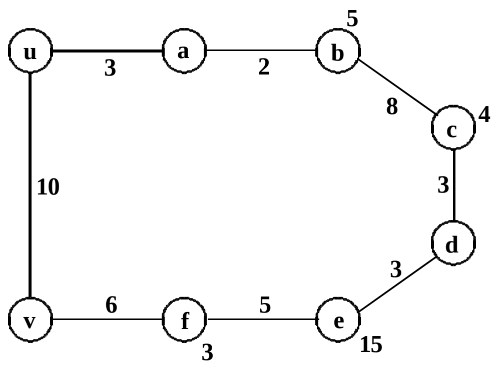


Figure 1: Part of a PCSP instance. Edge $(u, v)$ can be eliminated by the SD test.

| | Original | | | Prep [9] | | New Prep | |
|---|---|---|---|---|---|---|---|
| Instance | Nodes | Edges | Terminals | Nodes | Edges | Nodes | Edges |
| P100 | 100 | 317 | 33 | 66 | 163 | 13 | 16 |
| P100.1 | 100 | 284 | 32 | 84 | 196 | 1 | 0 |
| P100.2 | 100 | 297 | 26 | 75 | 187 | 7 | 9 |
| P100.3 | 100 | 316 | 24 | 91 | 237 | 8 | 10 |
| P100.4 | 100 | 284 | 32 | 69 | 186 | 7 | 9 |
| P200 | 200 | 587 | 48 | 166 | 438 | 34 | 50 |
| P400 | 400 | 1200 | 94 | 345 | 1002 | 142 | 262 |
| P400.1 | 400 | 1212 | 120 | 323 | 983 | 137 | 252 |
| P400.2 | 400 | 1196 | 107 | 341 | 997 | 72 | 114 |
| P400.3 | 400 | 1175 | 113 | 334 | 969 | 121 | 216 |
| P400.4 | 400 | 1144 | 94 | 344 | 949 | 165 | 298 |
| **Avg. Ratio** | | | | **80.9%** | **73.8%** | **19.3%** | **10.9%** |

Table 1: Reductions on series P instances.

| | Original | | | Prep [9] | | New Prep | |
|---|---|---|---|---|---|---|---|
| Instance | Nodes | Edges | Terminals | Nodes | Edges | Nodes | Edges |
| C1-A | 500 | 625 | 5 | 116 | 214 | 105 | 190 |
| C1-B | 500 | 625 | 5 | 125 | 226 | 49 | 77 |
| C2-A | 500 | 625 | 10 | 109 | 207 | 82 | 148 |
| C2-B | 500 | 625 | 10 | 111 | 209 | 71 | 125 |
| C3-A | 500 | 625 | 83 | 160 | 277 | 113 | 190 |
| C3-B | 500 | 625 | 83 | 185 | 304 | 79 | 121 |
| C4-A | 500 | 625 | 125 | 178 | 300 | 72 | 119 |
| C4-B | 500 | 625 | 125 | 218 | 341 | 71 | 113 |
| C5-A | 500 | 625 | 250 | 163 | 274 | 7 | 9 |
| C5-B | 500 | 625 | 250 | 199 | 314 | 1 | 0 |
| C6-A | 500 | 1000 | 5 | 355 | 822 | 346 | 792 |
| C6-B | 500 | 1000 | 5 | 356 | 823 | 344 | 778 |
| C7-A | 500 | 1000 | 10 | 365 | 842 | 353 | 806 |
| C7-B | 500 | 1000 | 10 | 365 | 842 | 342 | 769 |
| C8-A | 500 | 1000 | 83 | 367 | 849 | 251 | 531 |
| C8-B | 500 | 1000 | 83 | 369 | 850 | 217 | 410 |
| C9-A | 500 | 1000 | 125 | 387 | 877 | 279 | 577 |
| C9-B | 500 | 1000 | 125 | 389 | 879 | 232 | 440 |
| C10-A | 500 | 1000 | 250 | 359 | 841 | 103 | 166 |
| C10-B | 500 | 1000 | 250 | 323 | 798 | 100 | 156 |
| C11-A | 500 | 2500 | 5 | 489 | 2143 | 485 | 1801 |
| C11-B | 500 | 2500 | 5 | 489 | 2143 | 480 | 1667 |
| C12-A | 500 | 2500 | 10 | 484 | 2186 | 453 | 1495 |
| C12-B | 500 | 2500 | 10 | 484 | 2186 | 441 | 1358 |
| C13-A | 500 | 2500 | 83 | 472 | 2113 | 343 | 799 |
| C13-B | 500 | 2500 | 83 | 471 | 2112 | 317 | 704 |
| C14-A | 500 | 2500 | 125 | 466 | 2081 | 190 | 365 |
| C14-B | 500 | 2500 | 125 | 459 | 2048 | 179 | 330 |
| C15-A | 500 | 2500 | 250 | 406 | 1871 | 1 | 0 |
| C15-B | 500 | 2500 | 250 | 370 | 1753 | 1 | 0 |
| C16-A | 500 | 12500 | 5 | 500 | 4740 | 499 | 2714 |
| C16-B | 500 | 12500 | 5 | 500 | 4740 | 499 | 2714 |
| C17-A | 500 | 12500 | 10 | 498 | 4694 | 494 | 2295 |
| C17-B | 500 | 12500 | 10 | 498 | 4694 | 494 | 2295 |
| C18-A | 500 | 12500 | 83 | 469 | 4569 | 374 | 1002 |
| C18-B | 500 | 12500 | 83 | 465 | 4538 | 374 | 997 |
| C19-A | 500 | 12500 | 125 | 430 | 3982 | 246 | 589 |
| C19-B | 500 | 12500 | 125 | 416 | 3867 | 249 | 592 |
| C20-A | 500 | 12500 | 250 | 241 | 1222 | 1 | 0 |
| C20-B | 500 | 12500 | 250 | 133 | 563 | 1 | 0 |
| **Avg. Ratio** | | | | **69.7%** | **59.9%** | **46.7%** | **29.1%** |

Table 2: Reductions on series C instances.

| | Original | | | Prep [9] | | New Prep | |
|---|---|---|---|---|---|---|---|
| Instance | Nodes | Edges | Terminals | Nodes | Edges | Nodes | Edges |
| D1-A | 1000 | 1250 | 5 | 231 | 440 | 223 | 422 |
| D1-B | 1000 | 1250 | 5 | 233 | 443 | 223 | 416 |
| D2-A | 1000 | 1250 | 10 | 257 | 481 | 238 | 450 |
| D2-B | 1000 | 1250 | 10 | 264 | 488 | 232 | 423 |
| D3-A | 1000 | 1250 | 167 | 301 | 529 | 114 | 194 |
| D3-B | 1000 | 1250 | 167 | 372 | 606 | 129 | 202 |
| D4-A | 1000 | 1250 | 250 | 311 | 541 | 171 | 297 |
| D4-B | 1000 | 1250 | 250 | 387 | 621 | 50 | 70 |
| D5-A | 1000 | 1250 | 500 | 348 | 588 | 84 | 125 |
| D5-B | 1000 | 1250 | 500 | 411 | 649 | 12 | 17 |
| D6-A | 1000 | 2000 | 5 | 740 | 1707 | 740 | 1697 |
| D6-B | 1000 | 2000 | 5 | 741 | 1708 | 736 | 1682 |
| D7-A | 1000 | 2000 | 10 | 734 | 1705 | 721 | 1664 |
| D7-B | 1000 | 2000 | 10 | 736 | 1707 | 702 | 1602 |
| D8-A | 1000 | 2000 | 167 | 764 | 1738 | 673 | 1489 |
| D8-B | 1000 | 2000 | 167 | 778 | 1757 | 561 | 1142 |
| D9-A | 1000 | 2000 | 250 | 752 | 1716 | 580 | 1260 |
| D9-B | 1000 | 2000 | 250 | 761 | 1724 | 439 | 841 |
| D10-A | 1000 | 2000 | 500 | 694 | 1661 | 235 | 425 |
| D10-B | 1000 | 2000 | 500 | 629 | 1586 | 36 | 56 |
| D11-A | 1000 | 5000 | 5 | 986 | 4658 | 977 | 3971 |
| D11-B | 1000 | 5000 | 5 | 986 | 4658 | 972 | 3740 |
| D12-A | 1000 | 5000 | 10 | 991 | 4639 | 960 | 3300 |
| D12-B | 1000 | 5000 | 10 | 991 | 4639 | 942 | 3040 |
| D13-A | 1000 | 5000 | 167 | 966 | 4572 | 708 | 1713 |
| D13-B | 1000 | 5000 | 167 | 961 | 4566 | 694 | 1631 |
| D14-A | 1000 | 5000 | 250 | 946 | 4500 | 571 | 1238 |
| D14-B | 1000 | 5000 | 250 | 931 | 4469 | 512 | 1062 |
| D15-A | 1000 | 5000 | 500 | 832 | 4175 | 139 | 217 |
| D15-B | 1000 | 5000 | 500 | 747 | 3896 | 4 | 5 |
| D16-A | 1000 | 25000 | 5 | 1000 | 10595 | 1000 | 6735 |
| D16-B | 1000 | 25000 | 5 | 1000 | 10595 | 1000 | 6725 |
| D17-A | 1000 | 25000 | 10 | 999 | 10534 | 999 | 6330 |
| D17-B | 1000 | 25000 | 10 | 999 | 10534 | 999 | 6330 |
| D18-A | 1000 | 25000 | 167 | 944 | 9949 | 812 | 2314 |
| D18-B | 1000 | 25000 | 167 | 929 | 9816 | 806 | 2276 |
| D19-A | 1000 | 25000 | 250 | 897 | 9532 | 686 | 1895 |
| D19-B | 1000 | 25000 | 250 | 862 | 9131 | 680 | 1870 |
| D20-A | 1000 | 25000 | 500 | 488 | 2511 | 1 | 0 |
| D20-B | 1000 | 25000 | 500 | 307 | 1383 | 1 | 0 |
| **Avg. Ratio** | | | | **70.5%** | **62.8%** | **50.9%** | **33.4%** |

Table 3: Reductions on series D instances.

[9] I. Ljubić, R. Weiskircher, U. Pferschy, G. Klau, P. Mutzel and M. Fischetti, "Solving the prize-collecting Steiner tree problem to optimality", TR-186-1-04-01, Technische Universität Wien, October 2004. (Submitted to *Mathematical Programing*).

[10] A. Lucena and M. Resende, "Strong lower bounds for the prize collecting Steiner problem in graphs", *Discrete Applied Mathematics* 141, 277–294, 2004.

[11] G. Klau, I. Ljubić, A. Moser, P. Mutzel, P. Neuner, U. Pferschy and R. Weiskircher, "Combining a memetic algorithm with integer programming to solve the prize-collecting Steiner tree problem", Proceedings of the GECCO-2004, *Lecture notes in Computer Science* 3102, 1304–1315, 2004.

[12] T. Polzin, "Algorithms for the Steiner Problem in Networks", PhD Thesis, Universität des Saarlandes, 2003.

[13] T. Polzin and S. Vahdati, "Improved algorithms for the Steiner problem in networks", *Discrete Applied Mathematics* 112, 263–300, 2001.

[14] T. Polzin and S. Vahdati, " Extending Reduction Techniques for the Steiner Tree Problem", Proceedings of the ESA 2002, 795–807, 2002.

[15] M. Poggi de Aragão, E. Uchoa and R.F. Werneck, "Dual heuristics on the exact solution of large Steiner problems", *Electronic Notes in Discrete Mathematics* 7, 2001.

[16] E. Uchoa, "Algoritmos para problemas de Steiner com aplicação em projeto de circuitos VLSI", PhD Thesis, Pontifícia Universidade Católica do Rio de Janeiro, 2001.

[17] E. Uchoa, M. Poggi de Aragão and C.C. Ribeiro, "Preprocessing Steiner problems from VLSI layout", *Networks* 40, 38–50, 2002.

[18] S. Voss, "A reduction based algorithm for the Steiner problem in graphs", *Methods of Operations Research* 58, 239–252, 1989.

[19] R.F. Werneck, "Problema de Steiner em Grafos: Algoritmos Primais, Duais e Exatos", Master's Thesis, Pontifícia Universidade Católica do Rio de Janeiro, 2001.