2-Matchings, the Traveling Salesman Problem, and the Subtour LP: A Proof of the Boyd-Carr Conjecture

Author(s): Frans Schalekamp, David P. Williamson and Anke van Zuylen

2-Matchings, the Traveling Salesman Problem, and the Subtour LP: A Proof of the Boyd-Carr Conjecture

Frans Schalekamp
Department of Mathematics, College of William and Mary, Williamsburg, Virginia 23185,
frans@wm.edu, fransschalekamp.com

David P. Williamson
School of Operations Research and Information Engineering, Cornell University, Ithaca, New York 14853,
dpw@cs.cornell.edu, www.davidpwilliamson.net/work

Anke van Zuylen
Department of Mathematics, College of William and Mary, Williamsburg, Virginia 23185, anke@wm.edu, ankevanzuylen.com

Determining the precise integrality gap for the subtour linear programming (LP) relaxation of the traveling salesman problem is a significant open question, with little progress made in thirty years in the general case of symmetric costs that obey triangle inequality. Boyd and Carr [Boyd S, Carr R (2011) Finding low cost TSP and 2-matching solutions using certain half-integer subtour vertices. *Discrete Optim.* 8:525–539. Prior version accessed June 27, 2011, http://www.site.uottawa.ca/~sylvia/recentpapers/halftri.pdf.] observe that we do not even know the worst-case upper bound on the ratio of the optimal 2-matching to the subtour LP; they conjecture the ratio is at most 10/9.

In this paper, we prove the Boyd-Carr conjecture. In the case that the support of a fractional 2-matching has no cut edge, we can further prove that an optimal 2-matching has cost at most 10/9 times the cost of the fractional 2-matching.

Keywords: traveling salesman problem; subtour elimination; linear programming; integrality gap; 2-matching; fractional 2-matching

MSC2000 subject classification: Primary: 90C05 linear programming, 90C27 combinatorial optimization, 05C70 factorization, matching, covering and packing

OR/MS subject classification: Primary: networks/graphs: matchings, traveling salesman; secondary: programming: integer, relaxation

History: Received June 7, 2012; revised December 29, 2012, April 2, 2013, and April 7, 2013. Published online in *Articles in Advance* August 1, 2013.

1. Introduction. The traveling salesman problem (TSP) is the most famous problem in discrete optimization. Given a set of n cities and the costs $c(i, j)$ of traveling from city i to city j for all i, j, the goal of the problem is to find the least expensive tour that visits each city exactly once and returns to its starting point. An instance of the TSP is called *symmetric* if $c(i, j) = c(j, i)$ for all i, j; it is *asymmetric* otherwise. Costs obey the *triangle inequality* if $c(i, j) \leq c(i, k) + c(k, j)$ for all i, j, k. The TSP is known to be NP-hard, even in the case that instances are symmetric and obey the triangle inequality. From now on we consider only these instances unless otherwise stated.

Because of the NP-hardness of the traveling salesman problem, researchers have considered approximation algorithms for the problem. An α-approximation algorithm for the traveling salesman problem is a polynomial-time algorithm that returns a tour of cost at most a factor of α times the cost of an optimal tour. The best approximation algorithm currently known is a $\frac{3}{2}$-approximation algorithm given by Christofides in 1976 (Christofides [9]). Better approximation algorithms are known for special cases. Exciting progress has been made recently in the case of graph-TSP, in which costs $c(i, j)$ are given by shortest path distances in an unweighted graph; we will discuss these results shortly. However, to date, Christofides' algorithm has the best known performance guarantee for the general case.

There is a well-known, natural direction for making progress that has also defied improvement for nearly 30 years. The following linear programming (LP) relaxation of the traveling salesman problem was used by Dantzig et al. [10] in 1954. For simplicity of notation, we let $G = (V, E)$ be a complete undirected graph on n nodes. We use (i, j) to denote the (undirected) edge between i and j. In the LP relaxation, we have a variable $x(e)$ for all $e = (i, j)$ that denotes whether we travel directly between cities i and j on our tour. Let $c(e) = c(i, j)$ and let $\delta(S)$ denote the set of all edges with exactly one endpoint in $S \subseteq V$. Then the relaxation is

$$\min \ \sum_{e \in E} c(e) x(e)$$

$$(SUBT) \quad \text{subject to:} \quad \sum_{e \in \delta(i)} x(e) = 2, \quad \forall i \in V, \tag{1}$$

403

$$\sum_{e \in \delta(S)} x(e) \geq 2, \quad \forall S \subset V, \quad 3 \leq |S| \leq |V| - 3 \tag{2}$$

$$0 \leq x(e) \leq 1, \quad \forall e \in E. \tag{3}$$

Constraints (1) are called the *degree constraints*. Constraints (2) are sometimes called *subtour elimination constraints* or sometimes just *subtour constraints* since they prevent solutions in which there is a subtour of just the nodes in S. As a result, the linear program is sometimes called the *subtour LP*. It is known that the equality sign in the first set of constraints may be replaced by $\geq$ in the case in which the costs obey the triangle inequality (Goemans and Bertsimas [15]; see also Williamson [27]).

The LP is known to give excellent lower bounds on TSP instances in practice, coming within a percentage or two of the length of the optimal tour (see, for instance, Johnson and McGeoch [16]). However, its theoretical worst case is not well understood. In 1980, Wolsey [28] showed that Christofides' algorithm produces a solution whose value is at most $\frac{3}{2}$ times the value of the subtour LP (also shown later by Shmoys and Williamson [26]). This proves that the *integrality gap* of the subtour LP is at most $\frac{3}{2}$; the integrality gap is the worst-case ratio, taken over all instances of the problem, of the value of the optimal tour to the value of the subtour LP or the ratio of the optimal integer solution to the optimal fractional solution. The integrality gap of the LP is known to be at least $\frac{4}{3}$ via a specific class of instances. However, no instance is known that has integrality gap worse than this, and it has been conjectured for some time that the integrality gap is at most $\frac{4}{3}$ (see, for instance, Goemans [14]).

Stronger bounds on the integrality gap are known in the case of graph-TSP. If the graph is cubic and 3-connected, Gamarnik et al. [13] show an approximation algorithm with guarantee slightly better than $\frac{3}{2}$, and their result also implies an upper bound on the integrality gap of less than $\frac{3}{2}$. Aggarwal et al. [1] improve the bound to $\frac{4}{3}$ for these instances. Boyd et al. [8] and Mömke and Svensson [17] extend the bound of $\frac{4}{3}$ to arbitrary cubic graphs and to subcubic graphs, respectively. Oveis Gharan et al. [21] show that without any assumptions on the underlying graph, graph-TSP can be approximated to within $\frac{3}{2} - \epsilon$ for a small constant $\epsilon > 0$, and this implies a bound on the integrality gap of $\frac{3}{2} - \epsilon$ for such instances as well. Mömke and Svensson [17] show that the integrality gap is at most 1.461, and Mucha [19] improved this result to $\frac{13}{9}$. Sebő and Vygen [25] combine the ideas of Mömke and Svensson [17] with an algorithm based on a carefully chosen ear decomposition of the graph to show that the integrality gap for graph-TSP instances is at most $\frac{7}{5}$.

There is some evidence that the conjectured gap of $\frac{4}{3}$ might be true. Benoit and Boyd [4] have shown via computational methods that the conjecture holds for $n \leq 10$, and Boyd and Elliot-Magwood [7] have extended this to $n \leq 12$. In a 1995 paper, Goemans [14] showed that adding any class of valid inequalities known at the time to the subtour LP could increase the value of the LP by at most $\frac{4}{3}$; this is necessary for the conjecture to be true. Somewhat weaker evidence is as follows. A *2-matching* is an integer solution to the subtour LP obeying only the degree constraints (1) and the bounds constraints (3).[1] A *fractional 2-matching* is a 2-matching without the integrality constraints. Boyd and Carr [5] have shown that the integrality gap for the 2-matching problem is $\frac{4}{3}$. Furthermore, Boyd and Carr [6] have shown that if the subtour LP solution is half-integral (that is, $x(i, j) \in \{0, \frac{1}{2}, 1\}$ for all $i, j \in V$) and has a particular structure, then there is a tour of cost at most $\frac{4}{3}$ times the value of the subtour LP.

Not only do we not know the integrality gap of the subtour LP, Boyd and Carr have observed that we don't even know the worst-case ratio of the optimal 2-matching to the value of the subtour LP, which is surprising because 2-matchings are well understood and well characterized. They make the following conjecture.

CONJECTURE 1 (BOYD AND CARR [6]).   *The worst-case ratio of an optimal 2-matching to an optimal solution to the subtour LP is* $\frac{10}{9}$.

We remark that both the optimal subtour LP solution and the optimal 2-matching solution give lower bounds on the cost of the optimal TSP tour. The cost of an optimal 2-matching may be less than the optimal value of the subtour LP, since the 2-matching solution does not need to be connected. However, it is known that there are cases for which the cost of an optimal 2-matching is at least $\frac{10}{9}$ times the optimal solution to the subtour LP; see Figure 1. Boyd and Carr have shown that the conjecture is true if the solution to the subtour LP has a very special structure: namely, all variables $x(e) \in \{0, \frac{1}{2}, 1\}$, the cycles formed by the edges e with $x(e) = \frac{1}{2}$ all have the same odd size k, and the support is $(k-1)$-edge-connected.[2] In the general case, the only bound on this ratio that we know of is the Boyd and Carr bound on the integrality gap of 2-matchings; since the constraints of the subtour LP are a superset of the fractional 2-matching constraints, this implies the ratio is at most $\frac{4}{3}$.

---

[1] We note that what we refer to here as 2-matchings are also sometimes called 2-factors.

[2] In fact, they show in this case the optimal 2-matching has cost at most $(3k+1)/(3k)$ times the cost of the optimal solution to the subtour LP.
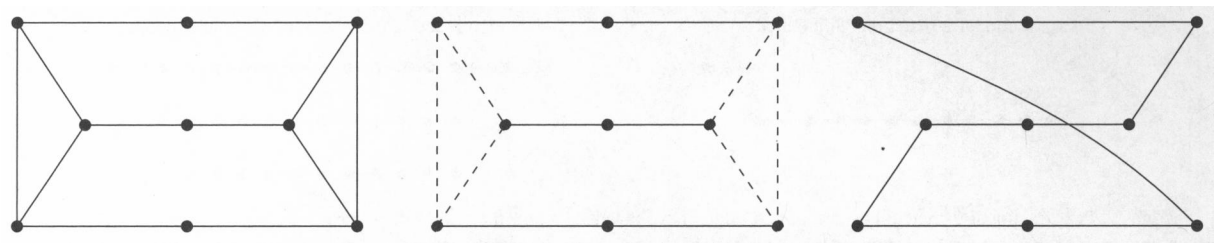
FIGURE 1. Illustration of the worst example known for the ratio of 2-matchings to the subtour LP. The figure on the left shows the instance: all edges in the graph have cost 1; all other edges have cost 2. The figure in the center gives the subtour LP solution, in which the dotted edges have x-value $\frac{1}{2}$ and the solid edges have x-value 1; this is also an optimal fractional 2-matching. The figure on the right gives an optimal 2-matching, which is also the optimal tour.

The work of Goemans [14] has some bearing on this conjecture. He studies the following linear program, which is essentially the same as the subtour LP in the case in which edge costs obey triangle inequality:

$$\min \sum_{e \in E} c(e) x(e)$$

$(SUBT')$ subject to: $\sum_{e \in \delta(S)} x(e) \geq 2, \quad \forall S \subset V, \ S \neq \varnothing,$ (4)

$$x(e) \geq 0, \quad \forall e \in E.$$ (5)

Goemans shows (among other things) that adding comb inequalities to this LP can increase the LP value by at most $\frac{10}{9}$; more precisely, he shows that if x is a feasible solution to $(SUBT')$, then $\frac{10}{9}x$ is feasible for the LP obtained by adding comb inequalities to $(SUBT')$. It is known that adding a subset of the comb inequalities to the degree constraints (1) and bounds (3) gives the 2-matching polytope. This would imply the Boyd-Carr conjecture if it were known that there is an optimal solution that obeys the degree constraints when the comb inequalities are added to $(SUBT')$; as mentioned above, it can be shown that there is an optimal solution for $(SUBT')$ that obeys the degree constraints when the edge costs obey the triangle inequality. But we do not know whether there is an optimal solution that obeys the degree constraints if the comb inequalities are added.[3]

The contribution of this paper is to improve our state of knowledge for the subtour LP by proving Conjecture 1.

We start by showing that in some cases the cost of an optimal 2-matching is at most $\frac{10}{9}$ the cost of a fractional 2-matching; in particular, we show this is true whenever the support of the fractional 2-matching has no cut edge. This is not implied by Conjecture 1 since there exist fractional 2-matching solutions for which the support has no cut edge but that are not feasible solutions to the subtour LP (and this holds even if we restrict ourselves to fractional 2-matching solutions that consist of a single component); see Figure 2 for an example.

As the first step in this proof, we give a simplification of the Boyd and Carr result bounding the integrality gap for 2-matchings by $\frac{4}{3}$. In the case that the support of an optimal fractional 2-matching has no cut edge, the proof becomes quite simple. The perfect matching polytope plays a crucial role in the proof: we use the matching edges to show us which edges to remove from the solution in addition to showing us which edges to add. We note that this idea was independently developed in the recent work of Mömke and Svensson [17], but also previously appeared in the reduction of the 2-matching polytope to the matching polytope; see, for instance, Schrijver [24, Section 30.7]. We also use a notion from Boyd and Carr [5] of a *graphical* 2-matching: in a graphical 2-matching, each node has degree either two or four; each edge has zero, one, or two copies; and each component has size at least three. Given the triangle inequality, we can shortcut any graphical 2-matching to a 2-matching of no greater cost.

To obtain our proof of the Boyd-Carr conjecture, we give a polyhedral formulation of the graphical 2-matching problem and use it to prove Conjecture 1. If x is a feasible solution for the subtour LP, then, roughly speaking, we show that $\frac{10}{9}x$ is feasible for the graphical 2-matching polytope. Our previous results give us intuition for the precise mapping of variables that we need. Using the graphical 2-matching polytope allows us to overcome the issues with the degree constraints faced in trying to use Goemans' results (Goemans [14]).

All the results above can be made algorithmic and have polynomial-time algorithms, though we do not explicitly determine running times.

---

[3] To quote Goemans [14, p. 348]: "One might wonder whether the worst-case improvements remain unchanged when one adds the degree constraints $x(\delta\{i\}) = 2$ for all $i \in V$ and restricts one's attention to cost functions satisfying the triangle inequality. We believe so but have been unable to prove it. The result would follow immediately if one could prove that the degree constraints never affect the value of the relaxation when the cost function satisfies the triangle inequality."
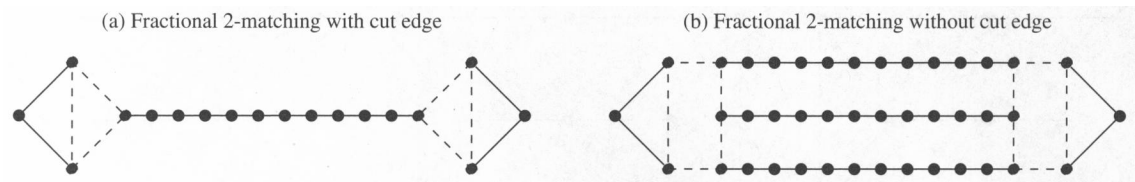
(a) Fractional 2-matching with cut edge   (b) Fractional 2-matching without cut edge

FIGURE 2. Two fractional 2-matchings; the dotted edges have x-value $\frac{1}{2}$ and the solid edges have x-value 1. All edges in the support graph have cost 1; the cost between any other pair of nodes is equal to the cost of the shortest path in the support graph. The figure on the left is a fractional 2-matching with cut edge, showing that the worst case ratio of the 2-matching cost over the fractional 2-matching cost is at least $\frac{4}{3}$. The figure on the right is a fractional 2-matching for which the support has no cut edge but which is not a feasible solution to the subtour LP since the subtour constraint is violated for the set containing the three rightmost nodes.

We conclude by posing a new conjecture: namely that the worst-case integrality gap of the subtour LP is achieved for solutions to the subtour LP that are fractional 2-matchings (that is, for instances such that adding the subtour constraints to the degree constraints and the bounds on the variables does not change the objective function value).

In a companion paper, Qian et al. [22] show that the proof of the Boyd-Carr conjecture can be used to help bound the integrality gap of the subtour LP for the 1,2-TSP. They show that the gap is at most $\frac{19}{15} \approx 1.267 < \frac{4}{3}$. They also give a proof that the cost of the optimal 2-matching is at most $\frac{10}{9}$ times the cost of a fractional 2-matching in the case that $c(i, j) \in \{1, 2\}$, which gives an alternative proof of the Boyd-Carr conjecture in this case.

Our paper is structured as follows. We introduce basic terms and notation in §2. In §3, we rederive the Boyd-Carr integrality gap for 2-matchings and show that the gap is at most $\frac{10}{9}$ in the case that the fractional 2-matching has no cut edge. In §4, we give the polytope for graphical 2-matchings and show how to use it to prove the Boyd-Carr conjecture. Finally, we close with our new conjecture in §5.

2. Preliminaries.
We will work extensively with fractional 2-matchings; that is, basic optimal solutions x to the LP (*SUBT*) with only constraints (1) and (3). For convenience we will abbreviate "fractional 2-matching" by F2M and "2-matching" by 2M. F2Ms have the following well-known structure (attributed to Balinski [3]). Each connected component of the support graph (that is, the edges e for which $x(e) > 0$) either is a cycle on at least three nodes with $x(e) = 1$ for all edges e in the cycle or consists of odd-sized cycles with $x(e) = \frac{1}{2}$ for all edges e in the cycle connected by paths of edges e with $x(e) = 1$ for each edge e in the path (see Figure 3 for an example). We call the former components *integer components* and the latter *fractional components*. Many of our results focus on transforming an F2M into a 2M, in which all components are integer. For that reason, we will often focus solely on how to transform the fractional components into integer components. We then call the edges of fractional components for which $x(e) = \frac{1}{2}$ *cycle edges* and the edges for which $x(e) = 1$ *path edges*. Note that removing a cycle edge can never disconnect a fractional component. If removing a path edge disconnects a fractional component, we call it a *cut edge*. The associated path of the path edge we will call a *cut path* since every edge in it will be a cut edge. We will say that a fractional 2-matching is *connected* if it has a single component.

We will use a concept introduced by Boyd and Carr [5] of a *graphical* 2-matching (G2M). As stated above, in a graphical 2-matching, each node has degree either two or four; each edge has zero, one, or two copies; and each component has size at least three. Given the triangle inequality, we can shortcut any G2M to a 2M of no greater cost. Our techniques for transforming an F2M to a 2M actually find G2Ms.
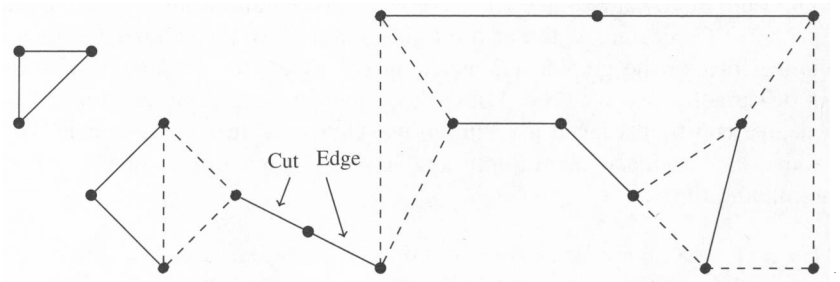

Cut Edge

FIGURE 3. Illustration of a fractional 2-matching that has two components, one integer component and one fractional component. The dotted edges have x-value $\frac{1}{2}$ and will be called cycle edges, and the solid edges have x-value 1 and will be called path edges.

We will often need to find minimum-cost perfect matchings. By a result of Edmonds [11], the perfect matching polytope is defined by the following linear program (M):

$$\min \sum_{e \in E} c(e)x(e)$$

$$(M) \quad \text{subject to:} \quad \sum_{e \in \delta(i)} x(e) = 1, \quad \forall i \in V, \tag{6}$$

$$\sum_{e \in \delta(S)} x(e) \geq 1, \quad \forall S \subset V, \ |S| \text{ odd}, \tag{7}$$

$$x(e) \geq 0, \quad \forall e \in E. \tag{8}$$

3. 2-matching integrality gaps. In this section, we bound the cost of a G2M in terms of an F2M via combinatorial methods. We start by giving a proof of a result of Boyd and Carr [5] that there is a G2M of cost at most $\frac{4}{3}$ the cost of an F2M. Our proof is somewhat simpler than theirs, but more importantly, it introduces the main ideas that we will need to obtain other results. We then show that if the F2M has no cut edges, we can improve the bound from $\frac{4}{3}$ to $\frac{10}{9}$. The main idea of this section is that given an F2M, we define a matching problem and compute a perfect matching. The perfect matching tells us how to modify the fractional components by either duplicating or removing edges so that we obtain a G2M. We then relate the cost of the perfect matching found to the F2M by providing a feasible solution to the perfect matching LP (M). We will need the following result of Naddef and Pulleyblank [20]; we give the proof since we will use some of its ideas later on.

Lemma 1 (Naddef and Pulleyblank [20]). *Let G be a cubic, 2-edge-connected graph with edge costs $c(e)$ for all $e \in E$. Then there exists a perfect matching in G of cost at most $\frac{1}{3} \sum_{e \in E} c(e)$.*

Proof. The main idea is to show that $x(e) = \frac{1}{3}$ is a feasible solution to the matching polytope (M). The lemma then follows because (M) has integer extreme points. Since G is cubic, $|V|$ must be even, and $\sum_{e \in \delta(i)} x(e) = 1$. Now consider any $S \subset V$ with $|S|$ odd. Because G is cubic, it must be that $|\delta(S)|$ is odd, and since G is 2-edge-connected, $|\delta(S)| \geq 2$. Therefore, $|\delta(S)| \geq 3$, and $\sum_{e \in \delta(S)} x(e) \geq 1$. $\square$

We note that Lemma 1 also holds for cubic, 2-edge-connected multigraphs.

Theorem 1. *There exists a G2M of cost at most $\frac{4}{3}$ times the cost of an F2M if the F2M has no cut edge.*

Proof. As described above, it is sufficient to focus on a single fractional component of the F2M. Let G be the support graph of this component.

To find the G2M, we find a minimum-cost perfect matching on the (multi)graph G' we obtain by replacing each path in G by a single edge, which we will call (at the risk of some confusion) a path edge. We set the cost of this edge to be the cost of the path in G, and we set the cost of a cycle edge in G' to the *negative* of the cost of the cycle edge in G. Note that G' is cubic and 2-edge-connected because the support graph G of the F2M has no cut edge.

Given a minimum-cost perfect matching in G', we construct a G2M in G by first including all paths from G. If a path edge is in the matching in G', we double the path in G. If a cycle edge is *not* in the matching in G', then we include the cycle edge in the G2M in G; otherwise, we omit the cycle edge. See Figure 4 for an illustration.

We first show that this indeed defines a G2M: for each node, the degree is four if the perfect matching contains the path edge incident to the node (since in that case, the two cycle edges on the node cannot be in the perfect matching, and hence both are added to the G2M together with two copies of the path), and it is two otherwise (since one cycle edge is in the perfect matching, and hence only the other cycle edge and one copy of the path are added to the graphical 2-matching). Note that any connected component indeed has at least three nodes: Every connected component contains (the edges corresponding to) at least one path edge; now, either a path edge corresponds to a path of length at least 2, or the path edge corresponds to a single edge. In the latter case, at least two cycle edges incident to the endpoints of the path edge are also contained in the connected component: if the path edge is in the perfect matching, all four cycle edges incident to its endpoints are in the connected component; if the path edge is not in the perfect matching, the component will contain one cycle edge incident to each endpoint of the path edge. Finally, these cycle edges are distinct from the edge that is the path since G has no doubled edges.

We let C denote the sum of the costs of the cycle edges and P the cost of the paths. Note that the cost of the F2M solution is $\frac{1}{2}C + P$. The cost of the G2M is equal to the cost of all edges in the support graph $(P + C)$
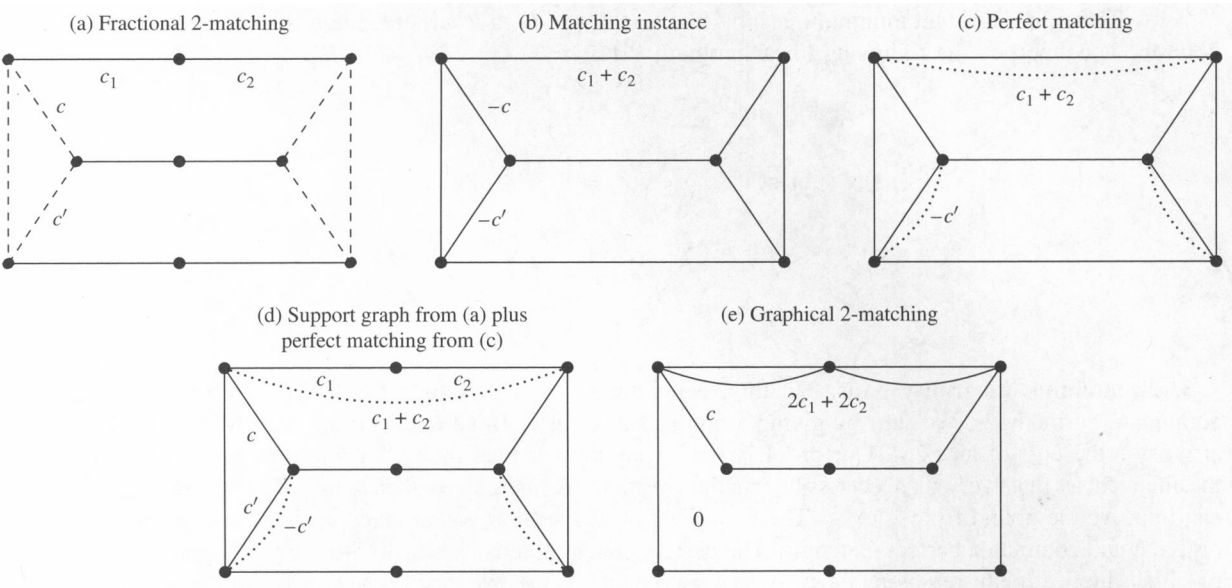
FIGURE 4. Illustrations of the algorithm in Theorem 1. Dotted edges are edges in the perfect matching. The cost next to the dotted edges represents the corresponding matching cost.

plus the cost of the perfect matching. Because G' is cubic and 2-edge-connected, we can invoke Lemma 1 to show that the perfect matching has cost at most a third the cost of the edges in G' or at most $\frac{1}{3}(P - C)$. Hence the cost of the G2M is at most

$$P + C + \frac{1}{3}(P - C) = \frac{4}{3}P + \frac{2}{3}C = \frac{4}{3}\left(P + \frac{1}{2}C\right)$$

or at most $\frac{4}{3}$ the cost of the F2M solution, as claimed. □

The idea of using edges from a perfect matching to decide which edges to include in a matching and which edges to remove has also been used recently by Mömke and Svensson [17].

We now modify the proof of the theorem above so that the result extends to the case in which the F2M has cut edges.

THEOREM 2 (BOYD AND CARR [5]). *There exists a G2M of cost at most $\frac{4}{3}$ times the cost of an F2M.*

PROOF. As described above, it is sufficient to focus on a single fractional component of the F2M, and we let G be the support graph of this component.

We once again create a new graph G' from G so that we can later define a matching problem in G'. The matching will again show us how to create a G2M in G. We extend the previous construction to deal with the case when the support graph has cut paths. We introduce a gadget in G' for each cut path in G, which replaces the cut path and its two endpoints. The other paths in G are again replaced by single edges in G' of cost equal to the cost of the path. Each cycle edge in G is also in G' with cost equal to the negative of its cost in G.

To introduce the cut-path gadget, we begin by using an idea of Boyd and Carr [5]; namely, we only need to consider three *patterns* to get an almost feasible graphical 2-matching on the cut path when we allow ourselves to increase the cost by a third compared to the F2M. Suppose the cut path has l edges and $l + 1$ nodes, and let $k = \lfloor l/3 \rfloor$. We can remove every third edge and double the remaining edges to obtain groups of nodes that are 2-edge-connected, where we get k groups of three nodes that are G2M components plus one group of $l - 3k + 1 \in \{0, 1, 2\}$ nodes. Alternatively, we could remove every third edge, starting from the first edge, and double the remaining edges, in which case the first group has one node; the next k or $k - 1$ groups have three nodes; and the last group again has one, two, or three nodes. The final pattern removes every third edge, starting from the second edge, so that the first group has two nodes; the next k or $k - 1$ groups have three nodes; and again the last group has one, two, or three nodes. Figure 5 illustrates the three patterns for $l = 9$. Note that if $l < 3$, then we can define the three patterns in a similar way. However, since there is no third edge (and no second edge if $l = 1$), the patterns obtained by removing the third (or second if $l = 1$) edge and doubling the remaining edges just result in doubling all the edges in the path. Hence, for $l = 1$, we get one pattern containing
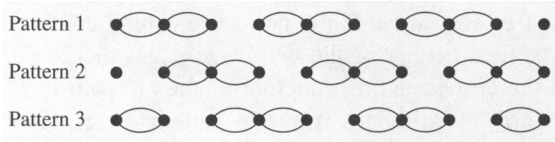
FIGURE 5. Illustrations of patterns for $l = 9$.

no edges and two patterns containing the doubled path; for $l = 2$, we get one pattern containing the doubled path and two patterns containing two copies of, respectively, the first and second edge of the path.

To get a G2M that contains a certain pattern, we will ensure that if a group has size less than three, the G2M will include the two cycle edges incident to the first node (if the group is at the start of the pattern) or last node (if the group is at the end of the pattern).

We remark that for $l \geq 2$, there is exactly one pattern that starts with a group of size one, two, and three, and hence two patterns need the G2M to include two cycle edges incident to the first node of the cut path. On the other hand, there is also exactly one pattern that ends with a group of size one, two, and three (the length of the cut path determines which of the three patterns ends with a group of size three: it is the second pattern if $l \pmod 3 = 0$, the third pattern if $l \pmod 3 = 1$, and the first pattern if $l \pmod 3 = 2$); hence there are also two patterns that need the G2M to include the two cycle edges incident to the last node of the cut path. If $l = 1$, there is one pattern that starts and ends with a group of size one; the other two patterns both start and end with a group of size two.

We are now ready to define the cut-path gadget. We replace each endpoint of the cut path in G by three dummy nodes connected by a path of length two in G'; each of these new edges will have cost zero. We replace the cut path by three *pattern edges*. Each dummy node is connected to one pattern edge, while ensuring that the middle node is connected to the pattern edge corresponding to the pattern which does not need two cycle edges incident to the endpoint of the cut path (i.e., the pattern for which the group containing the endpoint has size three). We set the cost of a pattern edge to the cost of the edges in the corresponding pattern. See Figure 6 for an illustration of the gadget when $l = 9$. If $l = 1$, the patterns associated with the illustration in Figure 6 are chosen so that pattern 3 corresponds to not adding the edge of the cut path, and patterns 1 and 2 correspond to doubling the edge.

If we replace each cut path in G by a cut-path gadget in G', once again G' will be a cubic graph. It is not hard to check that G' is also 2-edge-connected because we have replaced the cut path in G with three pattern edges crossing the cut in G'.

We argue that there is a minimum-cost perfect matching that uses exactly one pattern edge from each cut-path gadget. Note that we replace only the cut paths in G by a cut gadget in G', which means that a perfect matching in G' contains an odd number of pattern edges in a gadget. If it contains three pattern edges, then we could find a matching of no greater cost by choosing only one pattern edge, namely, the pattern edge that is not incident to the middle node for the either one of its endpoints. Note that we can add two edges of cost 0 that connect the four nodes incident to the other two pattern edges to again have a perfect matching without increasing the cost.

Now we show how to obtain a G2M in G from the minimum-cost perfect matching in G'. In the G2M we include all edges from G that are in paths that are not cut paths, the cycle edges in G that are *not* chosen by the perfect matching, duplicates of edges in non-cut paths in G that are chosen by the perfect matching, and the edges in a pattern if the corresponding pattern edge is in the perfect matching.

We argue that this set of edges is a G2M in G. Note that if the perfect matching contains only the pattern edge incident to the middle node, then the two cycle edges that are adjacent to the gadget are also in the matching. Hence the corresponding endpoint in G of the cut path has no cycle edges incident to it in the G2M, but since the pattern edge is incident to the middle node, the corresponding pattern ensures that the node has degree two
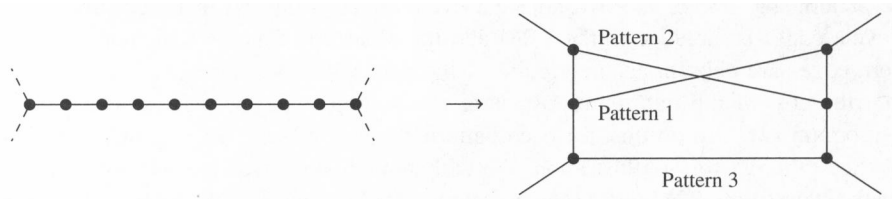

FIGURE 6. Pattern gadget for $l = 9$.

and is in a connected component of size three. If the perfect matching contains the pattern edge incident to a node other than the middle node, then neither of the two cycle edges that are adjacent to the gadget in G' are in the perfect matching. Hence the corresponding endpoint of the cut path in G has both of these cycle edges incident to it in the G2M and zero or two edges from the pattern corresponding to the chosen pattern edge. Hence the node has degree two or four and it is in a connected component of size at least three.

As before, because G' is cubic and 2-edge-connected, we can apply Lemma 1 to bound the cost of the perfect matching in G'. Let P_1 be the cost of the paths in G that are not cut paths and P_2 the cost of the cut paths in G so that the cost of the F2M is $P_1 + P_2 + \frac{1}{2}C$. Note that the cost of the three pattern edges in the gadget corresponding to a cut path sums up to four times the cost of the cut path. Thus the total cost of the edges in G' is $P_1 + 4P_2 - C$. By Lemma 1, the cost of the perfect matching in G' is at most $\frac{1}{3}(P_1 + 4P_2 - C)$, and, as we argued above, the perfect matching in G' contains at least one pattern edge corresponding to each cut path. The cost of the G2M corresponding to the minimum-cost perfect matching is therefore at most

$$ P_1 + C_1 + \frac{1}{3}(P_1 + 4P_2 - C) = \frac{4}{3}(P_1 + P_2) + \frac{2}{3}C = \frac{4}{3}\left(P_1 + P_2 + \frac{1}{2}C\right), $$

as claimed. □

We now show how to use the ideas behind the cut-path gadget to obtain a better G2M if no cut paths exist.

THEOREM 3. *If an F2M has no cut edge, then there exists a G2M of cost at most $\frac{10}{9}$ times the cost of the F2M.*

PROOF. Once again we define a new graph G' from the support graph G of a fractional component of the optimal F2M. Each cycle edge in G is in G' with cost that is the negative of its cost in G. Each path in G and its two endpoints are replaced by the cut-path gadget used in the proof of Theorem 2. The costs of the pattern edges in G' are slightly different than in the previous proof: we subtract the cost of the original path from the cost of the edges in the pattern to obtain the cost of the pattern edge. In other words, the cost of a pattern edge in G' is obtained by adding once the cost of the edges that appear twice in the pattern and subtracting the cost of the edges that do not appear in the pattern. Note that the sum of the costs of the three pattern edges in G' is equal to the cost of the original path in G. Also, note that the sum of the costs of any two pattern edges in G' is nonnegative: an edge on the path contributes its cost either positively to one pattern and negatively to the other or positively to both patterns.

We first argue that there is a minimum-cost perfect matching that chooses either zero or one pattern edge in each cut-path gadget. Suppose the perfect matching contains two pattern edges in a gadget. Note that on both sides of the gadget, these pattern edges must be incident to the middle node; otherwise, some middle node is not matched. Hence the four endpoints of the two pattern edges are connected in G' by two edges of cost zero (i.e., by two vertical edges in the gadget as drawn in Figure 6). By the observation above, the cost of the two pattern edges is nonnegative, and so we can remove the two pattern edges from the matching and add the two edges of cost zero without increasing the cost of the matching. By the same argument, we can handle the case that the perfect matching contains three pattern edges from a gadget by choosing the pattern edge that is not incident to the middle node on both sides of the gadget and replacing the other two pattern edges in the matching by the cost zero edges that connect their endpoints.

Therefore, we can assume the perfect matching chooses either zero or one pattern edge in a gadget. If it chooses zero pattern edges, then we add the path from G to the G2M. Otherwise, the pattern corresponding to the chosen pattern edge is added to the G2M. We also add the cycle edges to the G2M corresponding to the cycle edges that are *not* in the perfect matching.

By almost the same arguments as before, the solution constructed is indeed a G2M. The only case not covered by previous arguments is the case in which zero pattern edges are chosen in G'. Then it must be the case for each side of the pattern that one of the two adjacent cycle edges is chosen in G' and the other is not, so that one of the two cycle edges is included in the G2M and the other not. Since we include the path from G in the G2M if no pattern edges are chosen, each endpoint of the path will have degree two.

To argue about the cost of the minimum-cost perfect matching in G', we create a feasible solution for the matching linear program (M). To do this, for each pattern edge e, we set $x(e) = \frac{1}{9}$, and for every other edge e', we set $x(e') = \frac{4}{9}$. See Figure 7 for an illustration. We will show this is a feasible solution in a moment. Let P be the cost of the path edges in the F2M and C the cost of the cycle edges so that the F2M has cost $P + \frac{1}{2}C$. Since the sum of the cost of the pattern edges in a gadget is equal to the cost of the path, the cost of this solution for
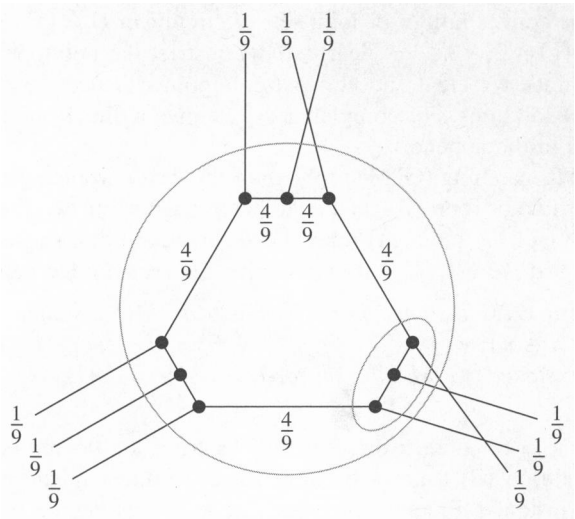
FIGURE 7. Illustration of the feasible fractional solution for the matching linear program used in the proof of Theorem 3.

(M) is $\frac{1}{9}P - \frac{4}{9}C$, and there exists a perfect matching of cost at most this much. Thus the cost of the G2M is at most

$$P + C + \left(\frac{1}{9}P - \frac{4}{9}C\right) = \frac{10}{9}P + \frac{5}{9}C = \frac{10}{9}\left(P + \frac{1}{2}C\right),$$

as claimed.

To see that x is a feasible solution for (M), consider any cut such that the number of nodes on each side of the cut is odd. If there exists a cycle from the F2M such that not all nodes in the gadgets for the nodes in the cycle are on the same side of the cut, then there are two edges crossing the cut with value $\frac{4}{9}$. Since G' is cubic, if the cut has odd size, then the total number of edges crossing the cut is odd, and there must be at least one more edge in the cut with value at least $\frac{1}{9}$. Hence the total value on the edges crossing the cut is at least one. For any other cut, since there is no cut path in G, there are at least three gadgets crossing the cut in G'. Since each gadget contains three pattern edges, the value of the edges crossing the cut is again at least one. $\square$

4. A polyhedral proof of the Boyd-Carr conjecture.
We will generalize the result in Theorem 3 and show that the ratio between the cost of the optimal 2-matching and the subtour LP is at most $\frac{10}{9}$. In the combinatorial proofs of the previous section, we heavily used that F2Ms have a nice, simple structure, and, unfortunately, this does not hold for the subtour LP solution. We therefore turn to a polyhedral rather than a combinatorial proof. We derive a polyhedral description for graphical 2-matchings, and we then use this description to construct a feasible (fractional) G2M solution from any solution to the subtour LP of cost not more than $\frac{10}{9}$ times the value of the subtour LP. The manner in which the feasible G2M solution is defined based on a solution to $(SUBT)$ is a generalization of the proof of Theorem 3.

We start by giving a polyhedral description of a generalization of 2-matching, where the node set consists of "mandatory nodes" (V_{man}) and "optional nodes" (V_{opt}). The former need to have degree two in the solution, whereas the latter can have degree zero or two. We will refer to this problem as the 2-Matching with Optional Nodes Problem (2MO).

THEOREM 4. *Let* $G = (V_{\text{man}} \cup V_{\text{opt}}, E')$ *be a 2MO instance. The convex hull of integer 2MO solutions is given by the following polytope*:

$$\sum_{e \in \delta(i)} y(e) = 2, \quad \forall i \in V_{\text{man}}, \tag{9}$$

$$\sum_{e \in \delta(i)} y(e) \leq 2, \quad \forall i \in V_{\text{opt}}, \tag{10}$$

$$\sum_{e \in \delta(S) \setminus F} y(e) + \sum_{e \in F}(1 - y(e)) \geq 1, \quad \forall S \subseteq V, \ F \subseteq \delta(S), \ F \text{ matching}, \ |F| \text{ odd}, \tag{11}$$

$$0 \leq y(e) \leq 1, \quad \forall e \in E. \tag{12}$$

Aráoz et al. [2] show that the convex hull of disjoint sets of circuits of $G = (V, E)$ is given by constraints (11), constraints (12), and $\sum_{e \in \delta(i)} y(e) \leq 2$, $\forall i \in V$. Clearly, if we restrict the polytope by replacing some inequality constraints by equality constraints, we create no new extreme points; hence the extreme points of the polytope in Theorem 4 are integer 2MO solutions. For completeness, we give a direct proof of Theorem 4 by a reduction to a perfect matching problem in the appendix.

We now relate the graphical 2-matching (G2M) problem to the 2MO problem. Recall the definition of a G2M: (i) each node has degree either two or four; (ii) each edge has zero, one, or two copies; and (iii) each component has size at least three. We will (for the moment) relax the second condition so that each edge has at most three copies. We remind the reader that we use (i, j) to denote the *undirected* edge between i and j.

LEMMA 2. *We can reduce a G2M instance $G = (V, E)$ to a 2MO instance $G' = (V', E')$ as follows: Let $V'_{\mathrm{man}} = \{i_m \colon i \in V\}$, $V'_{\mathrm{opt}} = \{i_o \colon i \in V\}$, $V' = V'_{\mathrm{man}} \cup V'_{\mathrm{opt}}$, and $E' = \{(i_m, j_m) \colon (i, j) \in E\} \cup \{(i_m, j_o) \colon (i, j) \in E\}$. We add an edge (i, j) to the (relaxed) G2M solution for each edge (i_m, j_m), (i_o, j_m), and (i_m, j_o) that is in the associated 2MO solution.*

PROOF. Note that condition (i) for node i directly follows from the degree constraints for nodes i_m and i_o in the reduction. Relaxed condition (ii) follows because for every edge in the G2M instance, there are three associated edges in the 2MO instance. Finally, since each node i_m has degree two in the 2MO solution, there cannot be a component of size one. Suppose there is a component of size two. Then this must be an isolated doubled or quadrupled edge, say, (i, j), because of the degree constraints. Clearly we can't have a quadrupled edge since there are at most three copies of edge (i, j) in the 2MO solution. We also can't have an isolated doubled edge: in order for the edge to be isolated, we would need (i_m, j_m) and (i_m, j_o) to be in the 2MO solution. But then j_o must have degree 2, and its second edge must be (j_o, k_m) for some $k \neq i$, j, since there are no edges (i_o, j_o), or (j_o, j_m) in the 2MO instance. $\square$

If the edges have nonnegative costs, we may assume without loss of generality that each edge appears at most twice in an optimal G2M solution: if any edge appears three times, we can remove two copies of it without affecting the parity of its endpoints, the connected components retain their sizes, and the cost cannot increase.

We will now use a solution x to the subtour LP on $G = (V, E)$ to define a feasible solution y to the 2MO instance $G' = (V', E')$ associated with the graphical 2-matching problem on G. If we treat every edge the same and set the y-value to $(1 - \alpha)x(i, j)$ on the edge between the mandatory copies of i and j, then this implies we have to set the y-value to $\alpha x(i, j)$ on the edge from a mandatory to an optional copy. The following lemma states that taking $\alpha = \frac{1}{9}$ indeed defines a fractional solution to the 2MO instance corresponding to the G2M instance G if x is a feasible solution to the subtour LP on G.

LEMMA 3. *Given a graph $G = (V, E)$, let x be a feasible solution to the subtour LP for G. Then the following solution is a feasible solution to the 2MO instance $G' = (V', E')$ associated with the graphical 2-matching instance given by G for $\alpha = \frac{1}{9}$:*

$$y(i_m, j_m) = (1 - \alpha)x(i, j)$$
$$y(i_m, j_o) = \alpha x(i, j)$$
$$y(i_o, j_m) = \alpha x(i, j)$$

for all $(i, j) \in E$.

Note that the cost of the constructed fractional 2MO solution is exactly $\frac{10}{9}$ times the cost of the solution of the subtour LP. Thus our result follows immediately from the lemma, Theorem 4, and Lemma 2.

COROLLARY 1. *There exists a G2M of cost at most $\frac{10}{9}$ times the value of the subtour LP.*

PROOF OF LEMMA 3. We need to show that y satisfies the Constraints (9)–(12) on G', where G' is defined as in Lemma 2. Constraints (9), (10), and (12) are obviously met, and we only need to show that constraints (11) are met. To this end, fix $S \subseteq V'$, $F \subseteq \delta(S)$, where F is a matching and $|F|$ is odd. For simplicity, for any set of edges $X \subseteq E'$, we define $y(X) = \sum_{e' \in X} y(e')$. We need to show that

$$y(\delta(S)) + |F| - 2y(F) \geq 1. \tag{13}$$

We call a node $i \in V$ a *split node* if $|\{i_o, i_m\} \cap S| = 1$. We now consider three cases.

Case 1 F contains an edge (i_o, j_m) for some $i_o \in V_{\mathrm{opt}}$ and $j_m \in V_{\mathrm{man}}$, and i is a split node. Since i is a split node, exactly one of the edges (i_o, j'_m) and (j'_m, i_m) crosses the cut for all $j' \in V$. If (j'_m, i_m) crosses the cut, the contribution to the left-hand side of (13) is $(1 - \alpha)x(i, j')$ if $(j'_m, i_m) \notin F$ or $1 - (1 - \alpha)x(i, j')$ if $(j'_m, i_m) \in F$. Note that since $x(i, j') \leq 1$ and $\alpha \leq \frac{1}{2}$, the contribution is always at least $\alpha x(i, j')$. If (i_o, j'_m) crosses the cut, the contribution to the left-hand side of (13) is either $\alpha x(i, j')$ if $(i_o, j'_m) \notin F$ or $1 - \alpha x(i, j')$ if $(i_o, j'_m) \in F$. Since F is a matching, note that $(i_o, j'_m) \in F$ only for $j' = j$.
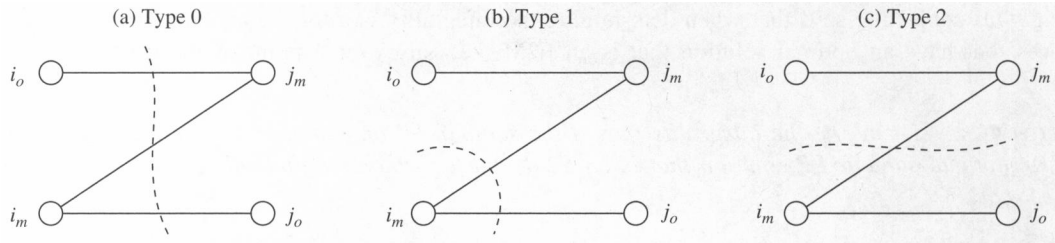
FIGURE 8. Illustrations of the three types of edges in F in Claim 1. The y-value on the top and bottom edge is $\alpha x(i, j)$ and the y-value on the middle edge is $(1 - \alpha)x(i, j)$.

The contribution to the left-hand side of (13) by any other edge e is nonnegative since $F \subset \delta(S)$ and $y(e) \leq 1$. We thus have that the left-hand side of (13) is at least $\alpha \sum_{j' \in V} x(i, j') + 1 - 2\alpha x(i, j) \geq 1$, where the last inequality follows since $\sum_{j' \in V} x(i, j') = 2$ and $x(i, j) \leq 1$.

Case 2 F contains an edge (i_o, j_m) for some $i_o \in V_{\mathrm{opt}}$ and $j_m \in V_{\mathrm{man}}$, and i is not a split node. In this case the edge (j_m, i_m) must also cross the cut, and $(j_m, i_m) \notin F$ since F is a matching. Hence, the left-hand side of (13) is at least $x(i, j) + 1 - 2\alpha x(i, j) \geq 1$ for any $\alpha \leq \frac{1}{2}$.

Case 3 It remains to consider the case when F only contains edges between mandatory nodes. Since F is not empty, it must be the case that S contains a node i_m for some $i \in V$, and $V' \backslash S$ contains a node i'_m for some $i' \in V$. Therefore, if we let $R = \{i: i_m \in S\}$, then $\varnothing \subsetneq R \subsetneq V$, and hence $x(\delta(R)) \geq 2$. Let $H = \{(i, j) \in \delta(R): (i_m, j_m) \in F\}$. Since for each edge $(i, j) \in \delta(R)$, we know that $(i_m, j_m) \in \delta(S)$, we get that $y(\delta(S)) \geq (1 - \alpha)x(\delta(R))$. Also, since F only contains edges between mandatory nodes, we have $y(F) = (1 - \alpha)x(H)$. We can now conclude that the left-hand side of (13) is at least $(1 - \alpha)x(\delta(R)) + |F| - 2(1 - \alpha)x(H) \geq (1 - \alpha)2 + |F| - 2(1 - \alpha)x(H)$. If $|F| = |H| = 1$, then (13) holds since $x(H) \leq 1$.

The only case remaining is the case when F contains only edges between mandatory nodes and $|F| \geq 3$. We will prove the following claim.

CLAIM 1.   *Let $S \subset V'$ and let $F \subseteq \delta(S) \cap \{(i_m, j_m): i, j \in V\}$ be a matching. Let $H = \{(i, j): (i_m, j_m) \in F\}$. Then $y(\delta(S)) \geq (1 + \alpha)x(H)$.*

Using the claim, we have that $y(\delta(S)) + |F| - 2y(F) \geq (1 + \alpha)x(H) + |F| - 2(1 - \alpha)x(H) = |F| - (1 - 3\alpha)x(H)$. For any $\alpha \leq \frac{1}{3}$, this is at least $3\alpha|F|$ since $x(e) \leq 1$ and $|H| = |F|$. Now, since we assume that $|F| \geq 3$, and we chose $\alpha = \frac{1}{9}$, we have that $3\alpha|F| \geq 1$.

It remains to prove the claim. Recall that i is a split node if $|\{i_o, i_m\} \cap S| = 1$. For an edge $(i_m, j_m) \in F$, either zero, one, or both of the nodes i, j are split nodes. We will refer to the edge (i, j) as a Type k edge if k of the nodes i, j are split nodes for $k \in \{0, 1, 2\}$. See Figure 8 for an illustration of the three possibilities. In the first case, all three edges in the gadget for edge (i, j) are in $\delta(S)$, and together they contribute $(1 + \alpha)x(i, j)$ to $y(\delta(S))$. To complete the proof, we will need to use that for a split node i and an arbitrary node $j \in V$, exactly one of the edges (i_o, j_m) and (j_m, i_m) crosses the cut. More strongly, $y(\delta(S) \cap \{(i_o, j_m), (j_m, i_m), (i_m, j_o)\})$ is at least $\alpha x(i, j)$ if i is a split node, and it is at least $2\alpha x(i, j)$ if both i and j are split nodes. Hence, we can use $y(\delta(S))$ to assign each split node i an amount $\alpha \sum_j x(i, j) = 2\alpha$.

We now consider how much of $y(\delta(S))$ remains unassigned. For an edge (i, j) of Type 0, neither i nor j is a split node, and hence $(1 + \alpha)x(i, j)$ remains unassigned. For an edge (i, j) of Type 1, $y(\delta(S) \cap \{(i_o, j_m), (j_m, i_m), (i_m, j_o)\}) = x(i, j)$, and since only i is a split node, $(1 - \alpha)x(i, j)$ remains unassigned. For an edge (i, j) of Type 2, $y(\delta(S) \cap \{(i_o, j_m), (j_m, i_m), (i_m, j_o)\}) = (1 - \alpha)x(i, j)$; since both i and j are split nodes, $(1 - 3\alpha)x(i, j)$ remains unassigned.

Let F_i be the number of edges in F of Type 0, 1, and 2, respectively, and let $H_i = \{(i, j): (i_m, j_m) \in F_i\}$. Let n_s be the number of split nodes. Then we have argued that $y(\delta(S)) \geq (1 + \alpha)x(H_0) + (1 - \alpha)x(H_1) + (1 - 3\alpha)x(H_2) + 2\alpha n_s$. Now, since $F \subseteq (\delta(S) \cap \{(i_m, j_m): i, j \in V\})$ is a matching, $n_s \geq |H_1| + 2|H_2| \geq x(H_1) + 2x(H_2)$, so $y(\delta(S)) \geq (1 + \alpha)x(H)$, as claimed.   □

5. Conjectures and conclusions.

I conjecture that there is no [polynomial-time] algorithm for the traveling salesman problem. My reasons are the same as for any mathematical conjecture: (1) It is a legitimate mathematical possibility, and (2) I do not know.

—Edmonds [12, p. 234]

We conclude our paper with a conjecture. We do so in the spirit of Jack Edmonds, quoted above; we do not know whether the conjecture is true or not, but we think that even a proof that the conjecture is false would be

interesting. Our conjecture says that when determining the integrality gap for the subtour LP, it suffices to look at instances that have an optimal solution that is an F2M, i.e., an extreme point of the fractional 2-matching polytope.

CONJECTURE 2. *Let α_n be the integrality gap of the subtour LP on n nodes. Then there exists an instance that has an optimal subtour LP solution that is an F2M and for which the optimal tour has cost α_n times the subtour LP cost.*

Let us call cost functions c such that there exists an optimal subtour LP solution that is an F2M *fractional 2-matching costs* for the subtour LP. We note that the family of instances that provide a lower bound of $\frac{4}{3}$ on the integrality gap is a family of fractional 2-matching costs.

We could make a similar conjecture for the ratio of the cost of the optimal 2-matching to the subtour LP, but by Corollary 1 and the example in Figure 1, we already know that the conjecture is true. However, its truth does not shed any light on the conjecture above.

Interestingly, we appear to know almost nothing about the consequences of Conjecture 2. Even for this very restricted set of cost functions, we do not know a better upper bound on the integrality gap of the subtour LP other than the bound of $\frac{3}{2}$. It would be very interesting to prove that for such costs the integrality gap is $\frac{4}{3}$. Boyd and Carr [6] have shown this for fractional 2-matching costs in which all the cycles of the F2M have size 3. Note that the F2M consists of a single component and will not have a cut edge since it is feasible for the subtour LP. Therefore, their result also follows from the technique of Theorem 1 since the resulting graphical 2-matching is Eulerian if all cycles have size three (the graphical 2-matching may not be connected if there are cycles of size five).

In a companion paper, Qian et al. [22] show that if an analogous conjecture for edge costs $c(i, j) \in \{1, 2\}$ is true, then the integrality gap for 1,2-TSP is at most $\frac{7}{6}$. They conjecture that the integrality gap for the 1,2-TSP is at most $\frac{10}{9}$; it is known that it can be no smaller than $\frac{10}{9}$. It would be nice to show that if the analogous conjecture is true, then the integrality gap for 1,2-TSP is at most $\frac{10}{9}$. In recent work, Mömke and Svensson [18] show that if an analogous conjecture holds for graph-TSP instances, then the integrality gap for graph-TSP is at most $\frac{4}{3}$.

Appendix. Polyhedral description of 2MO. We repeat Theorem 4 for the sake of completeness.

THEOREM 4. *Let $G = (V_{\mathrm{man}} \cup V_{\mathrm{opt}}, E)$ be a 2MO instance. The convex hull of integer 2MO solutions is given by the following polytope:*

$$\sum_{e \in \delta(i)} x(e) = 2, \quad \forall i \in V_{\mathrm{man}}, \tag{14}$$

$$\sum_{e \in \delta(i)} x(e) \leq 2, \quad \forall i \in V_{\mathrm{opt}}, \tag{15}$$

$$\sum_{e \in \delta(S) \backslash F} x(e) + \sum_{e \in F} (1 - x(e)) \geq 1, \quad \forall S \subseteq V, \ F \subseteq \delta(S), \ F \text{ matching}, \ |F| \text{ odd}, \tag{16}$$

$$0 \leq x(e) \leq 1, \quad \forall e \in E. \tag{17}$$

PROOF. The proof that we present here is similar to the proof of the polyhedral description of the 2-matching polytope (Theorem 30.8) in Schrijver [24]. We will first show that any 2MO solution is contained in the polytope and next show that the extreme points of the polytope coincide with the 2MO solutions.

Constraints (14), (15), and (17) obviously hold for a 2MO solution. To show that constraint (16) is satisfied, we consider two cases:

● *Case* 1 There is an $\bar{e} \in F$ with $x(\bar{e}) = 0$. This makes the left-hand side of constraint (16) at least 1 since $x(e) \geq 0$ for all e.

- *Case 2* $x(e) = 1$ for all $e \in F$. Since $|F|$ is odd, and each node is incident to an even number of edges in an 2MO solution, it follows that there has to be an edge in the solution in $\delta(S)$ that is not in F. So the constraint also holds in this case.

The polytope thus contains all 2MO solutions. We will now show that its extreme points coincide with 2MO solutions by reducing 2MO instances to matching instances for which perfect matchings correspond to 2MO solutions. We will show that any feasible point in the 2MO polytope corresponds to a feasible point in the perfect matching polytope. Because any point in the perfect matching polytope can be written as a convex combination of perfect matchings, this implies that any point in the 2MO polytope can be written as a convex combination of 2MO solutions, and therefore all extreme points of the 2MO polytope correspond to 2MO solutions.

Before we consider the reduction to perfect matchings, we will first show that adding constraint (16) for all $F \subseteq E$ of odd cardinality does not change the 2MO polytope. These additional constraints will be convenient when showing that a feasible point in the 2MO polytope is in the perfect matching polytope.

We prove this by induction on $|F|$. Consider $\bar{S}$ and $\bar{F} \subseteq \delta(\bar{S})$ so that $\bar{F}$ is not a matching; i.e., $|\bar{F} \cap \delta(i)| \geq 2$ for some $i \in V$. We consider three cases.

- *Case 1* $|\bar{F} \cap \delta(i)| \geq 3$. Then

$$\sum_{e \in \delta(\bar{S}) \setminus \bar{F}} x(e) + \sum_{e \in \bar{F}} (1 - x(e)) \geq \sum_{e \in \bar{F}} (1 - x(e))$$

$$\geq \sum_{e \in \bar{F} \cap \delta(i)} (1 - x(e))$$

$$\geq 3 - \sum_{e \in \bar{F} \cap \delta(i)} x(e)$$

$$\geq 3 - \sum_{e \in \delta(i)} x(e) \geq 3 - 2 \geq 1.$$

- *Case 2* $|\bar{F} \cap \delta(i)| = 2$ and $i \in \bar{S}$. Let $F' = \bar{F} \setminus \delta(i)$ and let $S' = \bar{S} \setminus \{i\}$. Then

$$\sum_{e \in \delta(\bar{S}) \setminus \bar{F}} x(e) + \sum_{e \in \bar{F}} (1 - x(e)) \geq \sum_{e \in \delta(S') \setminus F'} x(e) - \sum_{e \in \delta(i)} x(e) + \sum_{e \in \delta(i) \cap \bar{F}} x(e)$$

$$+ \sum_{e \in F'} (1 - x(e)) + \sum_{e \in \delta(i) \cap \bar{F}} (1 - x(e))$$

$$= \sum_{e \in \delta(S') \setminus F'} x(e) + \sum_{e \in F'} (1 - x(e)) - \sum_{e \in \delta(i)} x(e) + 2.$$

By induction and the degree bound for i, this quantity is at least 1.

- *Case 3* $|\bar{F} \cap \delta(i)| = 2$ and $i \notin \bar{S}$. Let $F' = \bar{F} \setminus \delta(i)$ as in the previous case but now let $S' = \bar{S} \cup \{i\}$. Then the same string of inequalities as in the previous case holds.

We now use the usual reduction from 2-matchings to matchings (see Theorem 30.7 in Schrijver [24], the notation of which we will also follow): for each node i in the 2MO, there will be two nodes in the matching instance, i' and i''. For each edge $e = (i, j)$ in the 2MO instance, there will be two nodes and five edges in the matching instance: nodes $p_{e,i}$ and $p_{e,j}$ and edges $(i', p_{e,i})$, $(i'', p_{e,i})$, $(p_{e,i}, p_{e,j})$, $(j', p_{e,j})$, and $(j'', p_{e,j})$. The only difference between the reduction from 2-matchings to matchings and the reduction from 2MO to matchings is that for optional nodes i we also add an edge between nodes i' and i''. An illustration of the reduction is given in Figure A.1, where the part of the matching instance is given that corresponds to an edge between a mandatory node i and an optional node j.

Given a (fractional) solution x to a 2MO instance, we define a solution y to the corresponding matching instance as follows:

$$y(i', p_{e,i}) = y(i'', p_{e,i}) = \frac{1}{2} x(e) \quad \text{and} \quad y(p_{e,i}, p_{e,j}) = 1 - x(e) \quad \text{for all } e = (i, j) \in E,$$

and

$$y(i', i'') = 1 - \frac{1}{2} \sum_{e \in \delta(i)} x(e) \quad \text{for all } i \in V_{\text{opt}}.$$
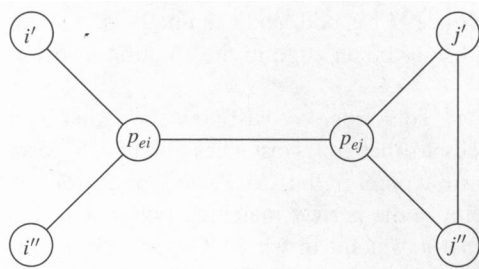
FIGURE A.1. Illustration of the reduction from 2MO to matchings. The part of the matching instance is drawn that corresponds to an edge between a mandatory node i and an optional node j.

We will now show that this solution is indeed in the perfect matching polytope given by the constraints (6)–(8) of the linear program (M) in §2 (where the variables are here called y instead of x). For nodes $p_{e,i}$, the degree bound constraints (6) follow directly from the definition of y (there are three edges incident to $p_{e,i}$ with y-values $\frac{1}{2}x(e)$, $\frac{1}{2}x(e)$, and $1 - x(e)$, which sum to 1). For the other nodes, constraint (6) follows directly from the degree bound constraints (14) or (15) in the 2MO instance and the definition of y. Constraints (8) follow directly from constraints (17).

We will now prove that constraints (7) also hold for all subsets of nodes of odd cardinality in our reduction. Let S' be such a subset. We consider four cases.

• *Case* 1 $|\{i', i''\} \cap S'| = 1$ for some $i \in V$. Note that we have edges $(i', p_{e,i})$ and $(i'', p_{e,i})$ in the reduction both of which have y-value $\frac{1}{2}x(e)$ and of which exactly one will be in $\delta(S')$. Furthermore, (i', i'') is in $\delta(S')$ if i is in V_{opt}. Therefore $\sum_{e' \in \delta(S')} y(e') \geq \sum_{e \in \delta(i)} \frac{1}{2}x(e) = 1$ if $i \in V_{\mathrm{man}}$ by the degree bound (14). Similarly $\sum_{e' \in \delta(S')} y(e') \geq \sum_{e \in \delta(i)} \frac{1}{2}x(e) + (1 - \frac{1}{2}\sum_{e \in \delta(i)} x(e)) = 1$ if $i \in V_{\mathrm{opt}}$ by the degree bound (15).

• *Case* 2 For some $e = (i, j) \in E$, $p_{e,i} \in S'$, $p_{e,j} \notin S'$, and $\{i', i''\} \cap S' = \varnothing$. Let $p = p_{e,i}$. Then $\sum_{e' \in \delta(S')} y(e') \geq y(p, i') + y(p, i'') + y(p, p_{e,j}) = \frac{1}{2}x(e) + \frac{1}{2}x(e) + 1 - x(e) = 1$.

• *Case* 3 For some $e = (i, j) \in E$, $p_{e,i} \in S'$, $p_{e,j} \notin S'$, and $\{j', j''\} \subseteq S'$. Let $p = p_{e,j}$. Then $\sum_{e' \in \delta(S')} y(e') \geq y(p, j') + y(p, j'') + y(p, p_{e,i}) = \frac{1}{2}x(e) + \frac{1}{2}x(e) + 1 - x(e) = 1$.

• *Case* 4 We may now assume that S' is such that $|\{i', i''\} \cap S'|$ is even for all $i \in V$ and that $\{i', i''\} \subseteq S'$ and $\{j', j''\} \cap S' = \varnothing$ if $p_{e,i} \in S'$ and $p_{e,j} \notin S'$ because otherwise we are in one of the previous cases. Define $\bar{S} = \{i \in V: i' \in S'$ and $i'' \in S'\}$ and $\bar{F} = \{e = \{i, j\} \in E: p_{e,i} \in S'$ and $p_{e,j} \notin S'\}$. Note that the previous argument implies that $\bar{F} \subseteq \delta(\bar{S})$.

Consider $e = (i, j) \in \delta(\bar{S})$ in the 2MO instance and assume without loss of generality that $i \in \bar{S}$. By the definition of $\bar{S}$, this means $\{j', j''\} \cap S' = \varnothing$. We consider $e \in \bar{F}$ and $e \notin \bar{F}$ separately. First of all, assume $e \in \bar{F}$. Since we are not in the previous cases, this means that $p_{e,i} \in S'$ and $p_{e,j} \notin S'$. So for each such e in the 2MO instance, we have $(p_{e,i}, p_{e,j}) \in \delta(S')$ in the matching instance, with a y-value of $1 - x(e)$. Second, assume $e \notin \bar{F}$. By the definition of $\bar{F}$, we know that either $p_{e,i}$ and $p_{e,j}$ are both in S' or both not in S'. So for each such e in the 2MO instance, we have either $\{(i', p_{e,i}), (i'', p_{e,i})\} \subseteq \delta(S')$ or $\{(j', p_{e,j}), (j'', p_{e,j})\} \subseteq \delta(S')$ in the matching instance, each of which carries a total y-value of $x(e)$.

We thus get $\sum_{e' \in \delta(S')} y(e') \geq \sum_{e \in \delta(\bar{S}) \setminus \bar{F}} x(e) + \sum_{e \in \bar{F}} (1 - x(e))$. We then note that the parity of $|\bar{F}|$ is equal to the parity of the number of nodes of the type $p_{e,i}$ in S', which implies that the parity of $|\bar{F}|$ and $|S'|$ are the same because the other nodes in S' appear in pairs. Thus since $|S'|$ is odd, $|\bar{F}|$ is odd, and we have $\sum_{e' \in \delta(S')} y(e') \geq \sum_{e \in \delta(\bar{S}) \setminus \bar{F}} x(e) + \sum_{e \in \bar{F}} (1 - x(e)) \geq 1$ by the feasibility of x for constraints (16).

We conclude the proof by noting that a perfect matching in the constructed instance corresponds to the 2MO solution consisting of all edges $e = (i, j)$ for which $(p_{e,i}, p_{e,j})$ is *not* in the perfect matching solution. $\square$

References

[1] Aggarwal N, Garg N, Gupta S (2011) A 4/3-approximation for TSP on cubic 3-edge-connected graphs. Available at http://arxiv.org/abs/1101.5586.

[2] Aráoz J, Cunningham WH, Edmonds J, Green-Krótki J (1983) Reductions to 1-matching polyhedra. *Networks* 13(4):455–473.

[3] Balinski ML (1965) Integer programming: Methods, uses, computation. *Management Sci.* 12(3):253–313.

[4] Benoit G, Boyd S (2008) Finding the exact integrality gap for small traveling salesman problems. *Math. Oper. Res.* 33(4):921–931.

[5] Boyd S, Carr R (1999) A new bound for the ratio between the 2-matching problem and its linear programming relaxation. *Math. Programming* 86:499–514.

[6] Boyd S, Carr R (2011) Finding low cost TSP and 2-matching solutions using certain half-integer subtour vertices. *Discrete Optim.* 8:525–539. Prior version available at http://www.site.uottawa.ca/~sylvia/recentpapers/halftri.pdf. Accessed June 27, 2011.

[7] Boyd S, Elliott-Magwood P (2010) Structure of the extreme points of the subtour elimination polytope of the STSP. Iwata S, ed. *Combinatorial Optimization and Discrete Algorithms, RIMS Kôkyûroku Bessatsu*, Vol. B23 (Research Institute for Mathematical Sciences, Kyoto University, Kyoto, Japan), 33–47.

[8] Boyd S, Sitters R, van der Ster S, Stougie L (2011) TSP on cubic and subcubic graphs. Günlük O, Woeginger GJ, eds. *Integer Programming Combin. Optim., 15th Internat. Conf., IPCO 2011.* 6655 in Lecture Notes in Computer Science (Springer, Berlin), 65–77.

[9] Christofides N (1976) Worst case analysis of a new heuristic for the traveling salesman problem. Report 388, Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, PA.

[10] Dantzig G, Fulkerson R, Johnson S (1954) Solution of a large-scale traveling-salesman problem. *Oper. Res.* 2:393–410.

[11] Edmonds J (1965) Maximum matching and a polyhedron with (0, 1) vertices. *J. Res. Nat. Bur. Standards Sect. B* 69B:125–130.

[12] Edmonds J (1967) Optimum branchings. *J. Res. National Bureau of Standards B* 71B:233–240.

[13] Gamarnik D, Lewenstein M, Sviridenko M (2005) An improved upper bound for the TSP in cubic 3-edge-connected graphs. *Oper. Res. Lett.* 33(5):467–474.

[14] Goemans MX (1995) Worst-case comparison of valid inequalities for the TSP. *Math. Programming* 69:335–349.

[15] Goemans MX, Bertsimas DJ (1990) Survivable networks, linear programming relaxations, and the parsimonious property. *Math. Programming* 60:145–166.

[16] Johnson DS, McGeoch LA (2002) Experimental analysis of heuristics for the STSP. Gutin G, Punnen AP, eds. *The Traveling Salesman Problem and Its Variants* (Kluwer Academic Publishers, Dordrect, The Netherlands), 369–444.

[17] Mömke T, Svensson O (2011) Approximating graphic TSP by matchings. *Proc. 52th Annual Sympos. Foundations Comput. Sci., Palm Springs, CA*, 560–569.

[18] Mömke T, Svensson O (2012) Personal communication.

[19] Mucha M (2012) 13/9-approximation for graphic TSP. *29th Internat. Sympos. Theoret. Aspects Comput. Sci., STACS 2012, LIPIcs*, Vol. 14 (Dagstuhl Publishing, Dagstuhl, Germany), 30–41.

[20] Naddef D, Pulleyblank WR (1981) Matchings in regular graphs. *Discrete Math.* 34:283–291.

[21] Oveis Gharan S, Saberi A, Singh M (2011) A randomized rounding approach to the traveling salesman problem. *Proc. 52th Annual Sympos. Foundations Comput. Sci., Palm Springs, CA*, 550–559.

[22] Qian J, Schalekamp F, Williamson DP, van Zuylen A (2012) On the integrality gap of the subtour LP for the 1, 2-TSP. *LATIN 2012: Theoret. Informatics, 10th Latin Amer. Sympos.*, Lecture Notes in Computer Science, Vol. 7256 (Springer, Berlin), 606–617.

[23] Schalekamp F, Williamson DP, van Zuylen A (2012) A proof of the Boyd-Carr conjecture. *Proc. 23nd ACM-SIAM Sympos. Discrete Algorithms, Kyoto, Japan*, 1477–1486.

[24] Schrijver A (2003) *Combinatorial Optimization—Polyhedra and Efficiency* (Springer, Berlin).

[25] Sebő A, Vygen O (2012) Shorter tours by nicer ears: 7/5-approximation for graphic TSP, 3/2 for the path version, and 4/3 for two-edge-connected subgraphs, http://arxiv.org/abs/1201.1870.

[26] Shmoys DB, Williamson DP (1990) Analyzing the Held-Karp TSP bound: A monotonicity property with application. *Inform. Processing Lett.* 35:281–285.

[27] Williamson DP (1990) Analysis of the Held-Karp heuristic for the traveling salesman problem. Master's thesis, MIT, Cambridge, MA.

[28] Wolsey LA (1980) Heuristic analysis, linear programming and branch and bound. *Math. Programming Study* 13:121–134.