Integer Models for the Asymmetric Traveling Salesman Problem with Pickup and Delivery

Ryan J. O'Neil · Karla Hoffman

October 21, 2018

Abstract We propose a new Mixed Integer Programming formulation for the Asymmetric Traveling Salesman Problem with Pickup and Delivery, along with valid inequalities for the Sarin-Sherali-Bhootra formulation. We study these models in their complete forms, relax complicating constraints of these models, and compare their performance. Finally, we present computational results showing the promise of these formulations when applied to pickup and delivery problems.

Keywords asymmetric traveling salesman problem; pickup and delivery; precedence constraints; subtour elimination constraints; integer programming; real-time optimization

1 Introduction

The Traveling Salesman Problem (TSP) is the search for a minimum cost Hamiltonian circuit connecting a set of locations. In general, the TSP is NP-complete (Papadimitriou, 1977). The first paper of note related to TSP research solved an instance of "49 cities, one in each of the 48 states and Washington, D.C." (Dantzig et al., 1954). Algorithms now exist that, given sufficient time, can solve TSPs with tens of thousands of nodes, and its computational boundaries continue to be pushed forward (Applegate et al., 2011).

The TSP is a well-studied problem that has inspired substantial theoretical developments impacting much of combinatorial optimization. It also provides a

Ryan J. O'Neil
Systems Engineering & Operations Research Department
George Mason University, Fairfax, Virginia 22030
E-mail: roneil1@gmu.edu
Decision Engineering Department
Grubhub, Chicago, Illinois 60602
E-mail: roneil@grubhub.com

Karla Hoffman
Systems Engineering & Operations Research Department
George Mason University, Fairfax, Virginia 22030
E-mail: khoffman@gmu.edu

practical application in its own right, particularly in the context of Vehicle Routing Problems (VRP). VRPs are common in industry and involve routing a set of vehicles to visit a set of nodes at minimum cost. At its surface, the only substantive difference between the TSP and the VRP is the latter's use of multiple vehicles to service customer demand.

The Pickup and Delivery Problem (PDP) is an increasingly prevalent industrial form of the VRP in which each of a set of requests must be picked up from one or more locations and delivered to one or more delivery points. Pickups must precede their associated deliveries in any feasible route, and each pair must be serviced by the same vehicle. Pickup and delivery locations can be distinct to each request, as in the many-to-many Dial-A-Ride Problem (DARP) of Psaraftis (1980) and Bertsimas et al. (2018), and the Meal Delivery Routing Problem (MDRP) of Reyes et al. (2018). Given multiple pickup and delivery requests, the PDP seeks to route a set of vehicles to service those requests at minimum cost (Ruland and Rodin, 1997).

Solving a PDP with a single vehicle is equivalent to solving a TSP with precedence constraints on the pickup and delivery nodes. This form is referred to as the Traveling Salesman Problem with Pickup and Delivery (TSPPD) (Dumitrescu et al., 2010). There is a great deal of literature on the TSP and some of its variants, such as the TSP with Time Windows (TSPTW) (Applegate et al., 2011). In contrast, there has been less attention paid to the TSPPD, despite its practical applicability (Ruland and Rodin, 1997). TSPPDs and their variants play an increasingly important role in industrial routing applications. This importance is witnessed by a proliferation of ride hailing and sharing companies, as well as on-demand delivery service providers for everything from groceries, alcoholic beverages, and meals, to snacks and convenience store items.

The divide between research and industrial use of the TSP becomes clearly evident when one considers practical limitations associated with routing. For example, delivery trucks are often limited to routes of fewer than 100 stops due to physical considerations, such as vehicle capacity (Caseau and Laburthe, 1997). Routes used for high volume restaurant delivery are even shorter due to the perishability of goods, and typically involve fewer than 10 stops.

Even in the context of the broader PDP with many vehicles and many more nodes (e.g. hundreds or thousands), the allowable individual routes tend to be relatively short. Solving real-world delivery problems often requires solving, in a very short time frame (i.e. fractions of a second to not more than minutes), a huge numbers of problems each having a very small number of nodes. And, often, re-solving some subset of these problems as the delivery service learns of changing demand and/or service times.

In a prior study, we compared a variety of modeling and algorithmic approaches to solving TSPPDs within strict time budgets (O'Neil and Hoffman, 2018). While the majority of the models we considered support asymmetric arc costs, the most effective model is a symmetric Mixed Integer Programming (MIP) model based on that of Ruland and Rodin (1997) (RR) with a warm start. The asymmetric Sarin et al. (2005) MIP model is better at finding feasible solutions without a warm start, but is less effective at finding high quality solutions (e.g. solutions within 10% or 5% of optimal) and at proving optimality.

This is dissatisfying both intuitively and from the perspective of practical implementation. The TSPPD is fundamentally an asymmetric problem: pickups must precede their respective deliveries. Further, it is inconvenient to be limited to symmetric edge costs in a problem formulation. Depending on one's problem data, it may be beneficial to individually penalize arcs connecting the same points in different directions.

To that end, we implement both symmetric and asymmetric models for the TSPPD. We begin with two formulations from the literature, one of which is symmetric and the other being asymmetric. We test these models using symmetric problem instances generated from real-world meal delivery data, built from actual pickup and delivery locations observed at Grubhub, along with expected symmetric travel times connecting location pairs. We then attempt to improve the performance of the asymmetric models to be competitive with that of the symmetric one.

Contributions of this paper include new valid inequalities for the Sarin-Sherali-Bhootra (SSB) model that are specific to the asymmetric TSPPD (ATSPPD). We also propose a new O'Neil-Hoffman (OH) MIP ATSPPD model which, similar to the SSB model, uses a secondary set of variables to handle precedence relationships and subtour elimination. Model OH ascribes more meaning to this second set of variables than SSB, and uses them to control how pairs of pickups and deliveries interact in terms of precedence. Finally, we relax the precedence and Subtour Elimination Constraints (SEC) for both the SSB and OH models, adding them lazily as they are violated, to better compare them to the RR model. Our test set and source code for all model implementations are available for continued experimentation (Grubhub, 2018; O'Neil, 2018). Asymmetric models are tested against symmetric models using symmetric data in order to compare them. Once that comparison is accomplished, we perturb the symmetric instances into asymmetric instances to show that the models exhibit similar behavior when using data more representative of real-world situations.

2 Background

The Traveling Salesman Problem with Pickup and Delivery (TSPPD) is a modification of the Traveling Salesman Problem (TSP) that includes side constraints enforcing precedence among pickup and delivery node pairs. Each of n requests has a pickup node and a delivery node, and its pickup must occur before its delivery for a route to be feasible. The objective of the problem is to minimize total distance traveled while visiting each node exactly once. The TSPPD is formally described in Ruland and Rodin (1997) and Dumitrescu et al. (2010).

The TSPPD is defined on an ordered set of pickup nodes $V_+ = \{+1, \ldots, +n\}$ and associated delivery nodes $V_- = \{-1, \ldots, -n\}$ such that $(+i, -i)$ form a request and $+i$ must precede $-i$ in a feasible route. V is defined as the union of V_+ and V_- with the addition of origin and destination nodes $\{+0, -0\}$. $E_\pm$ is the set of edges connecting $V_+ \cup V_-$. E is the union of $E_\pm$ with all feasible edges connecting to the origin and destination nodes. The graph $G = (V, E)$ includes all nodes and edges required to describe the TSPPD.

$$V = \{+0, -0\} \cup V_+ \cup V_- \tag{1}$$

$$E = \{(+0, -0)\} \cup \{(+0, +i) \mid i \in V_+\} \cup \{(-0, -i) \mid i \in V_-\} \cup E_\pm \tag{2}$$

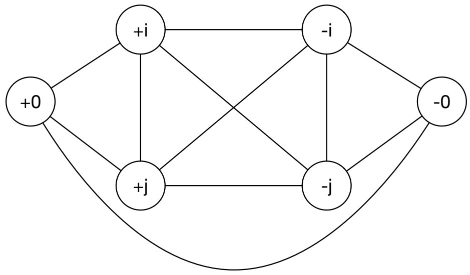

Fig. 1 Example TSPPD graph structure.

For modeling convenience, the edge $(+0, -0)$ must be included in any feasible solution. The set of edges E is defined such that it includes only feasible edges. That is, it is not possible that a TSPPD route begins with a delivery or ends with a pickup. Figure 1 shows an example of the feasible edges for a TSPPD with two requests. Note that it is possible to use the same formulation to solve Hamiltonian paths with precedence rather than circuits by simply assigning a cost of zero to the $(+0, -0)$ edge.

2.1 Symmetric Models

In the case of the symmetric TSPPD (STSPPD), each edge $(i, j) \in E$ is the same as the edge (j, i) and has the same costs and edge variables. c_{ij} specifies a nonnegative cost for each edge $(i, j) \in E$. By convention, $c_{+0,-0} = 0$. $x_{ij} \in \{0, 1\}$ is a binary decision variable for each $(i, j) \in E$ with the value $x_{ij} = 1$ if the edge (i, j) is in a solution and 0 otherwise. $\delta(S) = \{(i, j) \in E \mid i \in S, j \notin S\}$ is the cutset containing edges that connect $S \subset V$ and $\bar{S} \subset V$. For any node $i \in V$, $\delta(i) = \delta(\{i\})$. The STSPPD is defined using Formulation 1, as in Ruland and Rodin (1997).

$$\text{minimize} \quad \sum_{(i,j)\in E} c_{ij} x_{ij} \tag{3}$$

$$\text{subject to} \quad x_{+0,-0} = 1 \tag{4}$$

$$x(\delta(i)) = 2 \quad \forall \quad i \in V \tag{5}$$

$$x(\delta(S)) \geq 2 \quad \forall \quad S \subset V \tag{6}$$

$$x(\delta(S)) \geq 4 \quad \forall \quad S \subset V,\ \{+0, -i\} \subset S, \{-0, +i\} \subset V \setminus S \tag{7}$$

$$x_{ij} \in \{0, 1\} \quad \forall \quad (i, j) \in A \tag{8}$$

Formulation 1: STSPPD as provided by Ruland and Rodin (1997)

Constraint (4) requires that the edge connecting the origin and destination nodes be part of any feasible solution. The degree constraints (5) require that each node is entered and exited in all feasible routes, but by itself leaves open the possibility of disconnected subtours. Constraints (6) accomplish subtour elimination, forming a complete representation of the TSP. The final set of constraints

(7) require that pickups occur in routes before their respective deliveries (Ruland and Rodin, 1997).

2.2 Asymmetric Models

The asymmetric TSP (ASTP) polytope can be similarly adapted to the ATSPPD by associating the x variables with arcs (i.e. directed edges) and replacing the two-degree constraints with assignment constraints. Instead of an edge set E we define an arc set A which contains only the feasible arcs for the ATSPPD.

$$\begin{aligned} A = &\{(-0, +0)\} \\ &\cup \{(+0, +i) \mid +i \in V_+\} \\ &\cup \{(-i, -0) \mid -i \in V_-\} \\ &\cup \{(+i, j) \mid +i \in V_+,\ j \in (V_+ \cup V_-) \setminus \{+i\}\} \\ &\cup \{(-i, j) \mid -i \in V_+,\ j \in (V_+ \cup V_-) \setminus \{+i, -i\}\} \end{aligned} \tag{9}$$

Formulation 2 is more general since it supports arc costs that are not the same bidirectionally. Further, it is intuitively satisfying to consider the TSPPD this way, as there is a natural asymmetry built into the structure of the problem: pickups must precede their associated deliveries. As in Formulation 1, the $x_{-0,+0}$ arc connecting the start and end nodes must be part of any feasible tour, $+0$ must connect to a pickup, and -0 must be preceded by a delivery. One small additional difference is that Formulation 2 does not include x variables for arcs starting at a delivery and ending at its associated pickup.

$$\begin{aligned}
\text{minimize} \quad & \sum_{(i,j)\in A} c_{ij} x_{ij} && && (10)\\
\text{subject to} \quad & \sum_{(i,j)\in A} x_{ij} = 1 && \forall \quad i \in V && (11)\\
& \sum_{(i,j)\in A} x_{ij} = 1 && \forall \quad j \in V && (12)\\
& x(\delta(S)) \geq 1 && \forall \quad S \subset V && (13)\\
& x(\delta(S)) \geq 4 && \forall \quad S \subset V,\ \{+0, -i\} \subset S, \{-0, +i\} \subset V \setminus S && (14)\\
& x_{ij} \in \{0, 1\} && \forall \quad (i, j) \in A && (15)
\end{aligned}$$

Formulation 2: ATSPPD

In Formulation 2, constraints (11) and (12) require that each node directly precede and follow exactly one other node. Constraints (13) accomplish subtour elimination, while precedence is enforced using the same constraints as in Formulation 1.

While formulations 1 and 2 give complete representations of the STSPPD and ATSPPD polytopes, the number of subtour elimination and precedence constraints grow quickly as a function of the number of nodes. Most exact approaches to the

TSP and its variants solve a relaxation of the problem that omits these constraints, and add constraints to the representation as they are violated by new solutions.

For these models we take the approach of solving a combinatorial relaxation containing only the degree constraints (5) for the STSPPD, or constraint sets (11) and (12) for the ATSPPD. When a candidate solution is found, we check to see if it contains subtours or violates precedence. If so, we add SEC and precedence constraints as "lazy" constraints within the solver. Once a tour covering all nodes and satisfying presence constraints is found, that tour is optimal.

As integer-feasible solutions are discovered in the branch-and-bound tree, they are scanned for subtours. $\mathcal{T}$ is the set containing all sets of nodes in a tour in the current solution. If $\mathcal{T} > 1$, then for each set of nodes corresponding to a subtour $S \in \mathcal{T}$, we add a lazy constraint of either the form (6) or (13), eliminating that subtour from future solutions.

Constraints (6) or (13) remove a given subtour from the final solution. Thereafter, not all edges in the subtour can exist in any new solution. An alternative and equivalent formulation of the constraint is shown as (16) below. We call constraints (6) and (13) the "cutset" form and constraint (16) the "subtour" form, respectively.

$$\sum_{i,j \in S} x_{ij} \leq |S| - 1 \tag{16}$$

We integrate SEC into the TSPPD by first eliminating subtours in Gurobi's callback. Once there is only one tour, that is, $|\mathcal{T}| = 1$, the current solution is a valid TSP solution. We then scan it for precedence violations, adding lazy cuts using constraint (7) as precedence violations are found.

2.3 Polynomial Length Models

Not all TSP formulations require relaxation. A number of models contain constraints that imply the Dantzig-Fulkerson-Johnson subtour elimination constraints. Perhaps most notably, the Miller-Tucker-Zemlin formulation use variables to indicate which position each node is in a feasible tour (Miller et al., 1960).

Sarin et al. (2005) introduce a polynomial-length ATSP model based on an assignment problem formulation. Their model uses x variables to represent directed arcs that indicate direct precedence and y variables to indicate route precedence. If $x_{ij} = 1$ then i directly precedes j, and if $y_{ij} = 1$ then i precedes j but need not directly precede it. This model is given in Formulation 3.

$$
\begin{aligned}
\text{minimize} \quad & \sum_{(i,j)\in A} c_{ij}x_{ij} & & & (17)\\
\text{subject to} \quad & \sum_{(i,j)\in A} x_{ij} = 1 & \forall \quad & i \in V & (18)\\
& \sum_{(i,j)\in A} x_{ij} = 1 & \forall \quad & j \in V & (19)\\
& x_{ij} \leq y_{ij} & \forall \quad & i,j \in V & (20)\\
& y_{ij} + y_{ji} = 1 & \forall \quad & i,j \in V, i \neq j & (21)\\
& y_{ij} + x_{ji} + y_{jk} + y_{ki} \leq 2 & \forall \quad & i,j,k \in V, i \neq j \neq k & (22)\\
& x_{ij} \in \{0,1\} & \forall \quad & i,j \in V, i \neq j & (23)\\
& 0 \leq y_{ij} \leq 1 & \forall \quad & i,j \in V, i \neq j & (24)
\end{aligned}
$$

Formulation 3: The ATSP Formulation of Sarin et al. (2005)

Constraints (20) establish the relationship between route and direct precedences. Constraints (21) require that, for each pair of nodes, one must precede the other. Subtour elimination is handled through route precedence using constraints (22). The ATSP model of Sarin et al. (2005) conveniently adapts to the ATSPPD by setting $y_{+0,i} = y_{i,-0} = 1$ for all $i \in V \setminus \{+0, -0\}$, and all $y_{+i,-i} = 1$.

3 Proposed Inequalities

The SSB ATSP model given in Formulation 3 is easily adapted to the ATSPPD by setting lower bounds directly on the y variables. In this section, we present additional valid inequalities for model SSB that apply to the ATSPPD. Constraint sets (25, 26) are collectively referred to as constraint set `A`, while constraint sets (27, 28, 29, 30, 31) are referred to as constraint set `B`. We follow the notation that $i \prec j$ means i precedes j in a route, though perhaps not directly.

With respect to constraint set (25), only one of $x_{+j,-i}$ and $x_{-i,+j}$ may be on, otherwise these arcs will form a subtour. Both $x_{+j,-i} = 1$ and $x_{-i,+j} = 1$ independently imply that $+i \prec +j$, thus the constraints are valid. Constraint set (26) follows from the fact that if $+j \prec +i$ then $+j \prec -i$ as well. If $-i \prec +j$ then $+i \prec +j$. Thus $y_{-i,+j} + y_{+j,+i} \leq 1$. If $x_{+j,-i} = 0$ then it is simply removed from the inequality, while if $x_{+j,-i} = 1$ then both $y_{-i,+j}$ and $y_{+j,+i}$ must be off.

$$
\begin{aligned}
x_{+j,-i} + x_{-i,+j} \leq y_{+i,+j} & \quad \forall \quad +i,+j \in V_+, i \neq j & (25)\\
x_{+j,-i} + y_{-i,+j} + y_{+j,+i} \leq 1 & \quad \forall \quad +i,+j \in V_+, i \neq j & (26)
\end{aligned}
$$

A second set of valid inequalities are inferred directly from the route precedence variables. For constraint set (27), if $+i \prec +j$ then $+i \prec -j$ since $+j \prec -j$. If $+i \prec -j$ then nothing additional is implied about the precedence relationships of $-i$ or $+j$. Constraint sets (28, 29, 30) are valid since $-i \prec +j$ implies that $+i \prec +j$ and $-i \prec -j$, and therefore also that $+i \prec -j$. Finally, constraint set (31) is valid since $-i \prec -j$ implies that $+i \prec -j$ as well. It is particularly interesting to note that, if $-i \prec +j$, then we can make three additional inferences on related y variables.

$$y_{+i,+j} \leq y_{+i,-j} \quad \forall \quad +i,+j \in V_+, i \neq j \tag{27}$$
$$y_{-i,+j} \leq y_{+i,+j} \quad \forall \quad +i,+j \in V_+, i \neq j \tag{28}$$
$$y_{-i,+j} \leq y_{-i,-j} \quad \forall \quad +i,+j \in V_+, i \neq j \tag{29}$$
$$y_{-i,+j} \leq y_{+i,-j} \quad \forall \quad +i,+j \in V_+, i \neq j \tag{30}$$
$$y_{-i,-j} \leq y_{+i,-j} \quad \forall \quad +i,+j \in V_+, i \neq j \tag{31}$$

4 Proposed Model

The SSB model uses two sets of variables: x_{ij} controls direct precedence from i to j and y_{ij} manages route precedence. In this section we propose an alternative MIP model for the ATSPPD in which there are also two sets of variables. In our model, x has the same function. Instead of y variables that control route precedence, we introduce a set of variables, w, that correspond to the allowable configurations of pairwise pickup and delivery node pairs. That is, for each pair of pickups and deliveries $\{(+i,-i),(+j,-j)\}$, there is a set of w_{ijm} binary variables. There are six allowable precedence configurations, each of which sets a w variable to one if it occurs in a solution.

$$w_{ij1} = 1 \implies +i \prec +j \prec -i \prec -j \quad \forall \quad i,j \in V_+, i < j \tag{32}$$
$$w_{ij2} = 1 \implies +i \prec +j \prec -j \prec -i \quad \forall \quad i,j \in V_+, i < j \tag{33}$$
$$w_{ij3} = 1 \implies +i \prec -i \prec +j \prec -j \quad \forall \quad i,j \in V_+, i < j \tag{34}$$

$$w_{ji1} = 1 \implies +j \prec +i \prec -j \prec -i \quad \forall \quad i,j \in V_+, i < j \tag{35}$$
$$w_{ji2} = 1 \implies +j \prec +i \prec -i \prec -j \quad \forall \quad i,j \in V_+, i < j \tag{36}$$
$$w_{ji3} = 1 \implies +j \prec -j \prec +i \prec -i \quad \forall \quad i,j \in V_+, i < j \tag{37}$$

For each set of w variables, exactly one configuration must be true in any feasible route, as enforced by constraint set (38). Since the w variables imply precedence relationships among the nodes, and we will use them for SEC below, there is no need to add explicit precedence constraints.

$$\sum_{m=1}^{3} (w_{ijm} + w_{jim}) = 1 \quad \forall \quad i,j \in V_+, i < j \tag{38}$$

The x and w variables are linked by allowing arcs to be turned on in the former only when allowed by configurations implied by the latter. This gives us constraint sets (39, 40, 41, 42, 43).

$$
\begin{aligned}
&x_{+i,-i} \leq w_{ij3} + w_{ji2} + w_{ji3} && \forall \quad +i \in V_+ && (39)\\
&x_{+i,+j} \leq w_{ij1} + w_{ij2} && \forall \quad +i, +j \in V_+, i \neq j && (40)\\
&x_{+i,-j} \leq w_{ji1} && \forall \quad +i \in V_+, -j \in V_-, i \neq j && (41)\\
&x_{-i,+j} \leq w_{ij3} && \forall \quad -i \in V_-, +j \in V_+, i \neq j && (42)\\
&x_{-i,-j} \leq w_{ij1} + w_{ji2} && \forall \quad -i \in V_-, -j \in V_-, i \neq j && (43)
\end{aligned}
$$

We now add SEC using the w variables in the same way the SSB model does. Define a function $s(i, j)$ which computes a summation of all w variables in which $i \prec j$, as shown in (44).

$$
\begin{aligned}
s(+i, +j) &= w_{ij1} + w_{ij2} + w_{ij3}\\
s(+i, -j) &= w_{ij1} + w_{ij2} + w_{ij3} + w_{ji1} + w_{ji2}\\
s(-i, +j) &= w_{ij3}\\
s(-i, -j) &= w_{ij1} + w_{ij3} + w_{ji2}
\end{aligned} \tag{44}
$$

Constraint set (45) employs $s(i, j)$ to add SEC to the model.

$$
\begin{aligned}
s(p, q) + s(q, r) + s(r, p) \leq 2 \ \forall \ & i, j, k \in V_+, i < j < k,\\
& p \in \{+i, -i\}, q \in \{+j, -j\}, r \in \{+k, -k\}
\end{aligned} \tag{45}
$$

These decision variables and constraints combine into Formulation 4.

$$
\begin{aligned}
\text{minimize} \quad & \sum_{(i,j)\in A} c_{ij} x_{ij} && && (46)\\
\text{subject to} \quad & \sum_{(i,j)\in A} x_{ij} = 1 && \forall \quad i \in V && (47)\\
& \sum_{(i,j)\in A} x_{ij} = 1 && \forall \quad j \in V && (48)\\
& x_{+i,-i} \leq w_{ij3} + w_{ji2} + w_{ji3} && \forall \quad +i \in V_+ && (49)\\
& x_{+i,+j} \leq w_{ij1} + w_{ij2} && \forall \quad +i, +j \in V_+, i \neq j && (50)\\
& x_{+i,-j} \leq w_{ji1} && \forall \quad +i \in V_+, -j \in V_-, i \neq j && (51)\\
& x_{-i,+j} \leq w_{ij3} && \forall \quad -i \in V_-, +j \in V_+, i \neq j && (52)\\
& x_{-i,-j} \leq w_{ij1} + w_{ji2} && \forall \quad -i \in V_-, -j \in V_-, i \neq j && (53)\\
& \sum_{m=1}^{3} (w_{ijm} + w_{jim}) = 1 && \forall \quad i, j \in V_+, i < j && (54)\\
& s(p, q) + s(q, r) + s(r, p) \leq 2 && \forall \quad i, j, k \in V_+, i < j < k, && (55)\\
& && \quad p \in \{+i, -i\}, &&\\
& && \quad q \in \{+j, -j\}, &&\\
& && \quad r \in \{+k, -k\} &&\\
& x_{ij} \in \{0, 1\} && \forall \quad i, j \in V, i \neq j && (56)\\
& 0 \leq w_{ijm} \leq 1 && \forall \quad i, j \in V, i \neq j, m \in \{1, 2, 3\} && (57)
\end{aligned}
$$

Formulation 4: Proposed ATSPPD MIP Model

5 Relaxed Models

The ATSPPD forms of the SSB model (Formulation 3) and the OH model (Formulation 4) both represent the ATSPPD completely and do not require additional constraints to eliminate subtours or enforce precedence relationships. However, as we shall see in section 6, these polynomial length models can be less effective at optimization than relaxed equivalents. We are looking for ATSPPD MIP formulations that are as effective in solving symmetric problems as the RR model (Formulation 1), and which have the additional benefit of handling asymmetric arc costs.

To that end, we also consider relaxed versions of the SSB and OH models in which the SEC on the y or w variables, respectively, are removed from the formulation. SEC and precedence constraints are then added lazily as they are violated within the branch-and-bound tree.

Both the SSB and OH models have the same set of x variables as the ATSPPD Formulation 2. One option for removing subtours and enforcing precedence is to use the same constraints on the x variables. This means adding constraints of the "subtour" (16) or "cutset" (13) form for SEC, and constraints of the form (14) for enforcing precedence. For the SSB model, precedence can also be enforced using the y variables by adding constraints of the form (22). All of these options are considered in section 6.

6 Results

We test various forms of TSPPD and ATSPPD models using instances constructed from pickup and delivery locations observed at Grubhub, along with expected travel times connecting location pairs. The test set has 10 instances per problem size. This allows us to gauge the performance of the models on realistic problems from meal delivery, in which pickup locations are likely to be clustered close together, and there may be many near optimal solutions that would have to be considered to prove optimality.

In order to easily compare asymmetric to symmetric models, we first use the symmetric Grubhub instances. One we determine the best configurations, we then perturb these test instances so they are asymmetric, showing that similar performance may be expected from our models in the presence of asymmetric arc costs. We accomplish this by multiplying the upper triangular portion of the cost matrix by a uniform random variable between 0.7 and 1.3. For each symmetric instance of the problem set, we generate a single asymmetric instance for testing.

Table 1 describes the various models and configurations.

We evaluate the models according to multiple criteria: the time taken to find a first feasible TSPPD route, the time to find a route within 10% of the true optimum, the time to find a route within 5% of the true optimum, the time to find an optimal route, and the time to prove optimality. The first four measures are important to real-time logistics, where high quality routes must be found quickly and there may not be enough time to optimize fully. The final measure quantifies the capacity of model formulations and algorithms to optimize. We use time to prove optimality as a proxy for general algorithmic performance while selecting configurations.

				Constraint Sets		
					Lazy Constraints	
Type	Model	Formulation	SEC	Relaxed	SEC	Precedence
Sym.	RR	Form. 1	Cutset	(6), (7)	(6)	(7)
			Subtour	(6), (7)	(16)	(7)
Asym.	AP	Form. 2	Cutset	(13), (14)	(13)	(14)
			Subtour	(13), (14)	(16)	(14)
	OH	Form. 4	-	-	-	-
	OH/X		Cutset	(45)	(13)	(14)
			Subtour	(45)	(16)	(14)
	SSB	Form. 3	-	-	-	-
	SSB+A	Form. 3 + (25-26)	-	-	-	
	SSB+A/X		Cutset	(22)	(13)	(14)
			Subtour	(22)	(16)	(14)
	SSB+A/Y		Cutset	(22)	(13)	(22)
			Subtour	(22)	(16)	(22)
	SSB+B	Form. 3 + (27-31)	-	-	-	-
	SSB+B/X		Cutset	(22)	(13)	(14)
			Subtour	(22)	(16)	(14)
	SSB+B/Y		Cutset	(22)	(13)	(22)
			Subtour	(22)	(16)	(22)
	SSB+All	Form. 3 + (25-31)	-	-	-	-
	SSB+All/X		Cutset	(22)	(13)	(14)
			Subtour	(22)	(16)	(14)
	SSB+All/Y		Cutset	(22)	(13)	(22)
			Subtour	(22)	(16)	(22)

Table 1 Model formulations and constraint sets tested.

Charts report the median and maximum of execution times for each problem size and model configuration with the vertical axis scaled to log time. These measures are useful in choosing models for production systems because they give us a sense of typical performance, poor performance, and the spread of performance in terms of execution time. Given a target problem size and time budget, we can use the results of these tests to determine the best approach for a given TSPPD problem size. Execution times are limited to 1000 seconds. Any model configuration that is not able to achieve a particular goal (e.g. finding a feasible solution, or finding a solution with 10% of optimal) for all instances of a given size is removed from consideration for that problem size.

We generate results using Gurobi 8.0.0 as the MIP solver. Model code is written using C++14. The test machine is a Lenovo X1 Carbon with a 4-core Intel Core i5 CPU and 16 GB of RAM. We test using a single thread, since such executions of are deterministic and thus easier to interpret and compare.

Section 6.1 compares the complete SSB and OH formulations to the relaxed RR and AP formulations, where AP is the asymmetric equivalent of the RR model given in Formulation 2. Section 6.2 adds valid inequalities to the complete SSB. Section 6.3 relaxes SEC associated with the y variables in the SSB model and examines the impact of adding different lazy SEC on x and y variables and adding precedence constraints, as well as the proposed valid inequalities. Section 6.4 relaxes SEC in the OH model. Finally, sections 6.5 and 6.6 provide comprehensive

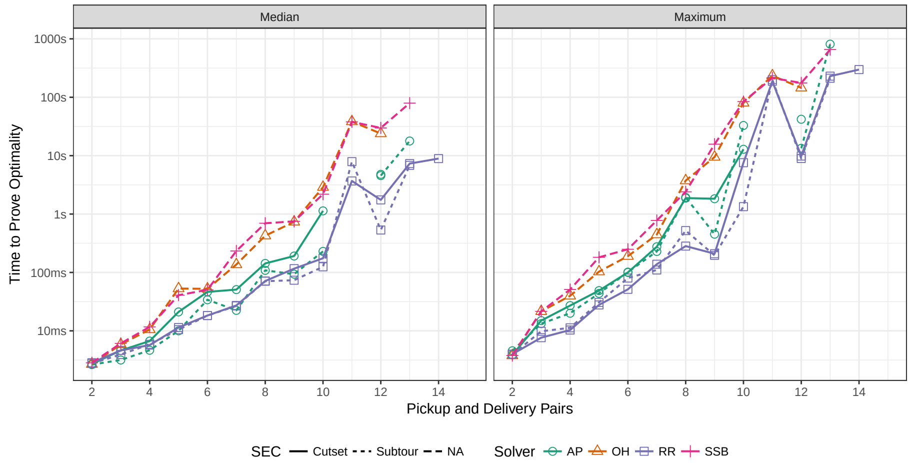

Fig. 2 Time to prove optimality for TSPPD MIP models.

results comparing the best of the previous model configurations against symmetric and asymmetric problem instances, respectively.

6.1 Polynomial Length Models

Figure 2 shows the performance of the complete SSB and OH models compared with the relaxed RR and AP models. For symmetric problems, the relaxed models outperform the complete ones as optimizers, often by an order of magnitude, though the AP formulation does not perform as well as the RR model on the larger instances. The subtour SEC form tends to be faster than the cutset form, and is able to fully optimize more instances in the AP model. A notable exception to this is the RR model, in which the cutset form can fully optimize all test instances with 14 pairs, while the subtour form cannot. For asymmetric problems, as discussed further in section 6.6, we observe similar order of magnitude speedups in median time to prove optimality due to model relaxation.

6.2 Valid Inequalities for the Sarin-Sherali-Bhootra Model

Figure 3 adds the valid inequality sets `A` and `B` to the complete SSB model. Since this is the complete model, it does not use SEC or precedence forms. The original SSB model without additional inequalities is denoted `None` while `All` incorporates the inequalities of both `A` and `B`. We observe that the addition of these inequalities does not significantly impact time to prove optimality for the complete SSB model.

6.3 Relaxed Sarin-Sherali-Bhootra Model

Figure 4 shows the same sets of inequalities applied to the relaxed SSB model. SEC are added to the relaxed form using the subtour and cutset forms, while precedence

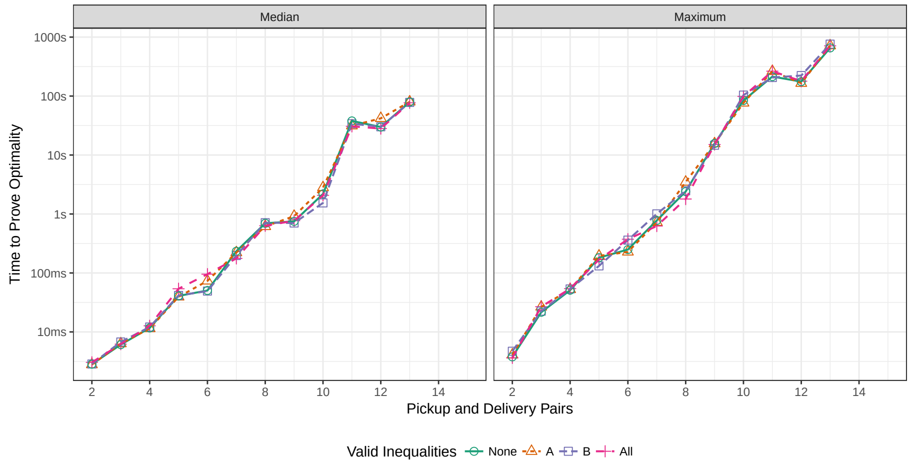

Fig. 3 Time to prove optimality for SSB model with valid inequalities.

is enforced using constraints on the x and y variables. The complete model does not use either of these, indicated by `NA`, and does not include additional valid inequalities. Points that are missing on the graph indicate model configurations which were unable to prove optimality for all test instances of that problem size within 1000 seconds.

In particular, constraint set `A` improves the ability of the relaxed SSB model to optimize ATSPPD instances. We observe that the relaxed SSB with constraint set `A`, the subtour SEC form, and precedence constraints on the x variables is the only configuration able to fully optimize all test instances through 14 pickup and delivery pairs in under 1000 seconds.

6.4 Relaxed O'Neil-Hoffman Model

Figure 5 compares the complete and relaxed versions of the OH model. SEC are added to the relaxed form using the subtour and cutset forms, while precedence is enforced using constraints on x variables. The complete model is denoted by `NA` for these options.

We observe a similar impact of relaxation among the SSB and OH models. Both experience speedups of an order of magnitude or more optimizing the larger instances of the test set, and both are able to optimize larger instances fully within the time limit. The subtour SEC form is frequently the most effective configuration.

6.5 Comparison of Models on Symmetric Instances

Figure 6 compares the dominant techniques from the previous sections on symmetric test instances. RR and AP are the original relaxed STSPPD and ATSPPD forms. SSB+A/X is the relaxed SSB model with valid inequality set `A` and precedence constraints lazily added to the x variables. OH/X is the relaxed OH model

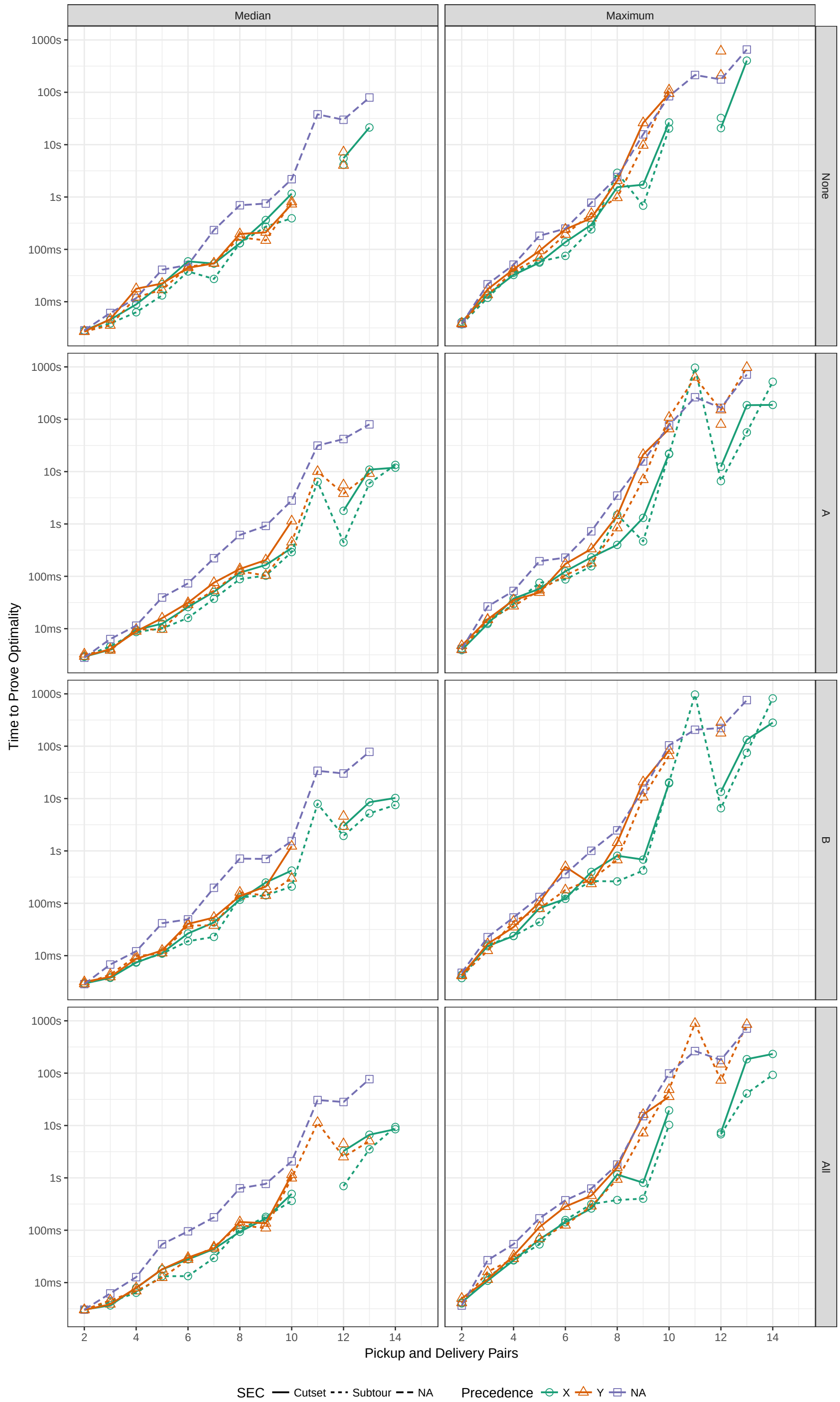

Fig. 4 Time to prove optimality for relaxed SSB models with valid inequalities, SEC forms, and precedence forms.

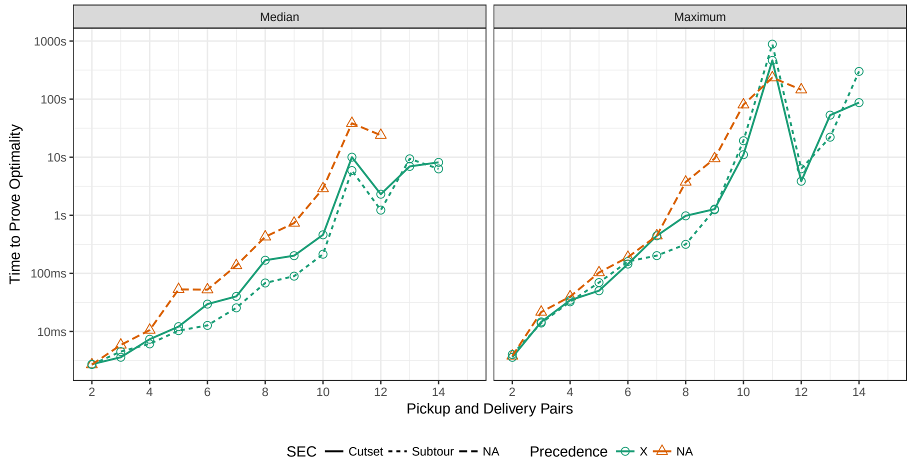

Fig. 5 Time to prove optimality for relaxed OH models with SEC and precedence forms.

with precedence constraints lazily added to the x variables. The rows in the chart correspond to the cutset and subtour SEC forms.

Tables 2, 3, 4, 5, and 6 provide median times to find feasible solutions, find solutions within 10% of the true optimum, find solutions within 5% of the true optimum, find optimal solutions, and prove optimality of solutions for the symmetric instances in the test set. If a configuration is unable to achieve one of these benchmarks for all instances of a given size, it is removed from consideration for that size. Charts for all computational experiments are provided in the appendix.

We observe that, of the models tested, the complete SSB formulation is the fastest to produce feasible solutions. This is consistent with results from previous studies comparing it to the relaxed RR model. Interestingly, in spite of not requiring relaxation, the complete OH model is particularly ill equipped to produce feasible solutions, frequently faring worse than its relaxed forms. This argues the utility of warm starting these models with high quality solutions.

The AP model with subtour SEC form tends to perform best at finding solutions within 10% and 5% of optimal up through 8 pickup and delivery pairs. After that point, the RR model is the best option. When it comes to finding optimal solutions and proving optimality, the RR model loses its edge and is often outperformed by either the OH/X or the SSB+A/X models. The OH/X model performs best at both of these tasks for problems with 14 pickup and delivery pairs, while the SSB+A/X performs best for problems with 12 and 13 pickup and delivery pairs.

6.6 Comparison of Models on Asymmetric Instances

Figure 7 compares the dominant techniques from the previous sections on asymmetric test instances. AP is the original relaxed ATSPPD form. SSB and OH are the complete ATSPPD formulation. SSB+A/X is the relaxed SSB model with valid inequality set `A` and precedence constraints lazily added to the x variables.

Model	SEC	Pickup and Delivery Pairs 2	3	4	5	6	7	8
AP	Cutset	1.9	3.1	5.1	7.9	15.1	15.3	**2.9**
AP	Subtour	**1.8**	3.0	3.7	**5.0**	11.3	**8.5**	**2.9**
OH	-	1.9	2.7	4.0	11.9	27.2	46.7	128.0
OH/X	Cutset	1.9	**2.1**	**2.3**	5.9	9.8	16.1	19.3
OH/X	Subtour	1.9	2.4	2.4	6.0	8.4	11.9	50.4
RR	Cutset	2.1	4.2	5.0	6.9	9.3	14.8	35.5
RR	Subtour	2.1	3.6	4.4	6.2	8.4	13.0	55.6
SSB	-	2.0	2.6	3.6	5.2	**8.0**	12.4	17.3
SSB+A/X	Cutset	1.9	2.4	5.1	7.1	11.0	15.5	30.0
SSB+A/X	Subtour	2.0	2.3	5.0	6.2	9.3	12.1	32.6

Model	SEC	9	10	11	12	13	14	15
AP	Cutset	64.8	260.0	1473.0	1639.0	5170.0	-	10,035
AP	Subtour	68.9	84.5	377.0	408.0	2192.0	5855.0	16,842
OH	-	197.0	749.0	1340.0	1757.0	2782.0	4874.0	7307
OH/X	Cutset	120.0	99.4	1365.0	605.0	2828.0	4484.0	6748
OH/X	Subtour	51.0	63.0	271.0	482.0	2804.0	1623.0	10,118
RR	Cutset	83.3	130.0	245.0	246.0	475.0	1236.0	6128
RR	Subtour	49.5	81.2	256.0	176.0	393.0	756.0	9460
SSB	-	**22.8**	**31.4**	**40.2**	**54.2**	**69.1**	**86.8**	**106**
SSB+A/X	Cutset	83.1	232.0	1348.0	314.0	2343.0	4869.0	11,911
SSB+A/X	Subtour	58.6	165.0	590.0	315.0	990.0	1827.0	1199

Table 2 Median time to find a feasible solution in milliseconds on symmetric instances.

Model	SEC	Pickup and Delivery Pairs 2	3	4	5	6	7	8
AP	Cutset	2.3	3.2	5.8	12.1	22.9	26.0	73.8
AP	Subtour	2.3	**3.0**	**3.7**	**5.5**	20.2	**8.5**	**31.5**
OH	-	**2.0**	4.6	8.3	15.9	32.5	64.0	247.0
OH/X	Cutset	**2.0**	3.2	4.7	7.6	11.2	17.1	106.0
OH/X	Subtour	2.1	4.0	4.7	8.0	9.6	13.7	50.4
RR	Cutset	2.5	4.3	5.0	7.4	9.3	14.8	60.1
RR	Subtour	2.6	3.6	4.6	6.2	**8.9**	13.0	60.8
SSB	-	2.5	5.3	9.3	16.3	35.5	54.0	276.0
SSB+A/X	Cutset	2.8	3.7	5.4	10.3	15.0	19.0	54.1
SSB+A/X	Subtour	2.9	3.7	5.4	7.1	11.3	13.8	66.4

Model	SEC	9	10	11	12	13	14	15
AP	Cutset	96.4	601.0	-	1889	-	-	109,564
AP	Subtour	68.9	190.0	-	2409	6503	7036	-
OH	-	349.0	998.0	2398	2048	5835	10,870	18,859
OH/X	Cutset	147.0	191.0	3539	766	3357	6373	28,909
OH/X	Subtour	65.9	147.0	810	599	2804	6141	21,146
RR	Cutset	83.3	130.0	503	381	**525**	4116	25,022
RR	Subtour	**64.4**	**81.2**	**386**	**194**	699	**1085**	**9505**
SSB	-	144.0	714.0	1454	2130	3206	7642	45,379
SSB+A/X	Cutset	93.1	245.0	3670	490	2962	5328	41,664
SSB+A/X	Subtour	84.9	187.0	699	348	1559	5955	22,013

Table 3 Median time to find a solution within 10% of optimal in milliseconds on symmetric instances.

		Pickup and Delivery Pairs						
Model	SEC	2	3	4	5	6	7	8
AP	Cutset	2.5	3.6	5.8	13.4	29.7	27.2	97.4
AP	Subtour	2.4	**3.0**	**3.7**	7.0	27.7	**8.5**	**34.4**
OH	-	**2.3**	5.1	8.5	17.1	32.5	72.4	338.0
OH/X	Cutset	2.6	3.3	5.1	7.6	13.4	21.4	152.0
OH/X	Subtour	2.4	4.0	5.0	8.0	9.6	13.7	58.0
RR	Cutset	2.5	4.3	5.2	7.4	9.3	14.8	62.0
RR	Subtour	2.6	3.6	5.0	**6.2**	**8.9**	26.2	60.8
SSB	-	2.5	5.4	10.3	16.3	35.5	58.2	387.0
SSB+A/X	Cutset	2.8	3.7	5.6	10.3	16.8	19.5	65.0
SSB+A/X	Subtour	2.9	3.8	5.7	7.1	12.8	24.1	72.7
		9	10	11	12	13	14	15
AP	Cutset	96.4	693.0	-	2747	-	-	-
AP	Subtour	79.2	190.0	-	2573	8938	-	-
OH	-	532.0	1113.9	4033	2648	18,480	34,112	-
OH/X	Cutset	173.0	227.0	3539	766	5010	7182	29,545
OH/X	Subtour	**65.9**	147.0	2789	920	4994	6196	35,481
RR	Cutset	92.9	134.0	**676**	381	**565**	6778	40,732
RR	Subtour	72.1	**94.9**	2413	**209**	2656	**4301**	**9505**
SSB	-	380.0	714.0	2931	2836	8158	20,253	143,594
SSB+A/X	Cutset	125.0	294.0	3964	711	3570	5328	64,569
SSB+A/X	Subtour	88.1	209.0	2293	348	2668	7402	-

Table 4 Median time to find a solution within 5% of optimal in milliseconds on symmetric instances.

		Pickup and Delivery Pairs						
Model	SEC	2	3	4	5	6	7	8
AP	Cutset	2.5	4.3	5.8	13.4	38.5	35.9	129.0
AP	Subtour	**2.4**	**3.0**	**4.3**	9.1	33.3	21.6	79.5
OH	-	2.6	5.1	9.9	26.7	47.1	82.5	340.0
OH/X	Cutset	2.6	3.3	6.5	9.6	29.0	30.2	166.0
OH/X	Subtour	2.6	4.4	5.5	10.0	**12.0**	22.7	65.5
RR	Cutset	2.5	4.4	5.2	7.4	15.7	**17.0**	**62.0**
RR	Subtour	2.6	3.8	5.1	**6.2**	15.4	26.2	63.3
SSB	-	2.5	5.6	10.6	26.0	50.1	139.0	555.0
SSB+A/X	Cutset	2.8	3.8	6.4	11.6	25.5	47.7	81.0
SSB+A/X	Subtour	2.9	3.8	6.3	9.9	13.0	37.1	75.2
		9	10	11	12	13	14	15
AP	Cutset	171.0	1107	-	4446	-	-	-
AP	Subtour	82.2	197	-	4479	12,663	-	-
OH	-	695.0	2227	35,248	14,114	34,552	-	-
OH/X	Cutset	197.0	415	6392	2023	6000	8083	-
OH/X	Subtour	79.6	194	5667	1104	8039	**6196**	-
RR	Cutset	109.0	161	**2481**	1728	6654	8830	-
RR	Subtour	**72.1**	**109**	6861	520	4412	8762	-
SSB	-	676.0	1438	31,749	20,570	37,519	-	-
SSB+A/X	Cutset	128.0	343	5681	1746	8306	7402	-
SSB+A/X	Subtour	97.6	288	5698	**431**	**3137**	12,584	-

Table 5 Median time to find an optimal solution in milliseconds on symmetric instances.

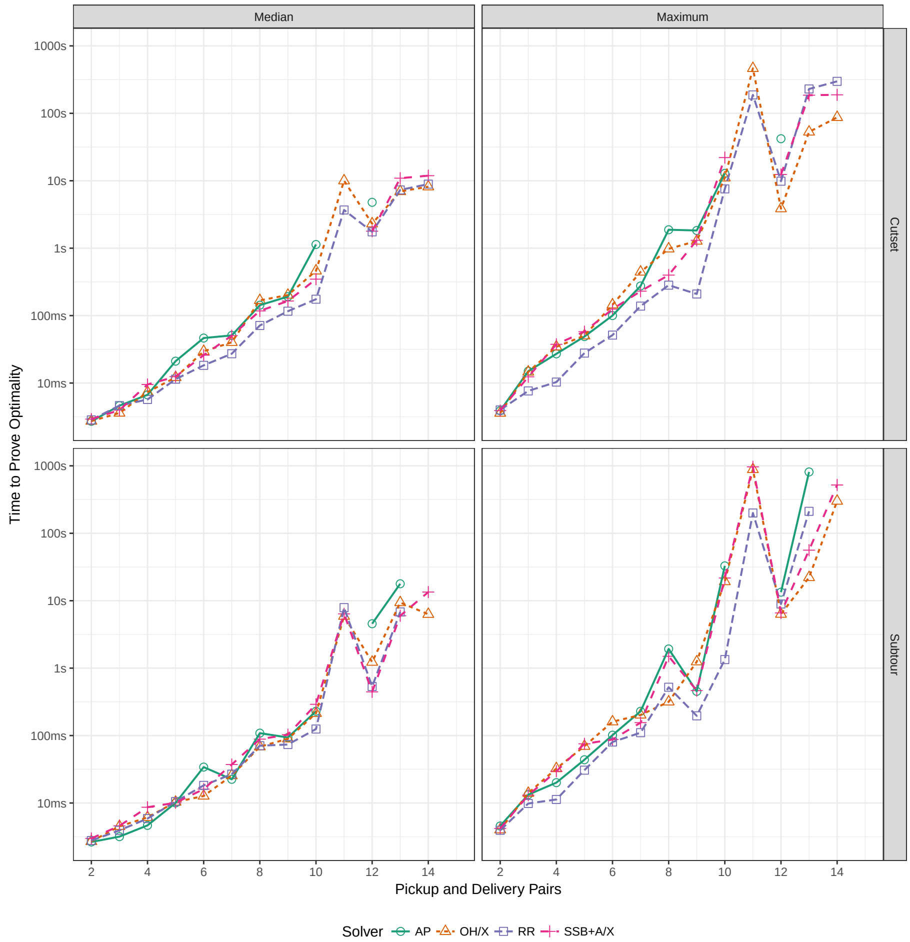

Fig. 6 Time to prove optimality for the best models.

OH/X is the relaxed OH model with precedence constraints lazily added to the x variables. The rows in the chart correspond to the cutset and subtour SEC forms.

Tables 7, 8, 9, 10, and 11 provide median times to find feasible solutions, find solutions within 10% of the true optimum, find solutions within 5% of the true optimum, find optimal solutions, and prove optimality of solutions for the asymmetric instances in the test set. Once again, if a configuration is unable to achieve one of these benchmarks for all instances of a given size, it is removed from consideration for that size.

Much of the general behavior of the models observed in section 6.5 applies to asymmetric instances, and performance of the asymmetric models is similar. Again, the complete SSB formulation is fastest to produce feasible solutions and the OH model suffers in this capacity.

One difference we note in solving asymmetric instances is the apparent power of the OH/X model at finding high quality solutions and at finding and proving

		Pickup and Delivery Pairs						
Model	SEC	2	3	4	5	6	7	8
AP	Cutset	2.7	4.6	6.7	21.1	46.5	50.9	143.0
AP	Subtour	**2.6**	**3.2**	**4.7**	**10.0**	34.1	**22.4**	109.0
OH	-	2.7	5.8	10.6	53.0	52.4	137.0	426.0
OH/X	Cutset	2.7	3.6	7.3	12.1	29.6	40.2	168.0
OH/X	Subtour	2.7	4.5	6.1	10.3	**12.7**	25.5	**68.3**
RR	Cutset	2.8	4.6	5.7	11.4	18.3	27.2	71.3
RR	Subtour	2.8	3.9	5.9	10.5	18.2	26.7	70.7
SSB	-	2.8	6.1	11.7	40.8	50.4	234.0	696.0
SSB+A/X	Cutset	2.9	4.0	9.5	12.4	25.9	50.4	118.0
SSB+A/X	Subtour	3.0	4.6	8.6	10.1	16.1	37.2	88.5
		9	10	11	12	13	14	15
AP	Cutset	191.0	1135	-	4792	-	-	-
AP	Subtour	93.6	229	-	4556	17,834	-	-
OH	-	735.0	2885	38,386	23,916	-	-	-
OH/X	Cutset	202.0	461	10,016	2307	6940	8157	-
OH/X	Subtour	89.6	214	5899	1228	9392	**6299**	-
RR	Cutset	116.0	175	**3689**	1748	7314	8911	-
RR	Subtour	**73.6**	**125**	7899	534	6817	-	-
SSB	-	750.0	2188	37,921	29,847	79,362	-	-
SSB+A/X	Cutset	165.0	347	-	1786	10,912	11,886	-
SSB+A/X	Subtour	103.0	291	6386	**445**	**5979**	13,440	-

Table 6 Median time to prove optimality in milliseconds on symmetric instances.

		Pickup and Delivery Pairs						
Model	SEC	2	3	4	5	6	7	8
AP	Cutset	**1.8**	3.5	5.3	5.3	9.2	46.3	2.8
AP	Subtour	**1.8**	3.1	4.2	**4.9**	8.5	30.4	**2.3**
OH	-	1.9	2.4	4.0	12.2	25.4	45.1	198.0
OH/X	Cutset	2.0	**2.1**	**2.5**	5.8	7.7	11.3	33.5
OH/X	Subtour	1.9	**2.1**	**2.5**	5.7	**7.5**	**9.4**	30.6
SSB	-	2.0	2.8	3.6	5.3	8.3	10.8	16.9
SSB+A/X	Cutset	2.2	2.3	5.3	6.1	8.8	10.8	33.7
SSB+A/X	Subtour	2.3	2.2	5.0	6.0	8.5	12.5	40.5
		9	10	11	12	13	14	15
AP	Cutset	187.0	101.0	1470.0	1535.0	2745.0	-	86,987
AP	Subtour	55.9	120.0	217.0	504 0	3598.0	1162.0	-
OH	-	183.0	726.0	1211.0	509 0	2662.0	3803.0	6453
OH/X	Cutset	63.9	205.0	1944.0	1540.0	3084.0	4899.0	4357
OH/X	Subtour	560.0	214.0	438.0	259.0	782.0	1322.0	10,430
SSB	-	**24.1**	**30.6**	**40.1**	**50.3**	**64.1**	**80.7**	**100**
SSB+A/X	Cutset	133.0	625.0	390.0	2183.0	2868.0	5740.0	6232
SSB+A/X	Subtour	56.8	90.5	497.0	427.0	662.0	5185.0	11,591

Table 7 Median time to find a feasible solution in milliseconds on asymmetric instances.

optimality for instances with 9 or more pickup and delivery pairs. In particular, OH/X with the subtour form of SEC is most frequently the fastest of the asymmetric models at these tasks.

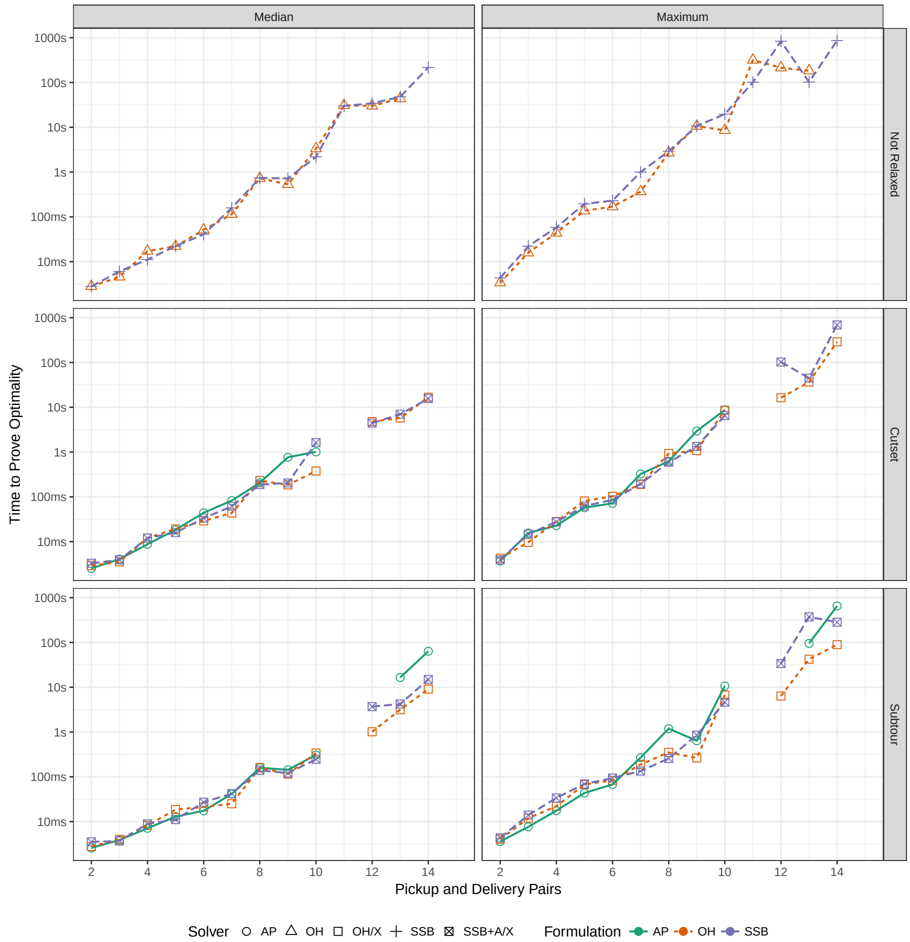

Fig. 7 Time to prove optimality for the best models on asymmetric instances.

7 Conclusions and Future Work

In the paper we examine the performance of different MIP models for solving TSPPDs, measuring their utility in solving real-time logistics problems, in which routes must be optimized within strict time budgets. Our goal is to build ATSPPD models that have similar performance characteristics to the STSPPD MIP version of the Ruland and Rodin (1997) model.

We implement relaxed models of symmetric and asymmetric forms based on two-matching and assignment problem relaxations. To these we compare the complete formulation of Sarin et al. (2005) with and without new valid inequalities for the ATSPPD, and a new MIP ATSPPD model that includes a secondary variable set specifying precedence configurations among multiple pickup and delivery pairs. Both complete ATSPPD forms are then relaxed. SEC are added to the x variables

Model	SEC	Pickup and Delivery Pairs 2	3	4	5	6	7	8
AP	Cutset	2.4	3.5	5.6	7.1	10.8	74.4	**74.4**
AP	Subtour	**1.9**	3.4	**4.9**	10.7	12.5	32.9	112.0
OH	-	2.3	4.3	9.3	15.8	29.6	52.5	261.0
OH/X	Cutset	2.0	**3.3**	6.6	**6.7**	**7.7**	12.7	78.2
OH/X	Subtour	2.0	3.6	5.3	7.3	9.8	**11.7**	106.0
SSB	-	2.7	5.4	9.9	16.1	32.5	51.3	226.0
SSB+A/X	Cutset	3.0	3.6	7.4	8.1	10.4	23.8	86.3
SSB+A/X	Subtour	3.0	3.4	6.4	9.1	9.8	25.9	77.9
		9	10	11	12	13	14	15
AP	Cutset	292.0	295	4461	2288	3603	-	196,670
AP	Subtour	**55.9**	**181**	4531	3385	5824	11,351	-
OH	-	226.0	843	1812	2382	3826	6744	**17,869**
OH/X	Cutset	150.0	342	3443	2261	3696	13,388	35,242
OH/X	Subtour	56.0	290	**877**	**350**	**936**	**5651**	30,334
SSB	-	368.0	798	1284	2256	6026	6437	27,090
SSB+A/X	Cutset	195.0	921	3457	2711	4779	8031	25,420
SSB+A/X	Subtour	86.9	**181**	4345	667	2219	12,009	-

Table 8 Median time to find a solution within 10% of optimal in milliseconds on asymmetric instances.

Model	SEC	Pickup and Delivery Pairs 2	3	4	5	6	7	8
AP	Cutset	2.4	3.5	6.4	**7.1**	14.0	76.5	**86.4**
AP	Subtour	**2.1**	3.4	**5.0**	10.7	12.5	35.0	141.0
OH	-	2.4	4.3	9.8	15.8	29.7	66.0	330.0
OH/X	Cutset	2.5	**3.3**	7.5	8.2	11.6	28.4	124.0
OH/X	Subtour	2.3	3.6	5.7	10.0	11.1	**13.7**	124.0
SSB	-	2.7	5.4	10.1	18.7	32.5	68.5	286.0
SSB+A/X	Cutset	3.0	3.7	8.7	11.2	12.0	25.0	114.0
SSB+A/X	Subtour	3.0	3.6	6.7	10.5	**10.6**	30.3	100.0
		9	10	11	12	13	14	15
AP	Cutset	302.0	518	6594	2542	4194	-	221,132
AP	Subtour	133.0	195	5002	4645	8635	13,845	-
OH	-	299.0	1002	4547	3045	9187	12,096	77,133
OH/X	Cutset	163.0	342	3738	2626	3696	14,704	**47,378**
OH/X	Subtour	**63.7**	301	**878**	**364**	**936**	**5706**	50,053
SSB	-	479.0	798	5738	2731	9929	10,806	62,624
SSB+A/X	Cutset	195.0	1077	3734	2960	5799	8115	-
SSB+A/X	Subtour	86.9	**182**	6156	1002	3365	13,091	-

Table 9 Median time to find a solution within 5% of optimal in milliseconds on asymmetric instances.

using lazy constraints of multiple forms, and precedence constraints are added on the x and y variables, where applicable.

We find that relaxation of SEC in the complete models is highly effective in improving their performance, and that the impact of valid inequalities added to the Sarin et al. (2005) model increases in the presence of relaxation. We also find that, on asymmetric problem instances, our new MIP ATSPPD model tends to be more effective than the other models tested at finding high quality solutions, and proving optimality, though it is poor at finding feasible solutions quickly on its own.

Model	SEC	Pickup and Delivery Pairs 2	3	4	5	6	7	8
AP	Cutset	**2.4**	3.8	7.5	16.1	21.3	80.2	169
AP	Subtour	2.5	3.8	**6.5**	12.2	**12.9**	38.8	155
OH	-	**2.4**	4.3	16.6	21.8	41.8	101.0	618
OH/X	Cutset	2.8	**3.4**	9.9	18.9	28.8	36.6	188
OH/X	Subtour	2.7	3.7	**6.5**	17.9	14.8	**16.3**	153
SSB	-	2.7	5.9	11.0	21.9	39.8	113.0	617
SSB+A/X	Cutset	3.0	3.7	10.3	14.3	20.4	41.5	173
SSB+A/X	Subtour	3.0	3.6	6.7	**10.6**	25.5	38.4	**112**
		9	10	11	12	13	14	15
AP	Cutset	544	1002	-	7054	-		–
AP	Subtour	139	266	-	7462	16,387	56,184	-
OH	-	515	1916	18,970	29,282	30,047	-	-
OH/X	Cutset	182	342	-	4609	5597	16,064	-
OH/X	Subtour	**111**	336	**2736**	**883**	**2930**	**8258**	-
SSB	-	718	1439	20,862	20,190	37,461	112,366	-
SSB+A/X	Cutset	202	1588	-	4387	6947	12,180	-
SSB+A/X	Subtour	116	**241**	-	3621	4178	14,491	-

Table 10 Median time to find an optimal solution in milliseconds on asymmetric instances.

Model	SEC	Pickup and Delivery Pairs 2	3	4	5	6	7	8
AP	Cutset	**2.5**	4.1	8.7	18.4	44.2	82.4	207
AP	Subtour	2.6	3.9	**7.1**	12.9	**17.4**	41.4	159
OH	-	2.8	4.6	17.2	22.0	51.4	114.0	728
OH/X	Cutset	2.9	**3.5**	12.1	19.2	29.0	43.6	230
OH/X	Subtour	2.8	4.0	8.2	18.6	**21.1**	**25.1**	160
SSB	-	2.8	6.0	11.1	22.2	40.8	159.0	738
SSB+A/X	Cutset	3.3	3.9	12.0	15.9	33.4	61.4	188
SSB+A/X	Subtour	3.5	3.7	8.9	**11.1**	27.0	42.1	**140**
		9	10	11	12	13	14	15
AP	Cutset	763	1013	-	-	-	-	-
AP	Subtour	143	308	-	-	16,542	63,458	-
OH	-	532	3340	30,957	30,032	43,887	-	-
OH/X	Cutset	185	376	-	4799	5752	16,741	-
OH/X	Subtour	**115**	339	-	**1014**	**3143**	**9085**	-
SSB	-	724	2182	30,065	34,142	48,025	216,286	-
SSB+A/X	Cutset	204	1621	-	4440	6981	15,604	-
SSB+A/X	Subtour	121	**246**	-	3693	4214	14,923	-

Table 11 Median time to prove optimality in milliseconds on asymmetric instances.

We believe these results are interesting and useful in that they propose asymmetric models that are competitive with their symmetric counterparts. We expect that our newly proposed MIP ATSPPD model may adapt particularly well to pickup and delivery problems involving multiple vehicles, and that there are additional inequalities that may improve its performance. Both of these are avenues for future research. Finally, we intend to see if alternative warm starts will improve overall computation time.

8 Compliance with Ethical Standards

On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

Applegate DL, Bixby RE, Chvatal V, Cook WJ (2011) The traveling salesman problem: a computational study. Princeton university press

Bertsimas D, Jaillet P, Martin S (2018) Online Vehicle Routing: The Edge of Optimization in Large-Scale Applications, accepted for publication in Operations Research

Caseau Y, Laburthe F (1997) Solving Small TSPs with Constraints. In: ICLP, vol 97, p 104

Dantzig G, Fulkerson R, Johnson S (1954) Solution of a large-scale traveling-salesman problem. Journal of the operations research society of America 2(4):393–410

Dumitrescu I, Ropke S, Cordeau JF, Laporte G (2010) The traveling salesman problem with pickup and delivery: polyhedral results and a branch-and-cut algorithm. Mathematical Programming 121(2):269

Grubhub (2018) TSPPD Test Instance Library. `https://github.com/grubhub/tsppdlib`

Miller CE, Tucker AW, Zemlin RA (1960) Integer programming formulation of traveling salesman problems. Journal of the ACM 7(4):326–329

O'Neil RJ (2018) TSPPD Hybrid Optimization Code. `https://github.com/ryanjoneil/tsppd-hybrid`

O'Neil RJ, Hoffman K (2018) Exact methods for solving traveling salesman problems with pickup and delivery in real time, `http://www.optimization-online.org/DB_HTML/2017/12/6370.html`

Papadimitriou CH (1977) The Euclidean travelling salesman problem is NP-complete. Theoretical computer science 4(3):237–244

Psaraftis HN (1980) A dynamic programming solution to the single vehicle many-to-many immediate request dial-a-ride problem. Transportation Science 14(2):130–154

Reyes D, Erera A, Savelsbergh M, Sahasrabudhe S, O'Neil RJ (2018) The meal delivery routing problem, `http://www.optimization-online.org/DB_HTML/2018/04/6571.html`

Ruland K, Rodin E (1997) The pickup and delivery problem: Faces and branch-and-cut algorithm. Computers & mathematics with applications 33(12):1–13

Sarin SC, Sherali HD, Bhootra A (2005) New tighter polynomial length formulations for the asymmetric traveling salesman problem with and without precedence constraints. Operations research letters 33(1):62–70

Wickham H (2009) ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, URL `http://ggplot2.org`

9 Appendix

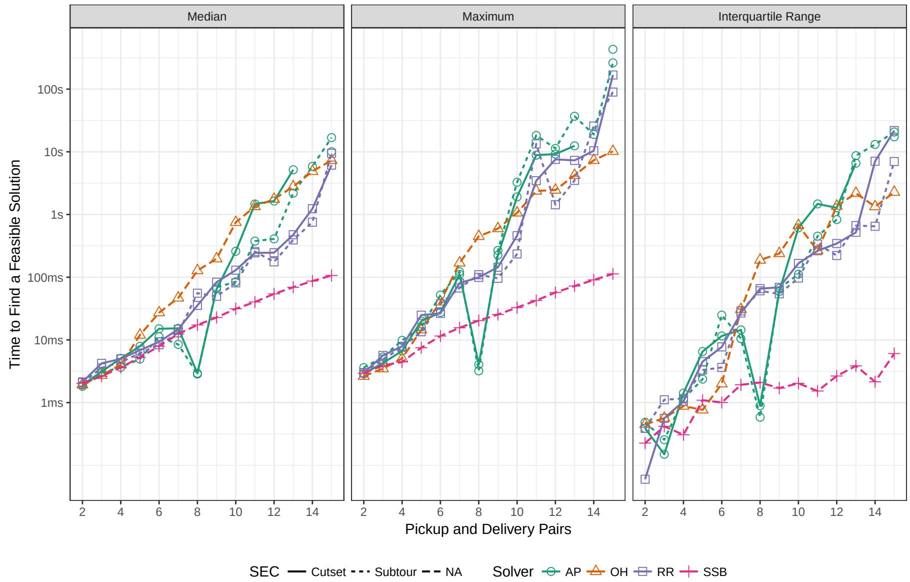

Fig. 8 Time to find a feasible solution for TSPPD MIP models.

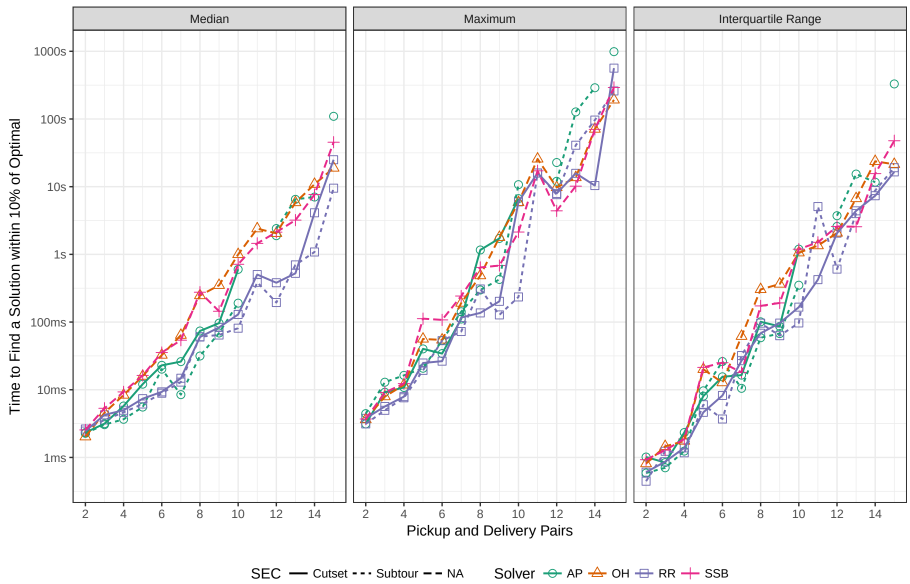

Fig. 9 Time to find a solution within 10% of optimal for TSPPD MIP models.

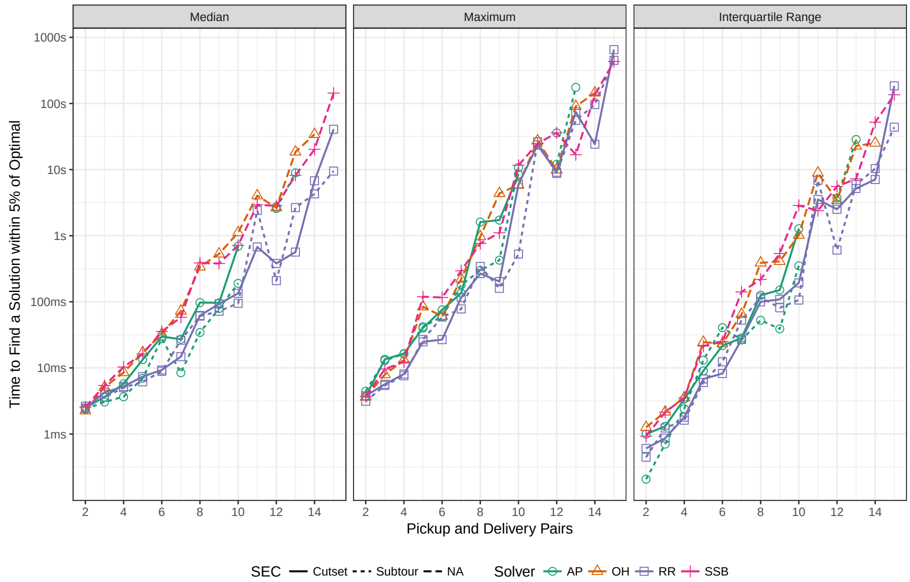

Fig. 10 Time to find a solution within 5% of optimal for TSPPD MIP models.

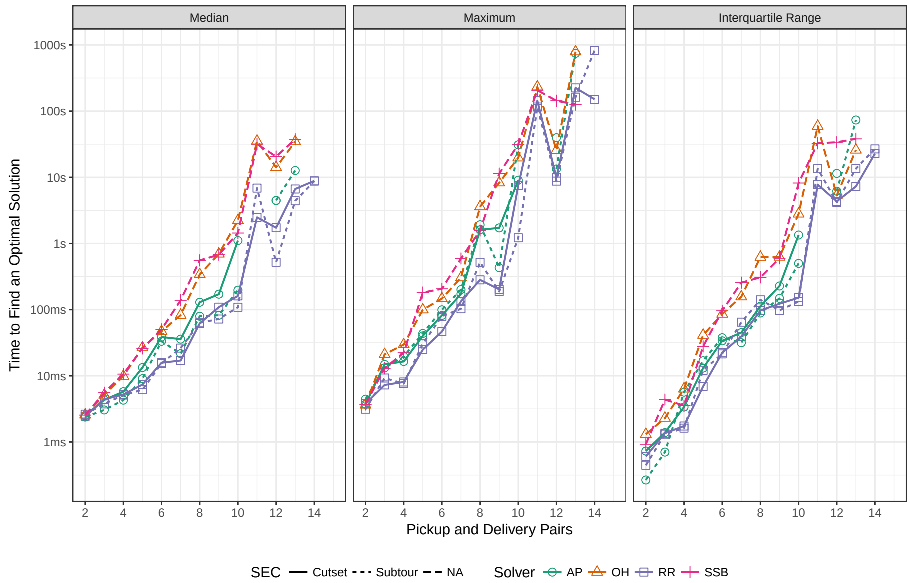

Fig. 11 Time to find an optimal solution for TSPPD MIP models.

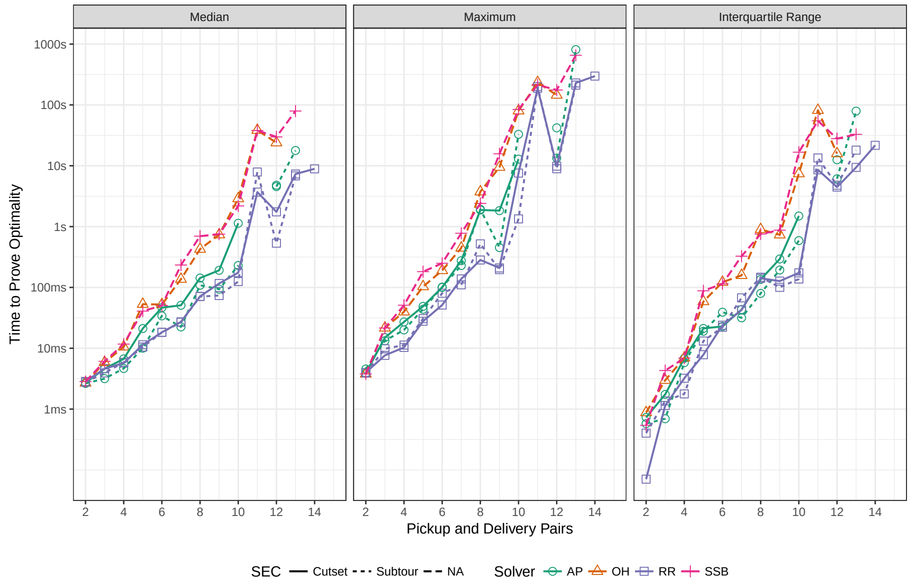

Fig. 12 Time to prove optimality for TSPPD MIP models.

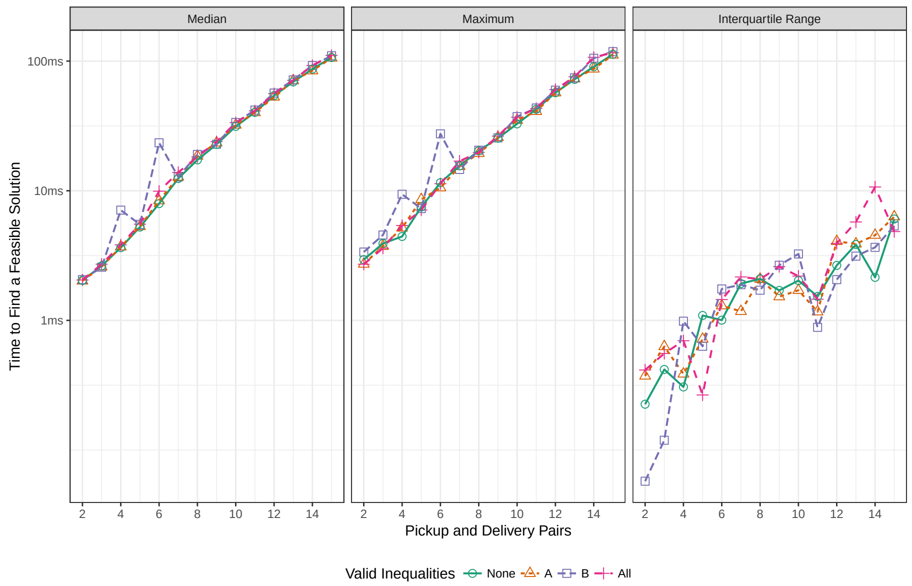

Fig. 13 Time to find a feasible solution for SSB model with valid inequalities.

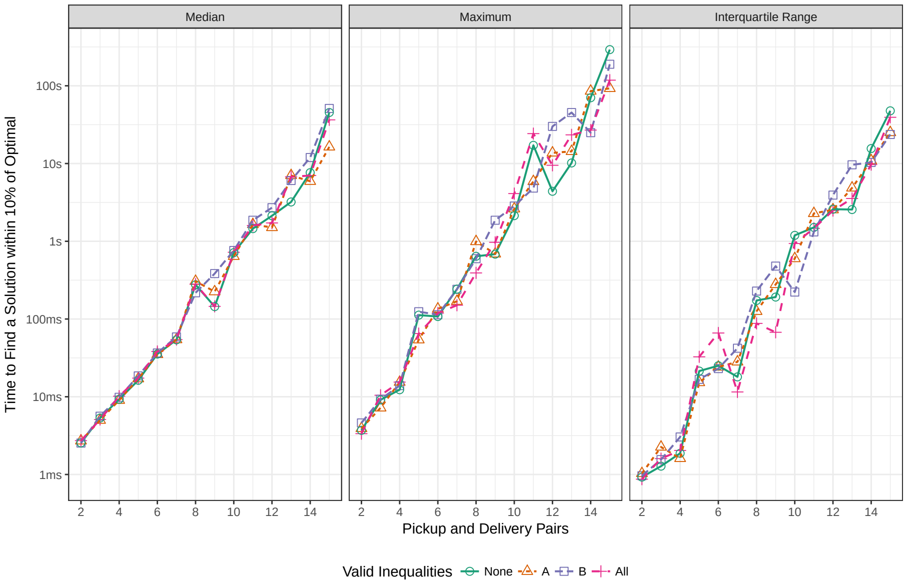

Fig. 14 Time to find a solution within 10% of optimal for SSB model with valid inequalities.

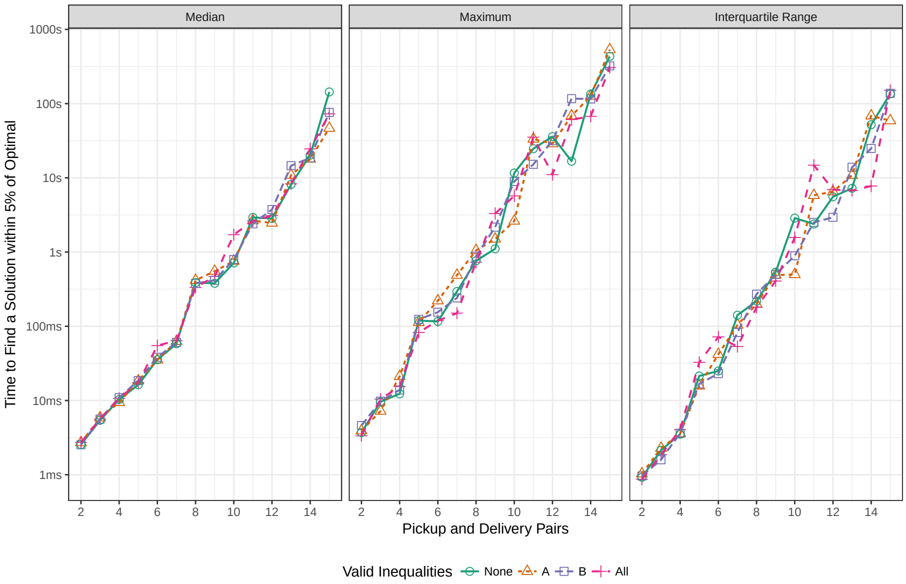

Fig. 15 Time to find a solution within 5% of optimal for SSB model with valid inequalities.

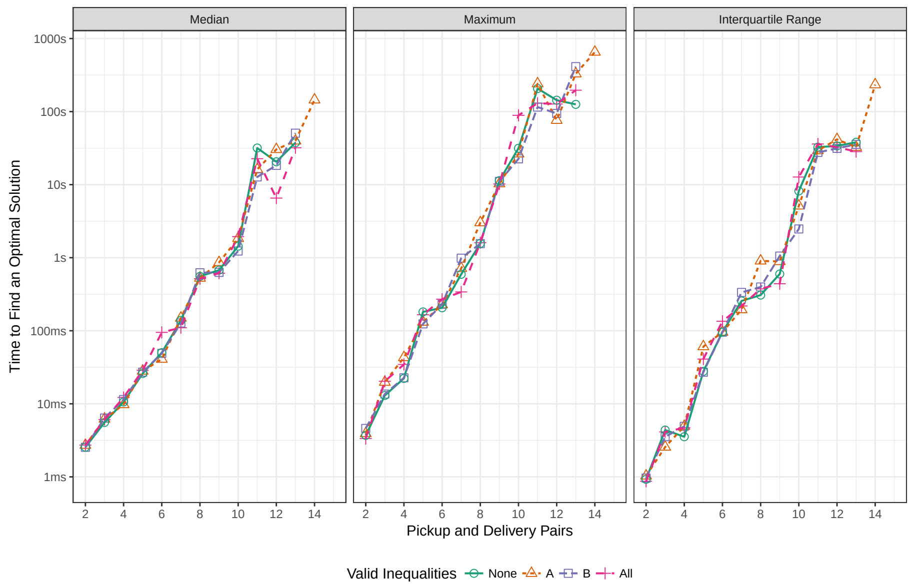

Fig. 16 Time to find an optimal solution for SSB model with valid inequalities.

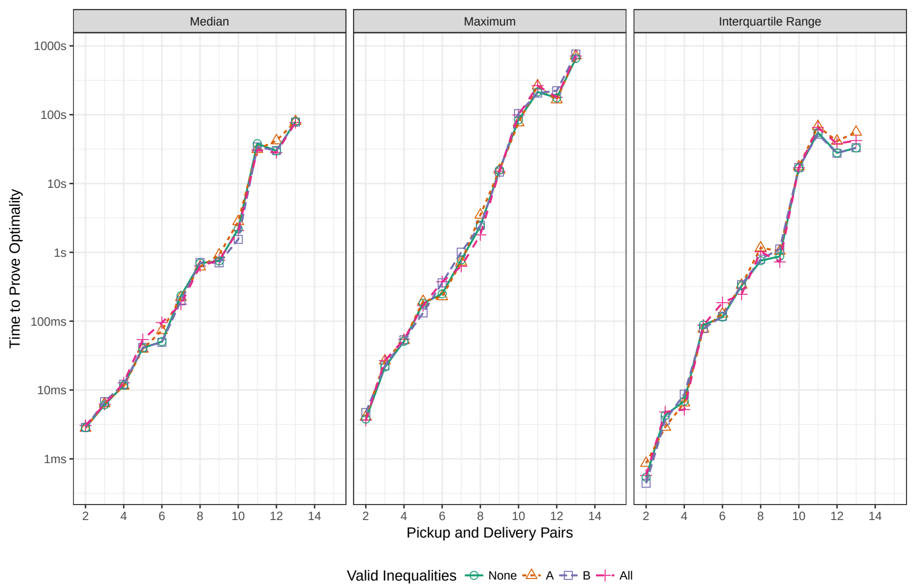

Fig. 17 Time to prove optimality for SSB model with valid inequalities.

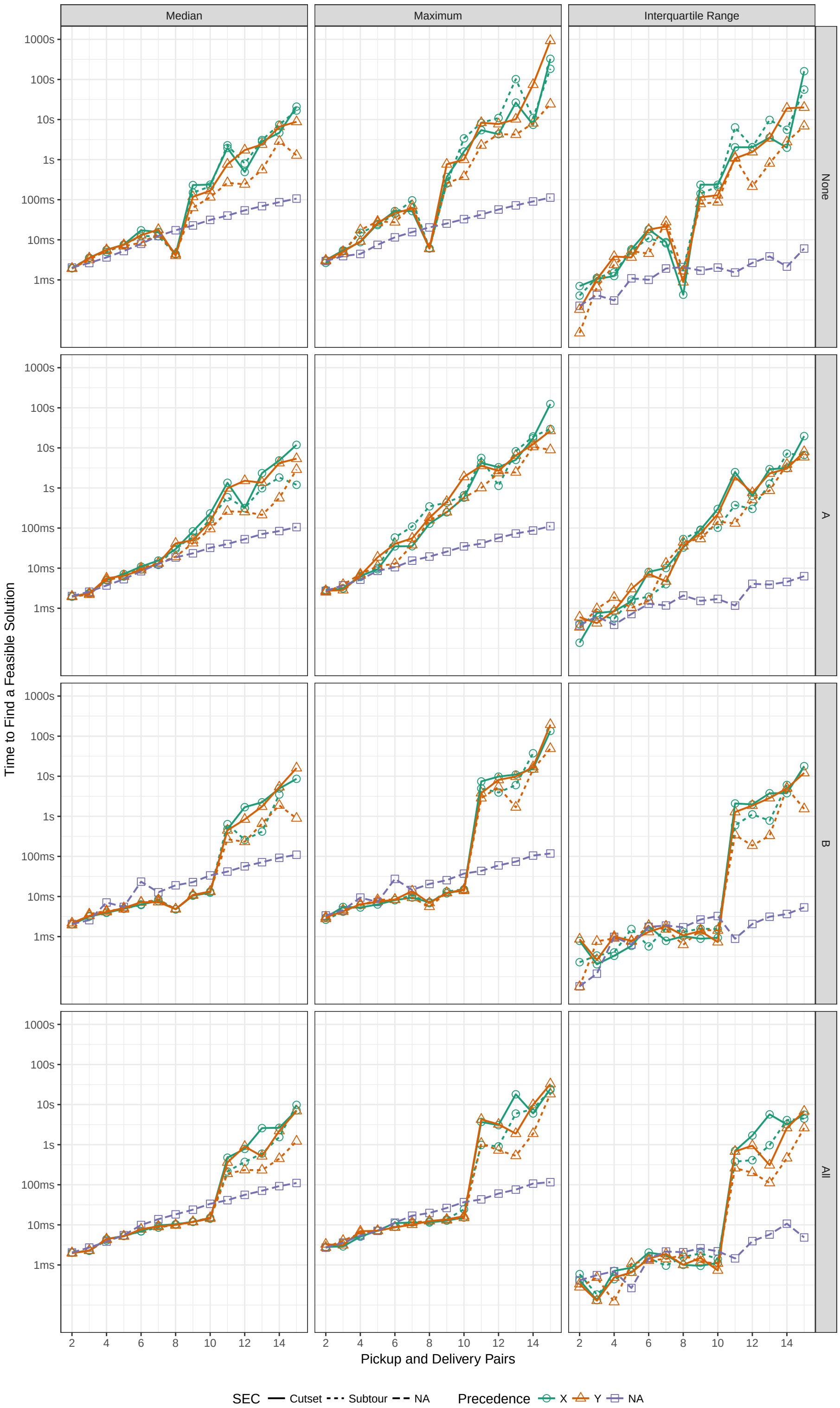

Fig. 18 Time to find a feasible solution for relaxed SSB models with valid inequalities, SEC forms, and precedence forms.

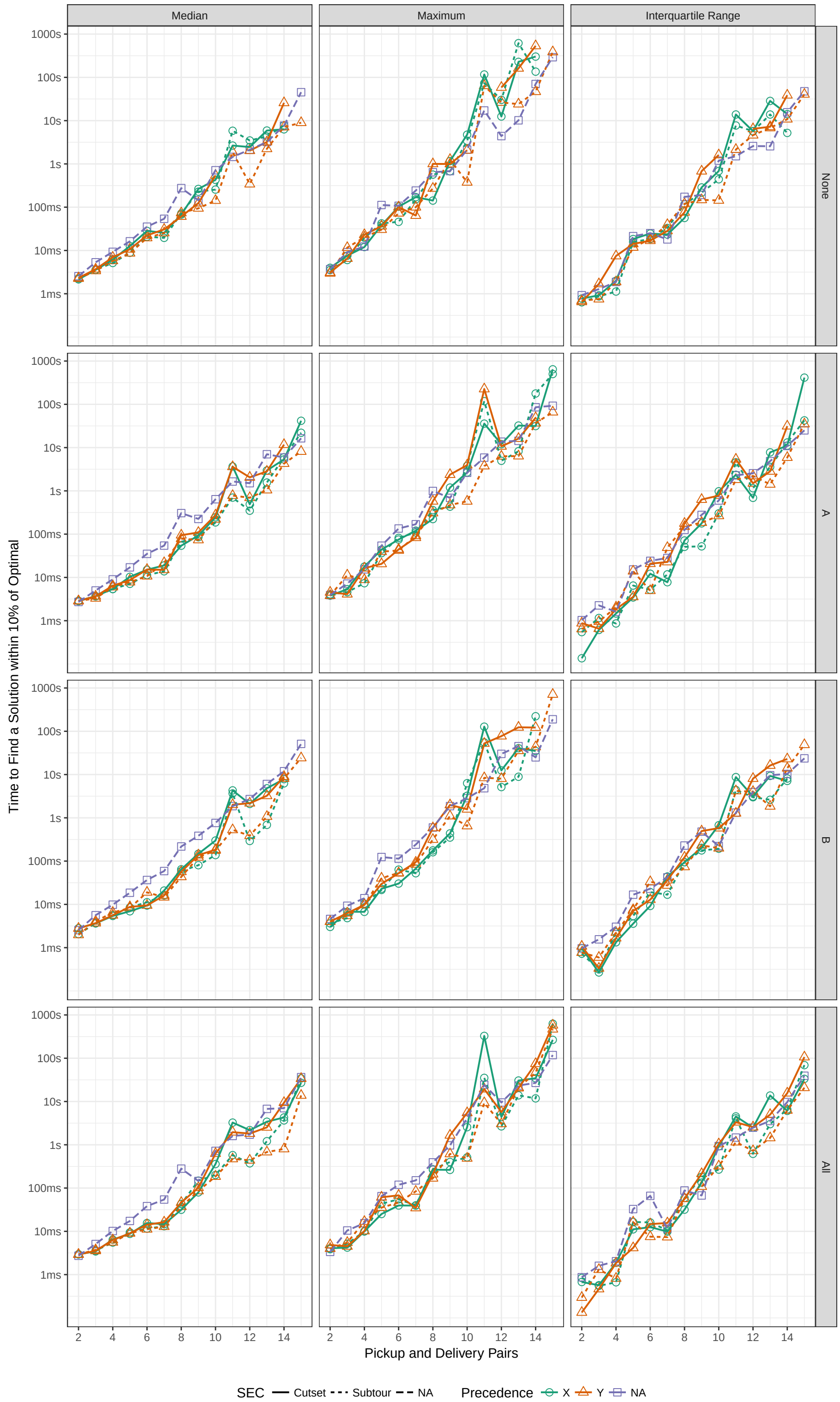

Fig. 19 Time to find a solution within 10% of optimal for relaxed SSB models with valid inequalities, SEC forms, and precedence forms.

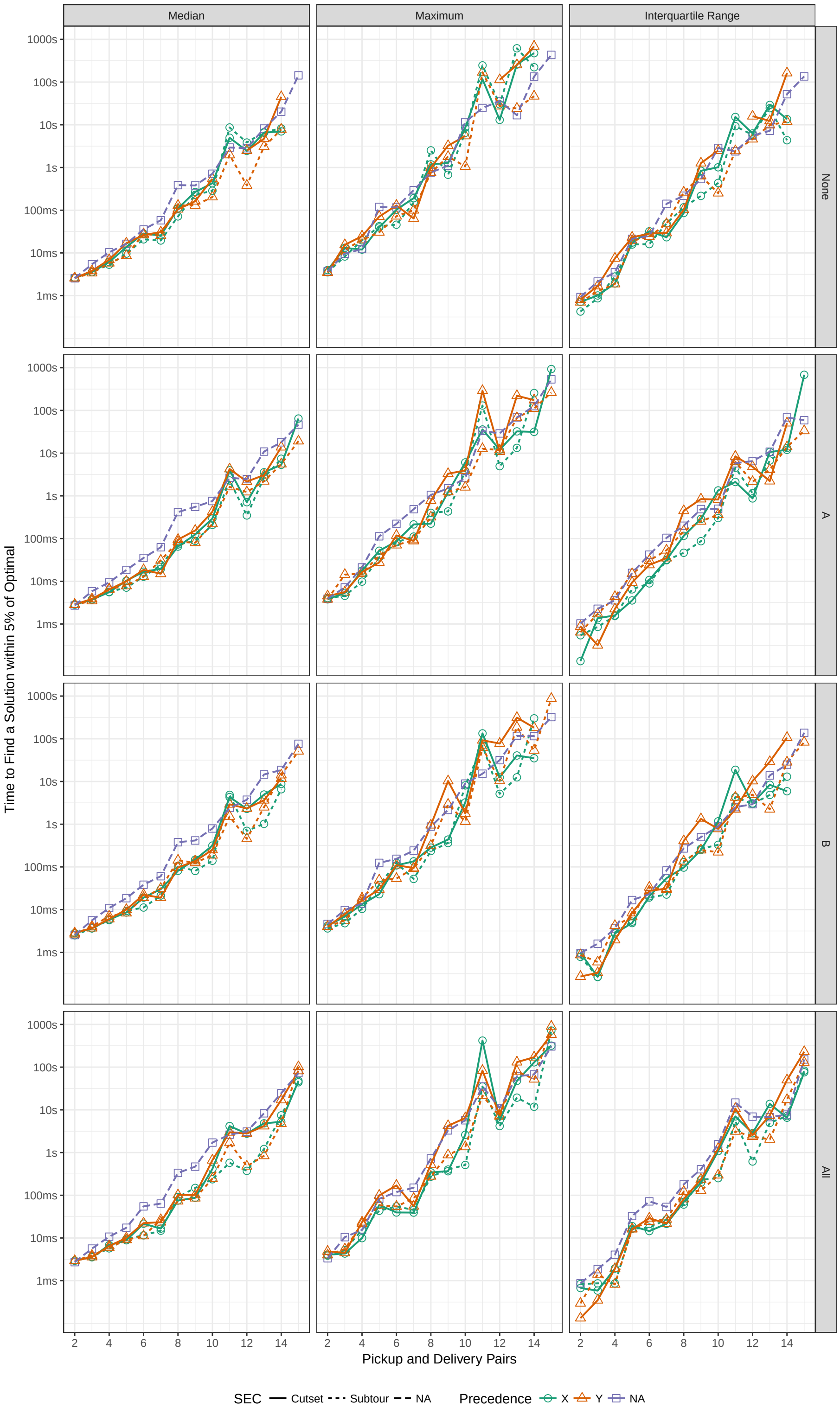

Fig. 20 Time to find a solution within 5% of optimal for relaxed SSB models with valid inequalities, SEC forms, and precedence forms.

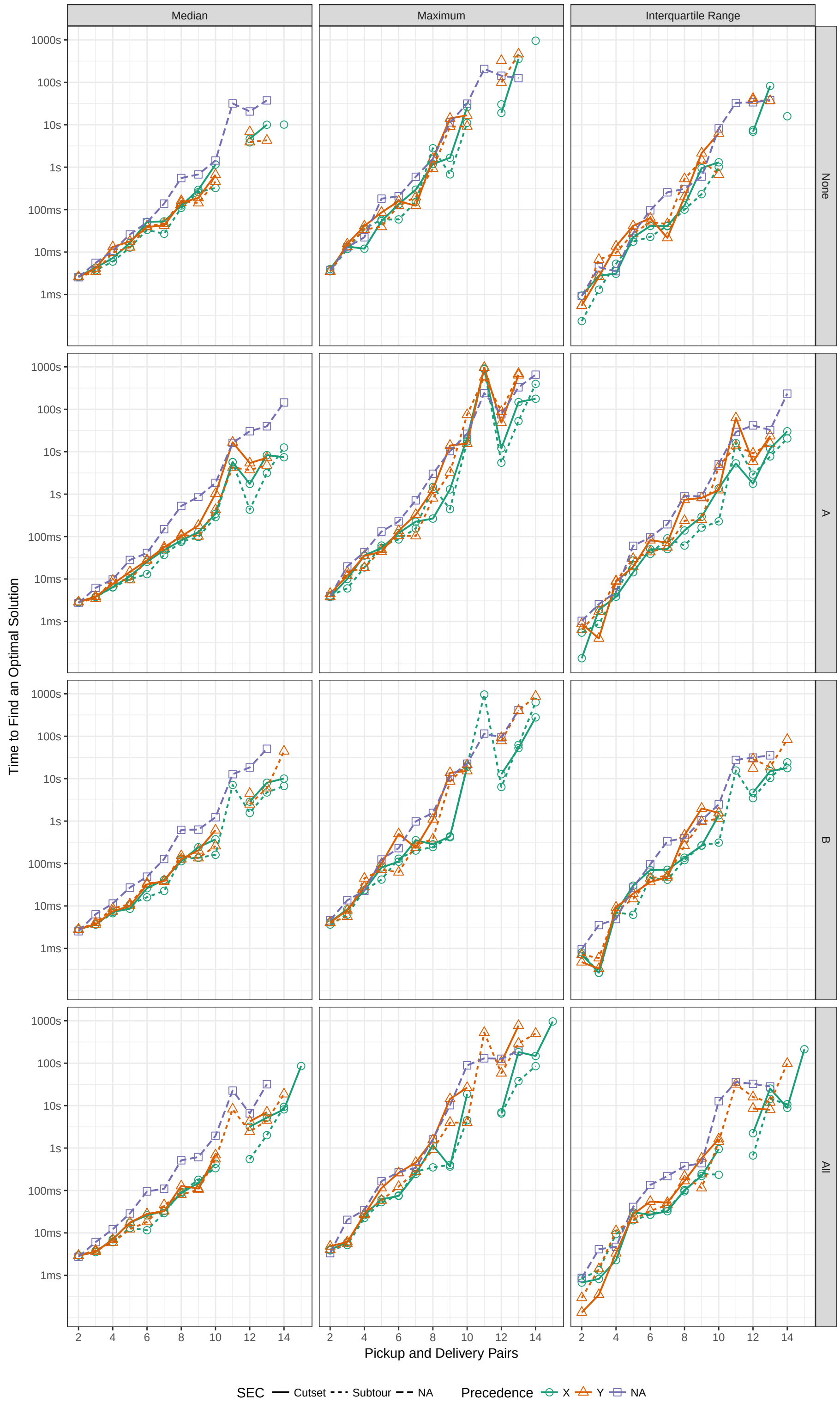

Fig. 21 Time to find an optimal solution for relaxed SSB models with valid inequalities, SEC forms, and precedence forms.

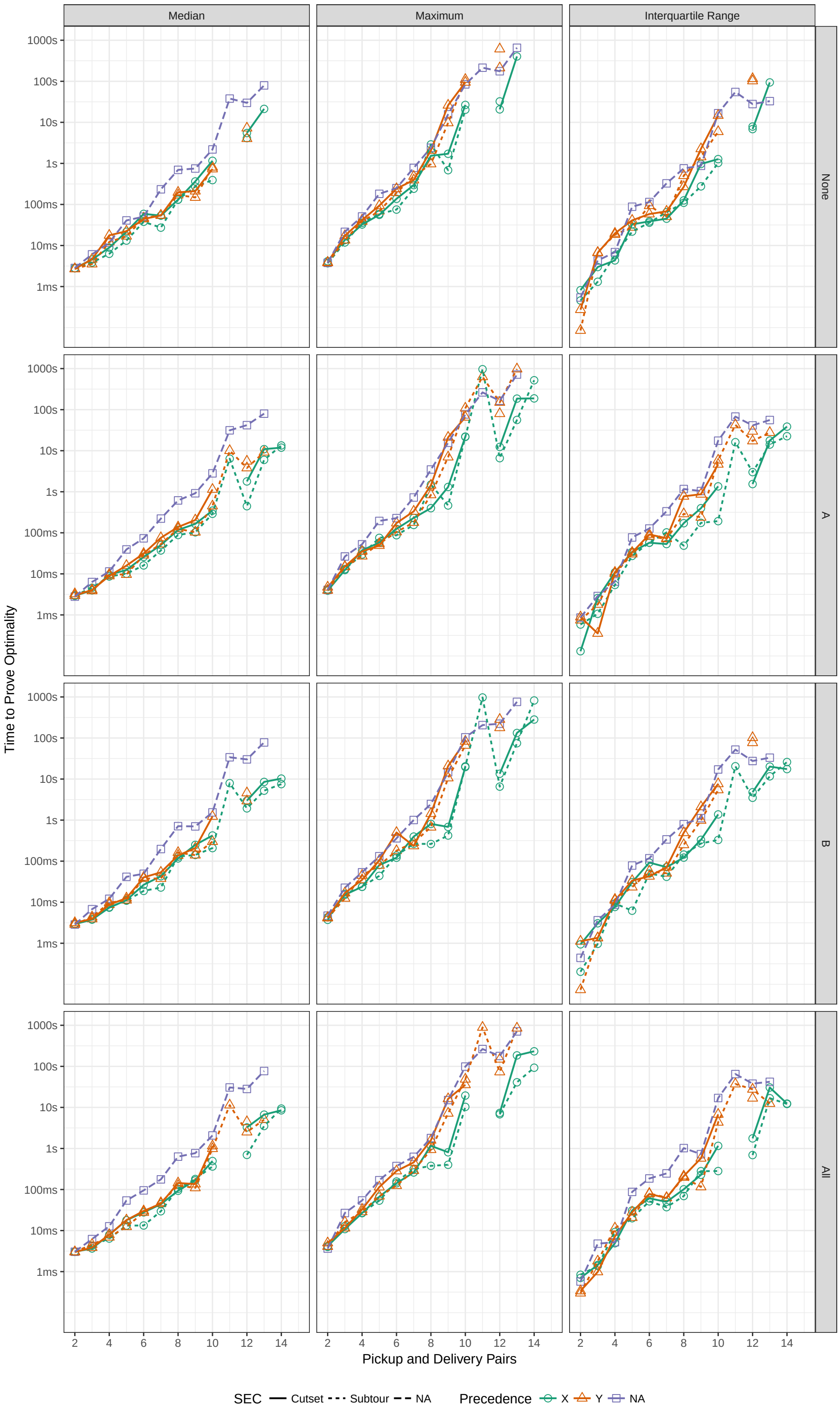

Fig. 22 Time to prove optimality for relaxed SSB models with valid inequalities, SEC forms, and precedence forms.

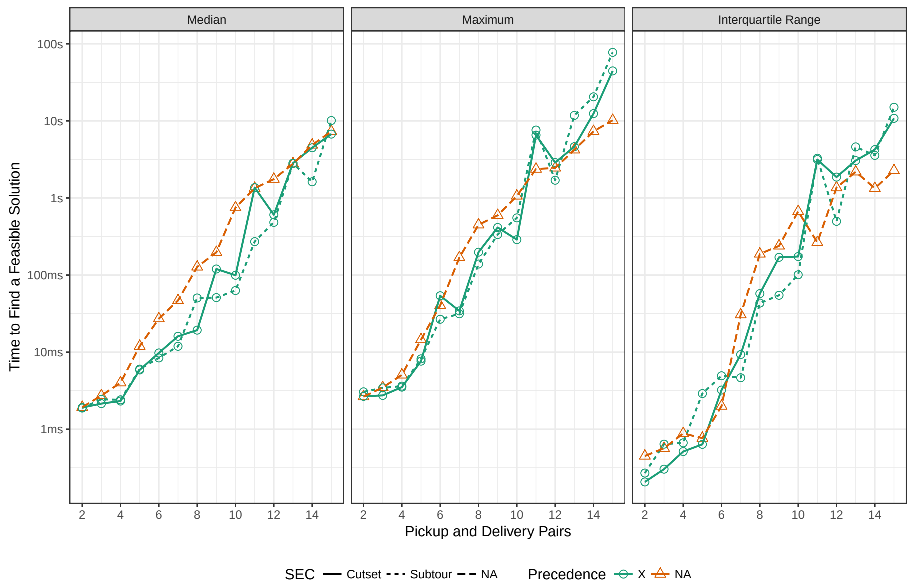

Fig. 23 Time to find a feasible solution for relaxed OH models with SEC and precedence forms.

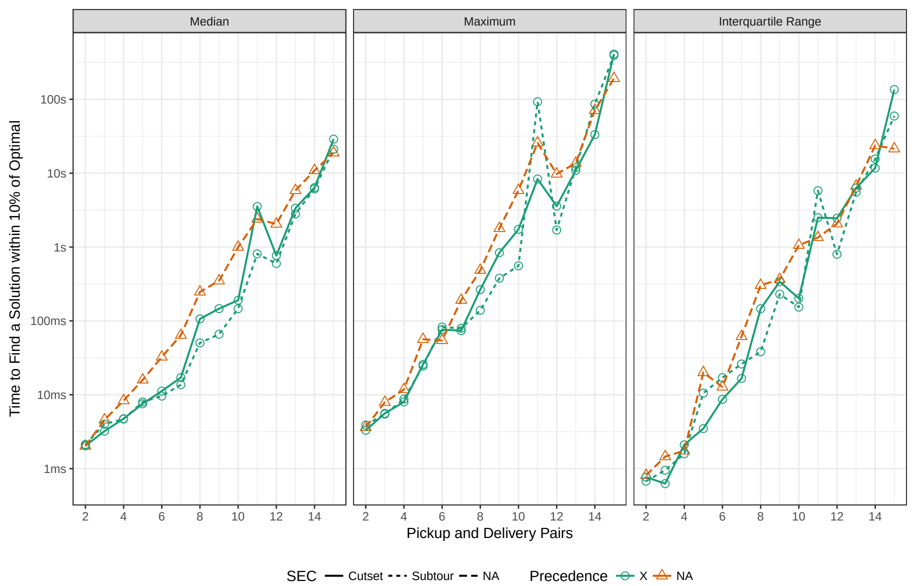

Fig. 24 Time to find a solution within 10% of optimal for relaxed OH models with SEC and precedence forms.

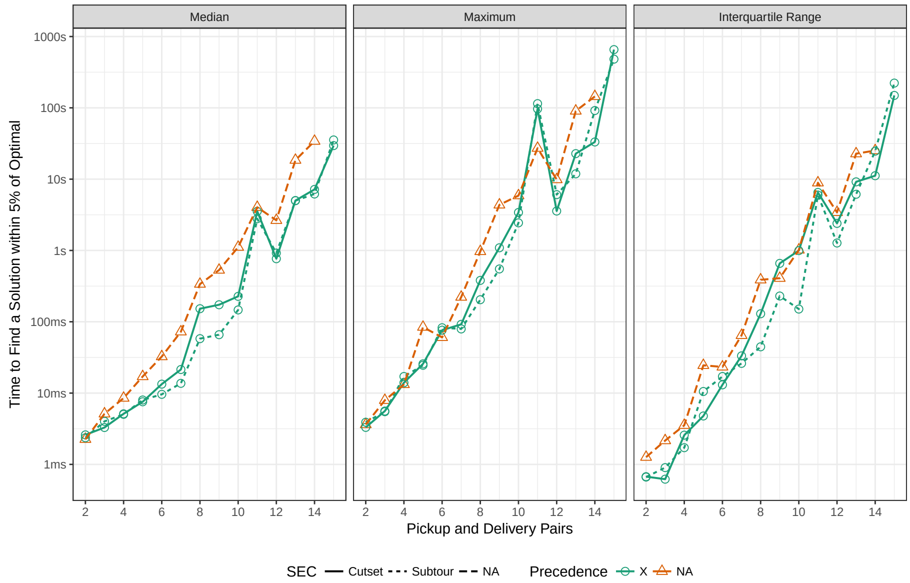

Fig. 25 Time to find a solution within 5% of optimal for relaxed OH models with SEC and precedence forms.

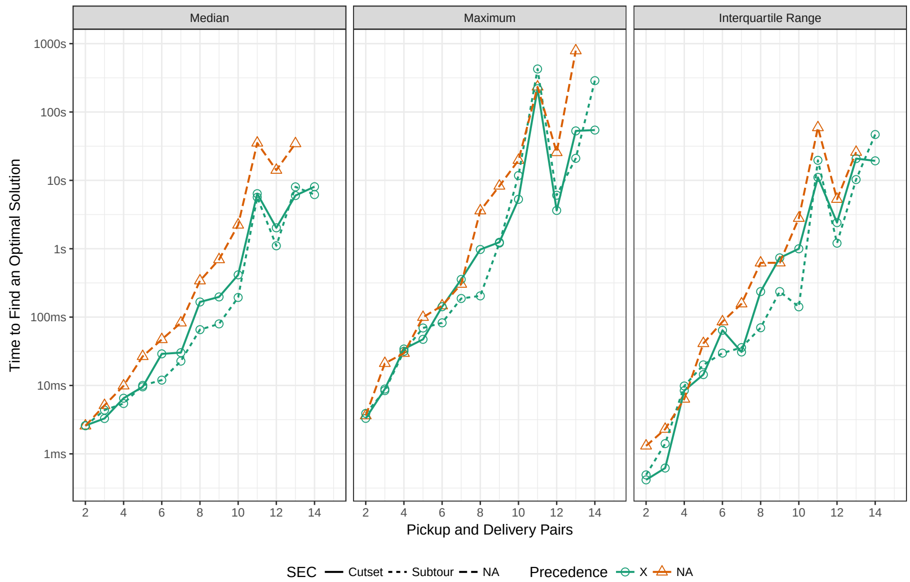

Fig. 26 Time to find an optimal solution for relaxed OH models with SEC and precedence forms.

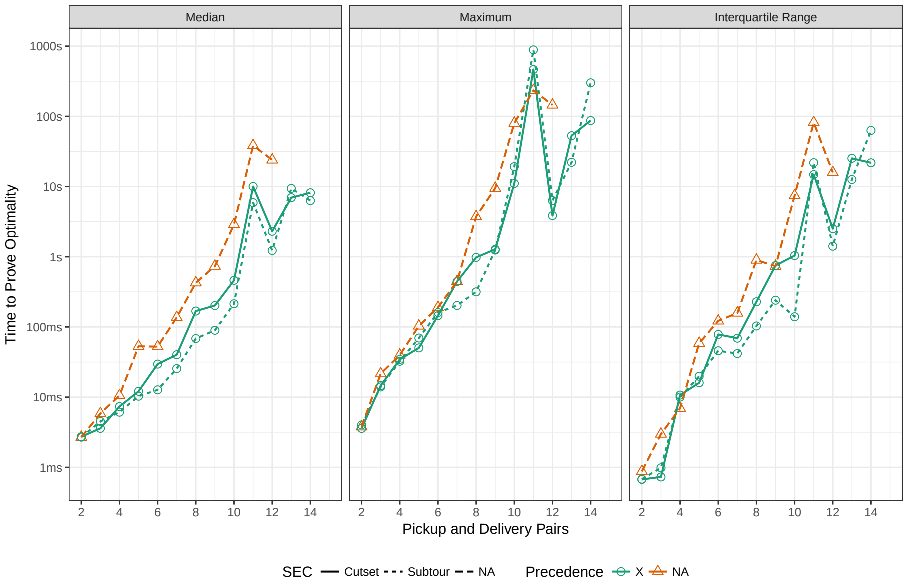

Fig. 27 Time to prove optimality for relaxed OH models with SEC and precedence forms.

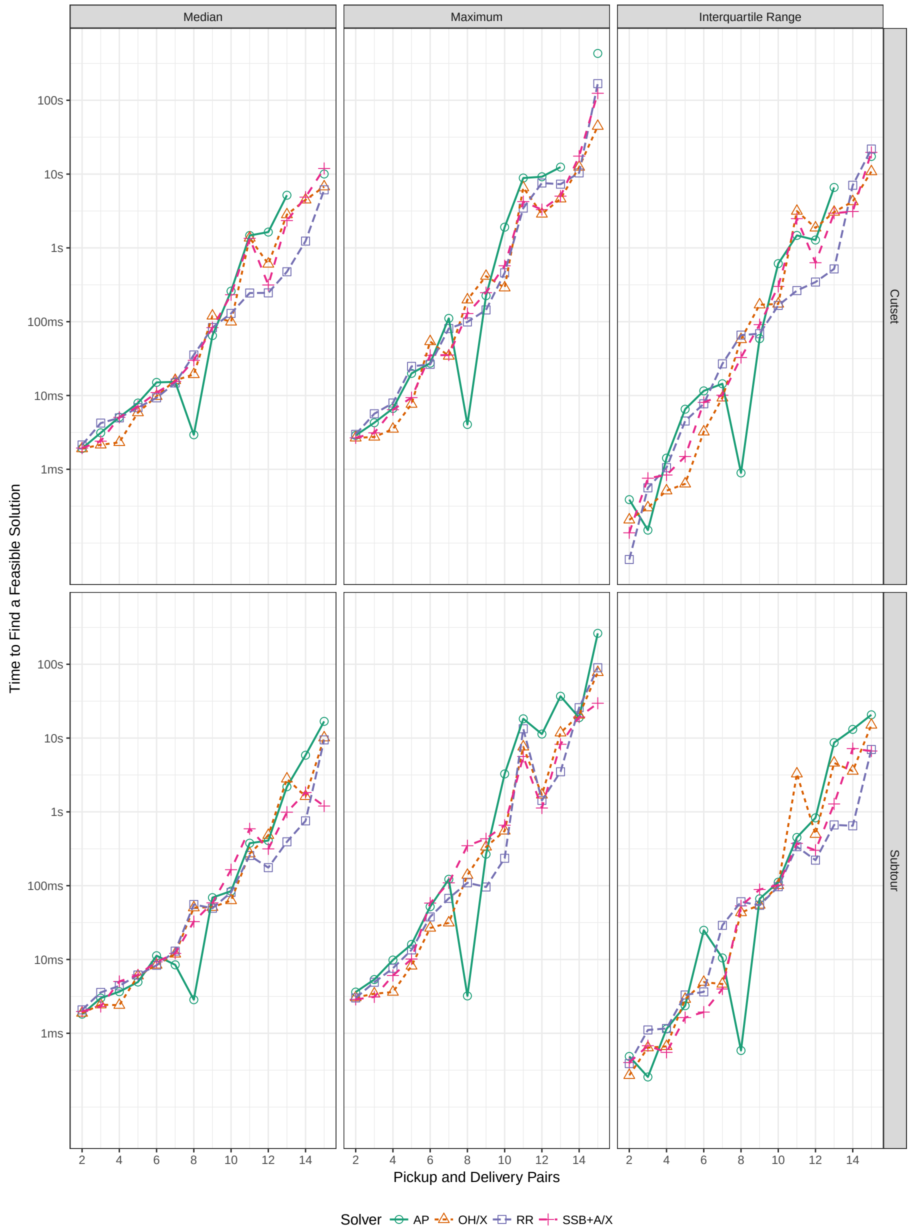

Fig. 28 Time to find a feasible solution for the best models on symmetric instances.

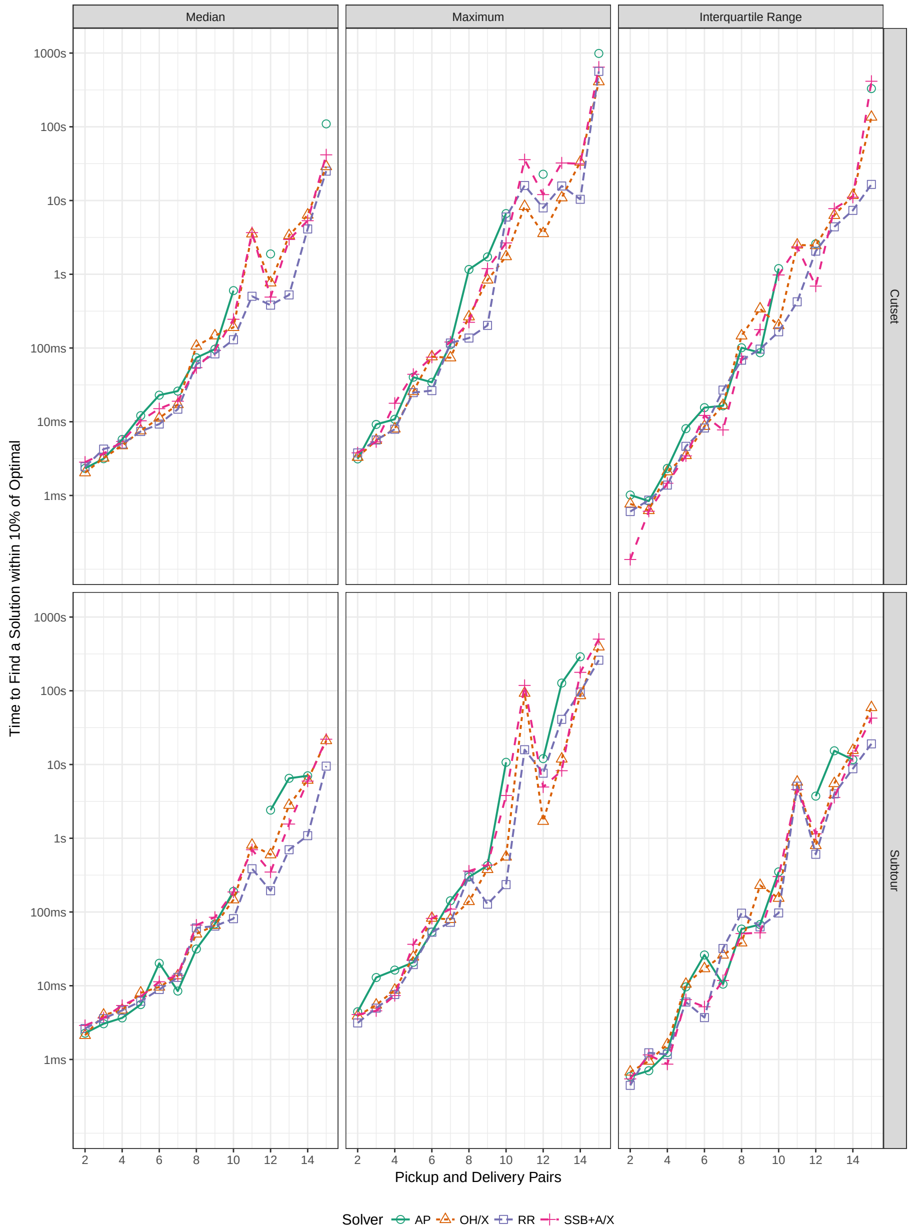

Fig. 29 Time to find a solution within 10% of optimal for the best models on symmetric instances.

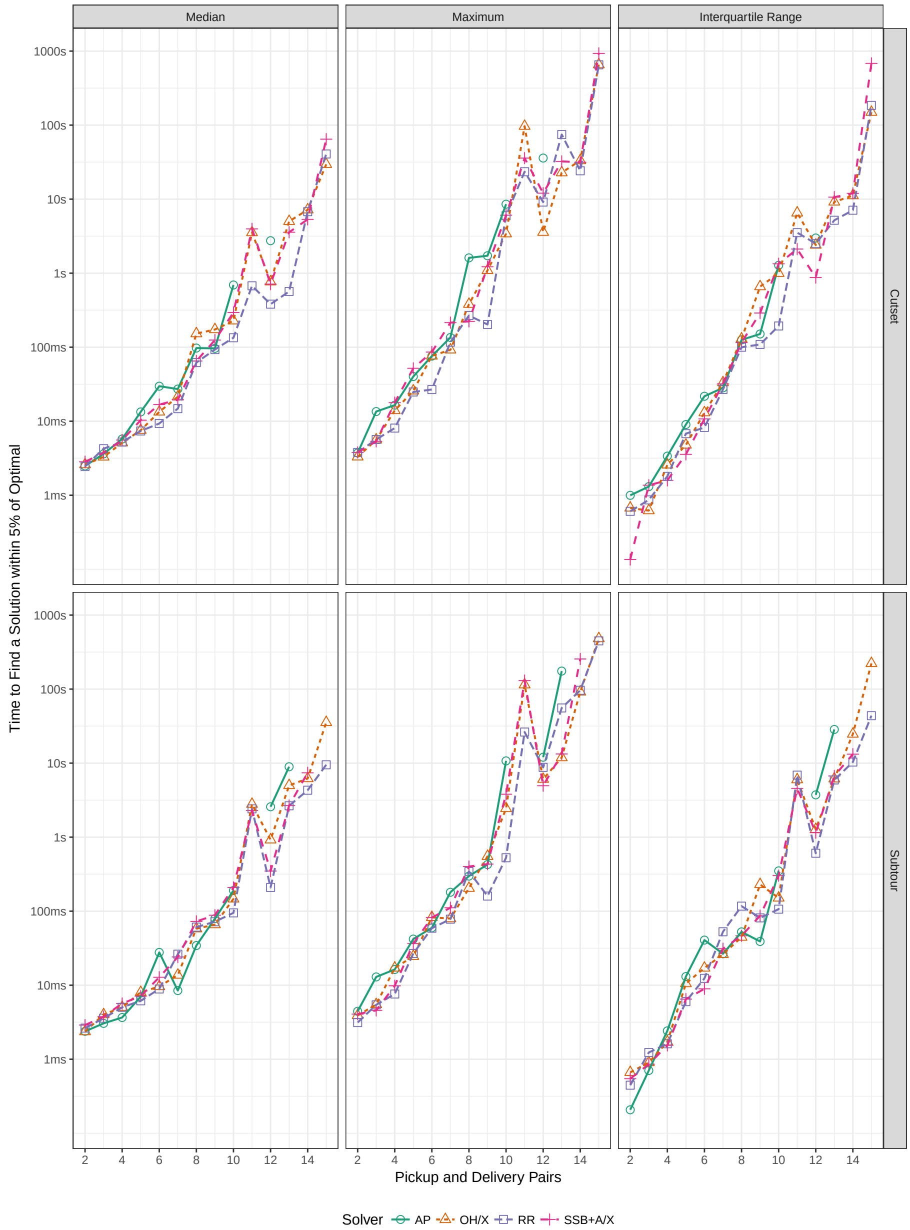

Fig. 30 Time to find a solution within 5% of optimal for the best models on symmetric instances.

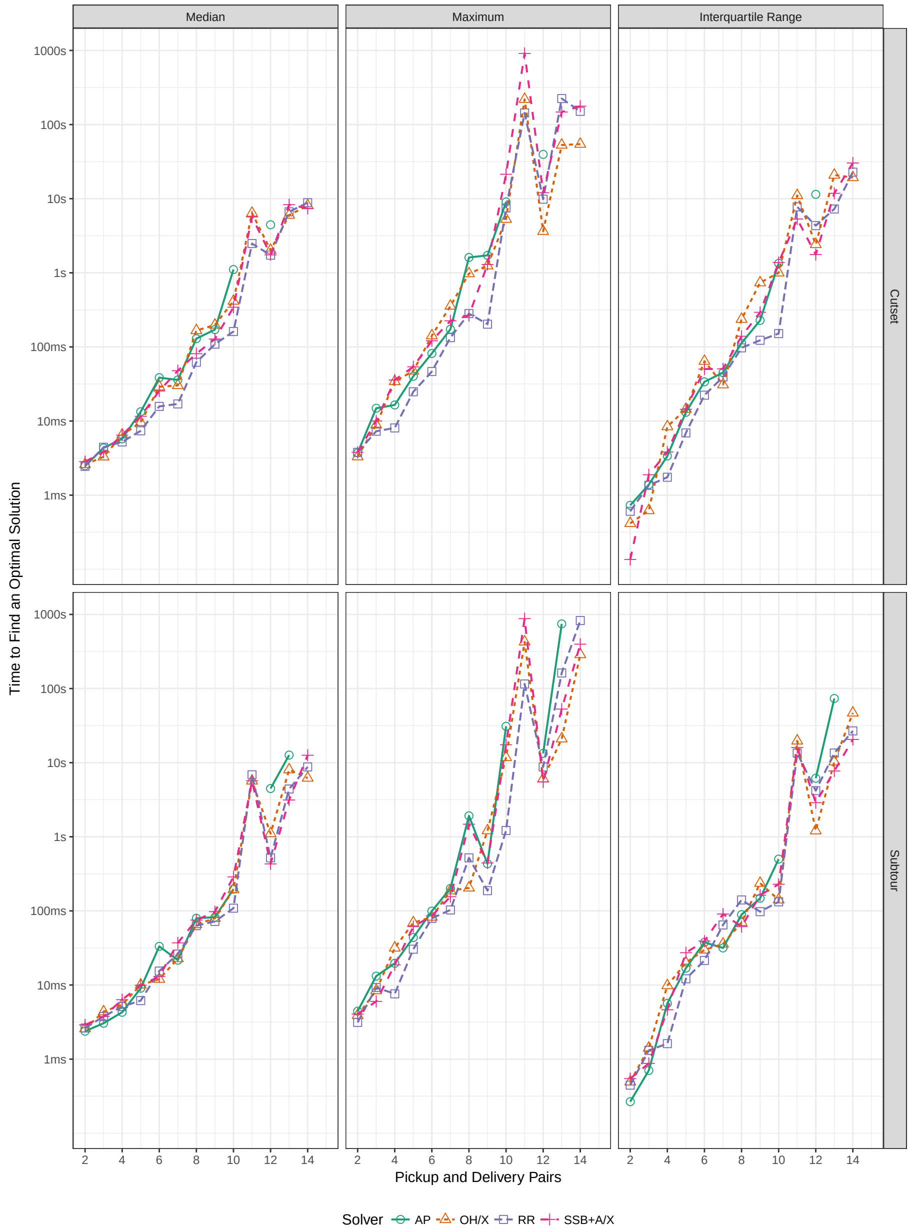

Fig. 31 Time to find an optimal solution for the best models on symmetric instances.

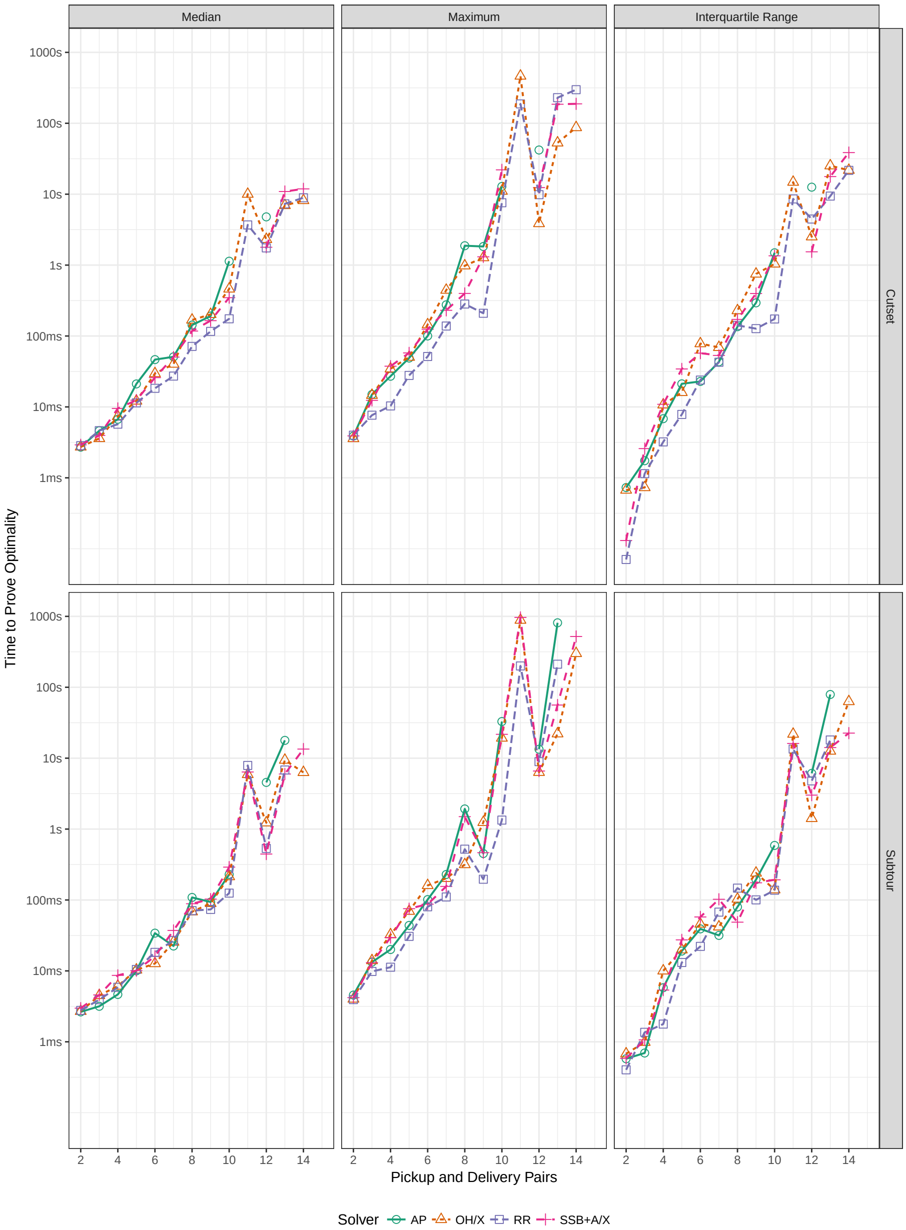

Fig. 32 Time to prove optimality for the best models on symmetric instances.

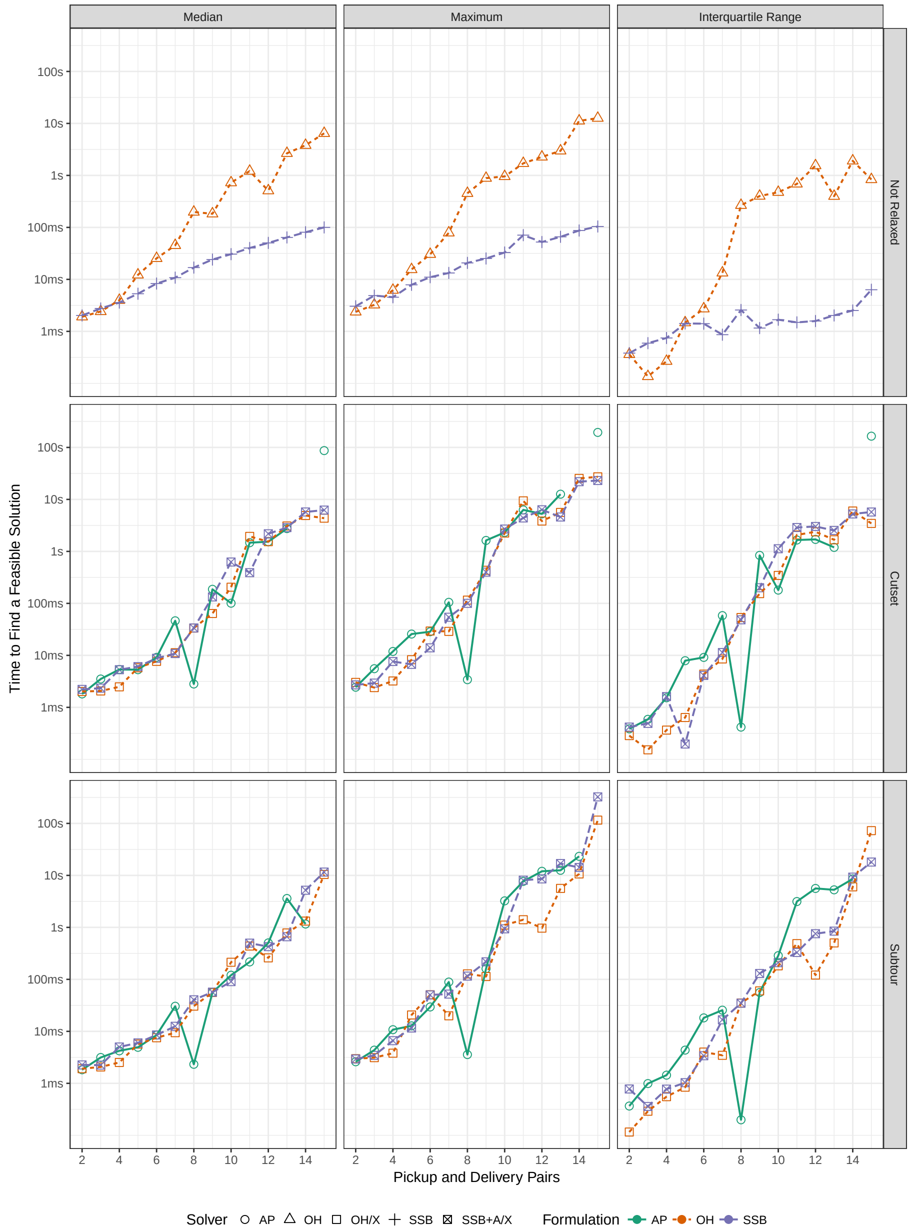

Fig. 33 Time to find a feasible solution for the best models on asymmetric instances.

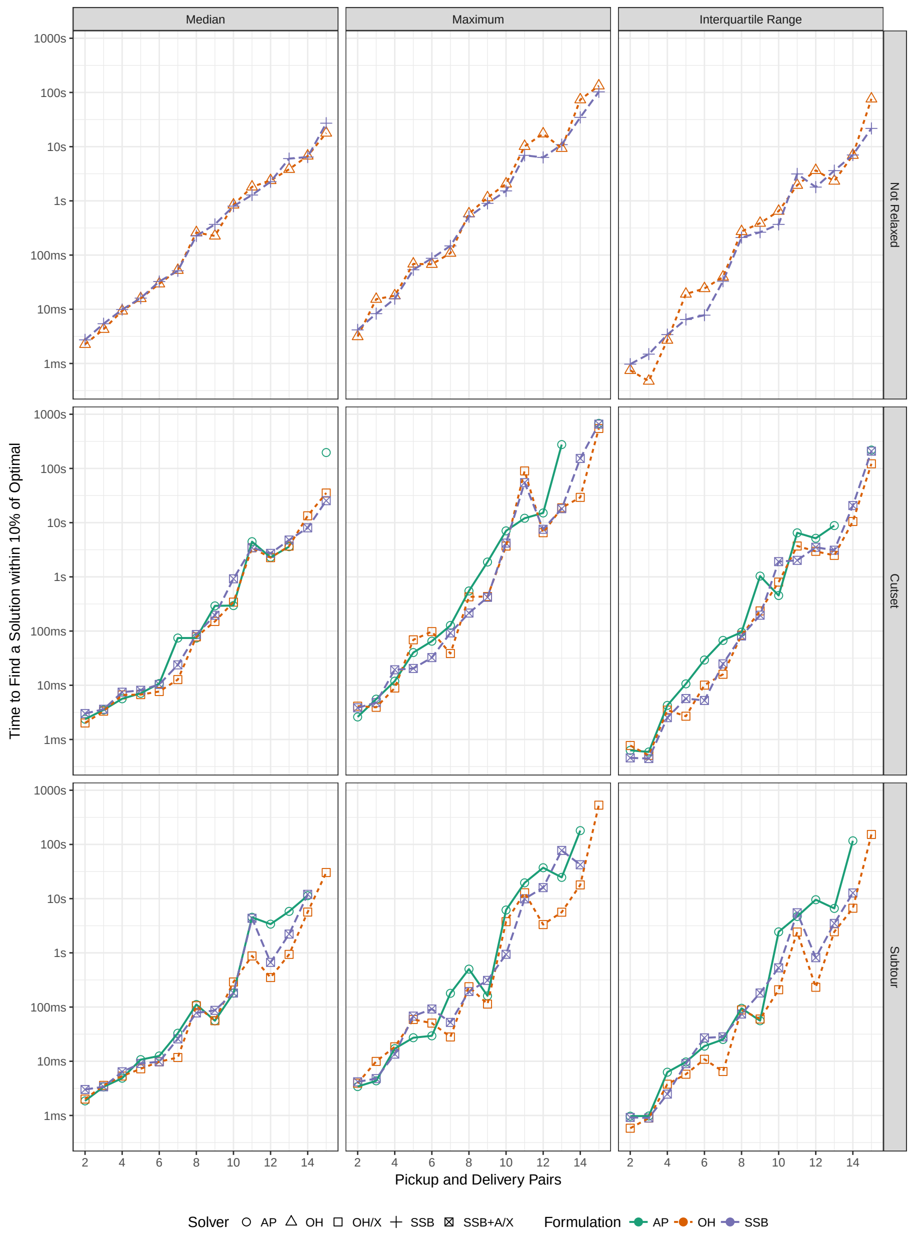

Fig. 34 Time to find a solution within 10% of optimal for the best models on asymmetric instances.

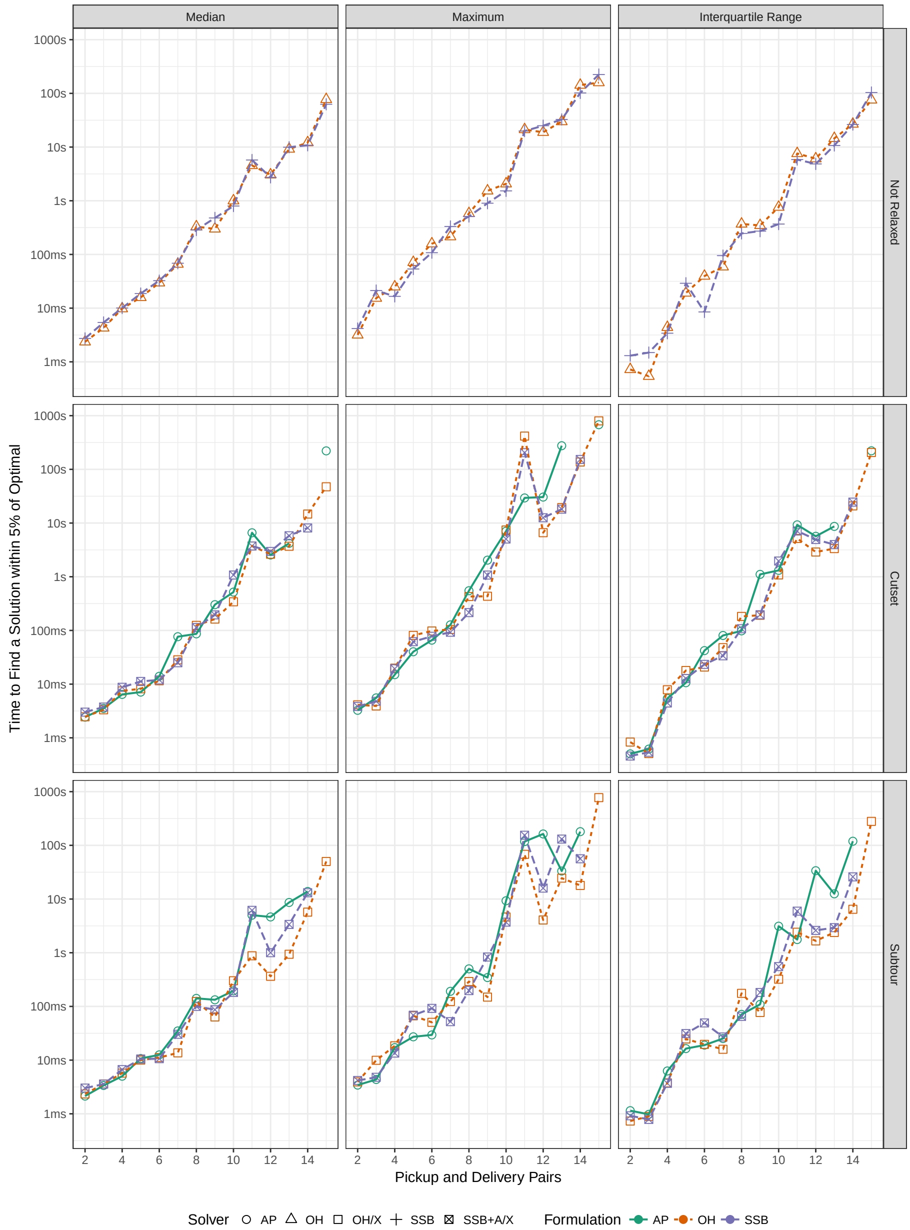

Fig. 35 Time to find a solution within 5% of optimal for the best models on asymmetric instances.

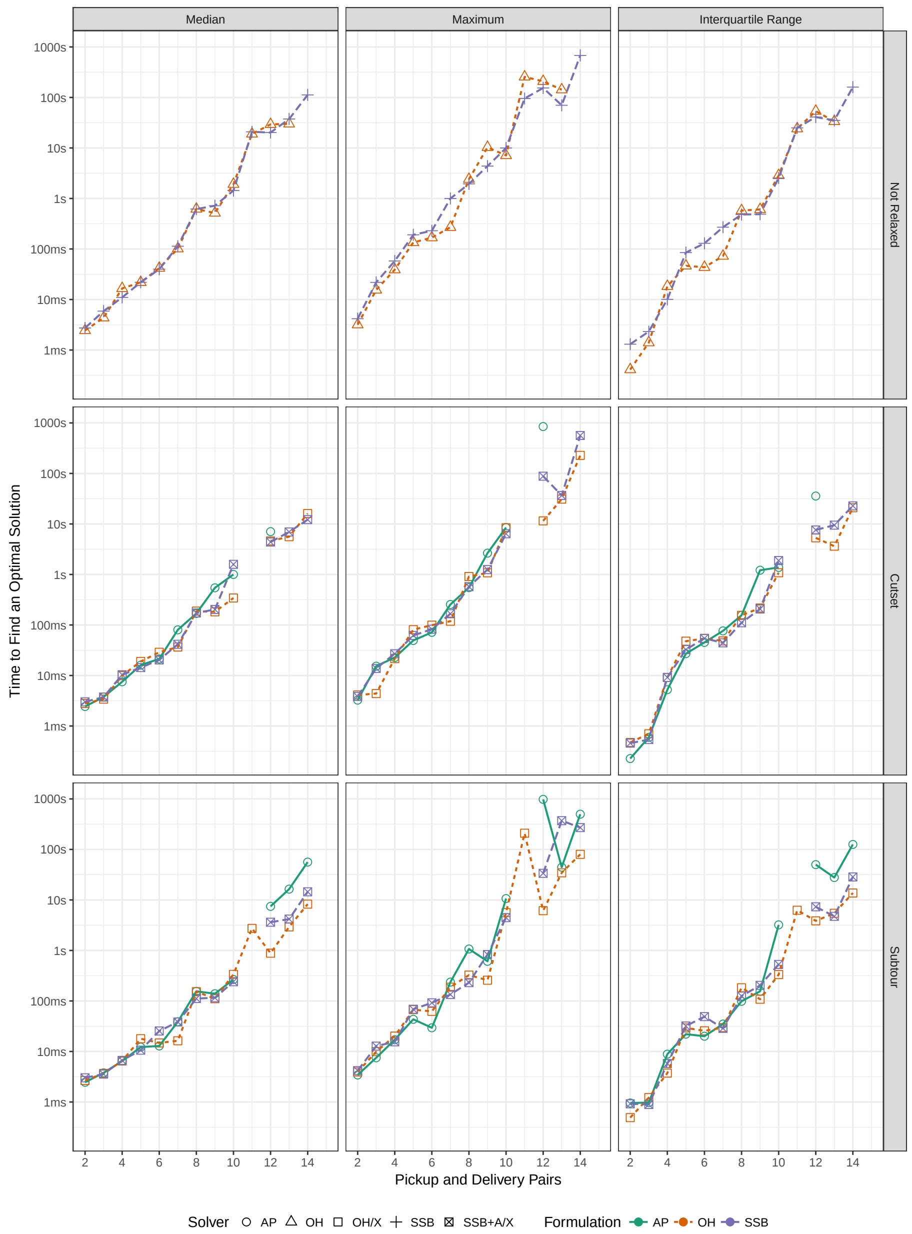

Fig. 36 Time to find an optimal solution for the best models on asymmetric instances.

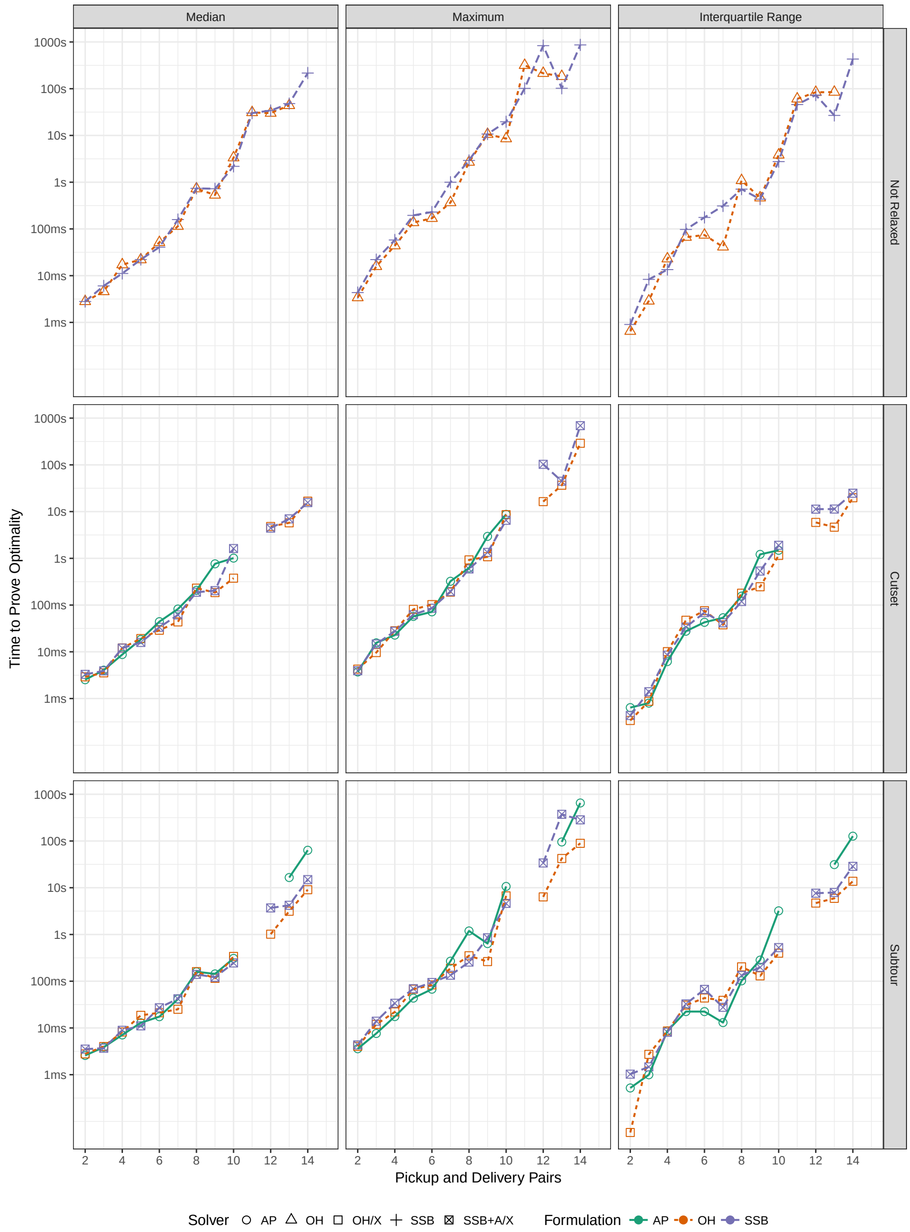

Fig. 37 Time to prove optimality for the best models on asymmetric instances.