In the name of God

Crew Scheduling Problem: A Column Generation Approach Improved by a Genetic Algorithm

Santos and Mateus (2007)

Spring 2009

Instructor: Dr. Masoud Yaghini

Outlines

- Problem Definition
- Modeling As A Set Partitioning Problem
- The Idea of Solution Method
- Column Generation Approach
- Proposed Heuristic Method
- Proposed Genetic Algorithm
- Computational Results
- Conclusions
- References

Problem Definition

Santos and Mateus (2007)

Problem Definition

- This model applied to a crew scheduling problem that arises in a bus company of Belo Horizonte, Brazil.
- On a given day, for each driver is to be assigned a set of trips to be performed.
- **Aim**: assigning trips to the drivers, i.e., assign a feasible set of trips, of which overall cost is minimum.

Santos and Mateus (2007)

Problem Definition

- Belo Horizonte is among the five most populated city in Brazil, with almost 3 million people living in its metropolitan area.
- The public transportation consists mainly of buses.
- There are a lot of different lines, buses that go from a neighborhood to downtown and go back to the same neighborhood.
- Buses go from one region of the city to another, crossing the downtown.

Santos and Mateus (2007)

Problem Definition

- Different companies operate those lines, but the same rules are valid for all of them.
- Just one company controls each bus line, but one company generally has more than one line.
- A duty performed by a driver may contain any number of trips.
- The total work time of a duty is 6:40, plus at most 2 hours of overtime.

Santos and Mateus (2007)

Problem Definition

- Special duty
  - At most one time interval between consecutive trips may have more than 2 hours.
  - During this long interval, changes of stations are allowed, because the driver has enough time to go from one to another station as a passenger.
  - The number of special duties is limited to a given amount or to a percentage, for example 10% of the duties. All special duties above this limit have a cost.
  - The driver works in two “blocks”, with enough interval time between them, so, no rule about minimal rest time.

Santos and Mateus (2007)

Problem Definition

- Regular duty
  - does not have two-hour interval
  - In a regular duty the driver must have at least 30 minutes of rest, adding up all the interval times between trips, and also eventual idle time while performing a trip –long stops at a terminal,
  - Those 30 minutes may be divided along the duty, or after the last trip, but at least one uninterrupted rest of 15 minutes must lie between the trips of the duty.
  - With the 30 minutes, a regular trip has 7:10 hours total work time.

Santos and Mateus (2007)

Problem Definition

- Changes of vehicles are allowed, with a cost.
- Changes of stations are allowed just in the long interval of a special duty.
- And change of lines is allowed if the lines belong to the same group. The group to which each line belongs is part of the input data.
- The cost of a duty depends on the total overtime, total idle time, number of vehicles, stations and line changes.

Santos and Mateus (2007)

A Partial Set of the Trips

Trip	Assigned Vehicle	Start Time	End Time	Depart Station	Arrival Station	Idle time
1	7	365	405	0	3	0
2	7	408	438	3	3	0
3	7	444	484	3	0	0
4	7	650	688	0	3	0
5	7	690	718	3	3	0
6	7	720	748	3	3	0
7	7	750	778	3	3	0
8	7	780	818	3	0	0
9	7	1022	1062	0	3	0
10	7	1068	1098	3	3	0
11	7	1104	1134	3	3	0
12	7	1140	1180	3	0	0
13	8	386	426	0	3	0
14	8	432	462	3	3	0
15	8	468	502	3	0	0
16	8	1034	1074	0	3	0
17	8	1080	1110	3	3	0
18	8	1116	1146	3	3	0
19	8	1155	1195	3	0	0

Santos and Mateus (2007)

Problem Definition

- For each trip, the company has already assigned a vehicle.
- The specific vehicle number is irrelevant in our crew scheduling problem, but if two sequentially trips have different vehicle number, this characterizes a change of vehicle, and the corresponding cost must be applied.
- The start and end time are given in minutes since midnight.
- There are trips departing from the garage (station 0), others finishing there, as well as some that start and end at the same location (station 3 in the example).

Santos and Mateus (2007)

Problem Definition

- None of the trips has idle time, but in some bus lines a trip may have some minutes of idle time.
- This idle time is for example small stops at some stations, whose time is not great enough to change the driver, so the same driver must perform the two legs of the trip.
- But these minutes are counted as part of the imposed 30 minutes of rest.
- Then, the effective work time of a trip may be small than its duration time.

Santos and Mateus (2007)

Modeling As
A Set Partitioning Problem

Santos and Mateus (2007)

Modeling As A Set Partitioning Problem

- The problem can be modeled as a set partitioning problem, where:
  - each element is a trip to be covered and
  - each set contains two or more trips, to be performed by one single driver.

Santos and Mateus (2007)

Modeling As A Set Partitioning Problem

S	the set of duties
n	the number of duties
T	the set of trips to be covered
m	the number of trips
g_j	A binary variable represents a duty (column) $j \in S$, with value 1 if the duty is assigned to a driver and 0 otherwise.
j	the index of duties
c_j	the cost of duty $j \in S$
a_{ij}	Given an m x n matrix A, with $a_{ij} = 1$ if the trip i is covered by the duty j, and 0 otherwise.

Santos and Mateus (2007)

Modeling As A Set Partitioning Problem

- The model (1)-(3) may be used:

$$\min \sum_{j \in S} c_j g_j \qquad (1)$$

$$\sum_{j \in S} a_{ij} g_j = 1 \quad \forall i \in T \qquad (2)$$

$$g_j \in \{0,1\} \quad \forall j \in S \qquad (3)$$

Santos and Mateus (2007)

Modeling As A Set Partitioning Problem

- (1) minimizes the total cost
- (2) assures that each trip is covered by only one duty, i.e., only one driver will perform it
- (3) are the integrality constraints

Santos and Mateus (2007)

The Idea of Solution Method

Santos and Mateus (2007)

The Idea of Solution Method

- The matrix A of our application is generally sparse, then it may be solved by an optimization package in a reasonable amount of time, unless the set of columns is too large.
- Actually this is the case on a real crew-scheduling problem, which number of feasible duties is generally huge, like thousands or millions columns.
- And there are a lot of constraints to build a duty, based on rules, laws, agreements, and so on.
- On the other hand, just a very small subset is selected in the optimal solution.

Santos and Mateus (2007)

The Idea of Solution Method

- It has thousands/millions of columns to select ten/hundreds of them.
- This is the perfect setting to use column generation

Santos and Mateus (2007)

The Idea of Solution Method

- A column generation approach:
  - It starts the problem with a small subset of columns, optimizes it by some method
  - based on the reduced costs of the current solution, generates a new “good” column,
  - adds it to the subset of columns
  - The problem is reoptimized
  - the subset of columns grows as the method is used, until no “good” column can be generated.
  - When there is no such column, the method stops and reports the last solution found.

Santos and Mateus (2007)

The Idea of Solution Method

- The process of generating a new column is done by solving another problem, usually called subproblem,
- The subproblem's objective is to select a column that was not yet added to the master problem.
- The inclusion of the new column into the "master" problem may improve its solution.
- The subproblem is an integer linear programming problem (ILP), using the dual prices of the master problem.

Santos and Mateus (2007)

The Idea of Solution Method

- In this model, the subproblem includes all rules, agreements, and any kind of constraint that states if a duty is feasible to be performed by a driver.
- It takes some time to generate the “best” feasible duty, according to the actual dual prices.
- As any feasible duty with negative reduced cost has a chance to improve the solution, one can generate it using any method, even adding more than one column in each step.
- This work use genetic algorithm to solve the subproblem

Santos and Mateus (2007)

The Idea of Solution Method

- The solution method:
  - At each iteration of the column generation process, an LP solver solves the master problem, sending the dual prices to the genetic algorithm
  - GA solves the subproblem, adding a set of new columns into the master problem. If the genetic algorithm fails in generating good columns, we may always use the exact ILP model to continue the process.
- The results show that this hybrid approach is faster than using just LP and ILP solvers, and it remains exact.

Santos and Mateus (2007)

Column Generation Approach

Santos and Mateus (2007)

Notation

η	the number of special duties allowed without extra cost
D	the set of special duties, D $\subset$ S
k	the number of special duties used besides the η allowed
b	extra cost for k duties

Santos and Mateus (2007)

The Master Problem

- Master problem is the traditional set-partitioning problem, with the additional constraint

$$\min \sum_{j \in S} c_j g_j + bk \qquad (4)$$

$$\sum_{j \in S} a_{ij} g_j = 1, \qquad \forall i \in T \qquad (5)$$

$$\sum_{j \in D} a_{ij} g_j \leq \eta + k \qquad (6)$$

$$g_j \in \{0,1\}, \qquad \forall j \in S \qquad (7)$$

$$k \geq 0 \qquad (8)$$

Santos and Mateus (2007)

Difficulties

- Difficulties:
  - The set S of feasible duties is very big, and solving the problem directly with the whole set may be a difficult task.
  - Besides, a more complex task would be generating all the duties, and just a very small fraction of them take part in the optimal solution.

Santos and Mateus (2007)

Column Generation Approach

- The main idea of the column generation is to work with smaller subsets of *S*, and to use the reduced costs of the current solution to guide the subproblem in the generation of new columns (in this case, new duties).
- Working with smaller problems, and adding one variable at each step, the set-partitioning problem can be quicker solved.
- The main work relies on the subproblem.

Santos and Mateus (2007)

Subproblem

- The subproblem has to generate a feasible duty: a set of trips.
- As each trip has associated a start and an end time, any set of trips has an implicit order
  - the driver will perform them starting with that of earliest start time
  - finish his working day conducting the vehicle in the trip with latest start time
  - performing all the other trips between them

Santos and Mateus (2007)

Subproblem

- The costs used when solving the subproblem:

π_i	the dual price of (5)
μ	the dual price of (6)

- The subproblem model may be built using a directed acyclic graph.

Santos and Mateus (2007)

Subproblem Graph

- The graph $G = (V, A)$ has a set V of $m+2$ vertices:
  - one for each trip $i \in T$,
  - a source vertex s and
  - a target vertex t
- The set A of arcs includes the arcs (s, i) and (i, t) $\forall i \in T$, and an arc (v, w) if trip w can be performed after trip v, that is, if they do not overlap in time, and obey the rules for change of vehicles, rest time, etc.
- Any path from s to t in this graph is a duty of consecutive trips. If this path has an appropriated total time and rest time, it is a feasible duty.

Santos and Mateus (2007)

The Subproblem Model

x_a	a binary variable associated to each arc of G, $a \in A$
y_i	a binary variable associated to each vertex of G (trip), $i \in T$
d_a	the duration time of arc $a \in A$, the duration time of an arc (v, w) is the total time of the trip w plus the interval time between the trips v and w.
l_a	the work time of arc $a \in A$, the work time of an arc (v, w) is the effective work time for trip w, that is, discounting all the idle time that eventually exists between the parts of the trip
f_a	the cost of arc $a \in A$, that includes for example changes of vehicles
h	the total overtime, in minutes

Santos and Mateus (2007)

The Subproblem Model

r	the total rest/idle time, in minutes
q	the variable is used to add a rest time at the end of the duty, it has not the minimum of 30 minutes
p	the binary variable is set to 1 if the path represents a special duty, and 0 for a regular duty
P	The set of all arcs with interval time greater or equal to 2 hours, characterizing a special duty.
Q	this set includes all the arcs with interval between trips of at least 15 minutes

Santos and Mateus (2007)

The Subproblem Model

$$\min \sum_{a \in A} f_a x_a + \alpha h + \beta r \quad - \sum_{i \in V \setminus \{s,t\}} \pi_i^* y_i - \mu^* p \qquad (9)$$

$$\sum_{a \in \delta^+(s)} x_a = \sum_{a \in \delta^-(t)} x_a = 1 \qquad (10)$$

$$\sum_{a \in \delta^+(i)} x_a = \sum_{a \in \delta^-(s)} x_a = y_i, \qquad \forall i \in V \setminus \{s,t\} \qquad (11)$$

$$\sum_{a \in A} (d_a - l_a) x_a + q \geq 30 - 30p \qquad (12)$$

$$\sum_{a \in A} d_a x_a + q \leq 430 - 30p + h \qquad (13)$$

$$\sum_{a \in A} l_a x_a + r \geq 430 - 30p + h \qquad (14)$$

$$h \leq 120 \qquad (15)$$

$$\sum_{a \in P} x_a = p \qquad (16)$$

$$\sum_{a \in Q} x_a \geq 1 \qquad (17)$$

$$h, r, q \geq 0 \qquad (18)$$

$$x_a, y_i, p \in \{0,1\} \qquad \forall a \in A, \forall i \in V \qquad (19)$$

Santos and Mateus (2007)

The Subproblem Model

- An optimal solution of this ILP is a path from vertex s to t, passing though the vertices i for which $y_i = 1$, using the arcs $a = (v, w)$ whose corresponding $x_a = 1$.
- (9) the function minimizes the cost of the duty,
  - Adding up:
    - the cost of performing consecutively the trips
    - the costs of overtime
    - the costs of idle time
  - Dropping:
    - the reduced costs of the trips covered by the duty
    - the reduced cost of the special duty, it is a special duty

Santos and Mateus (2007)

The Subproblem Model

- (10) The set of constraints assure that there is one arc leaving vertex *s* and one arc reaching vertex *t*, beginning and ending the path appropriately.
- (11) The set of constraints binds the *x* and *y* variables:
  - if the duty includes trip i, then $y_i = 1$, and there must be one arc leaving and one reaching the vertex i
  - if, otherwise, the trip i is not covered by the duty, $y_i = 0$, and no arcs leaves nor reaches vertex i.
- (16) By this we know if the path represents a special duty: the variable *p* is set to 1 if any arc in *P* is chosen.

Santos and Mateus (2007)

The Subproblem Model

- In case of a special duty, the term –30p in (13) and (14) corrects the total time, that should be 400 (6 hr 40 min), instead of 430 minutes (7 hr 10 min).
- (12) This Constraint counts the total rest time, and if it is not at least 30 minutes (in case of a regular duty), q holds the difference, that is added in (13)
- (13) calculates h, the overtime
- (14) give us the total idle time r
- (15) limits the overtime to 2 hours
- (17) assures at least one interval of 15 minutes in the duty

Santos and Mateus (2007)

The Subproblem Model

- Alternative solution methods of subproblem:
  - the exact integer solution method
  - an heuristic based on its linear relaxation
  - a genetic algorithm

Santos and Mateus (2007)

Proposed Heuristic Method

Santos and Mateus (2007)

Proposed Heuristic Method

- The experiments showed that, for some data, the integer solution of the subproblem model could not be fast obtained by the optimizer packages, when used to generate the first columns.
- Its linear relaxation, instead, is quickly solved.
- The heuristic tries to obtain a feasible duty based on the optimal linear solution of the model.

Santos and Mateus (2007)

Proposed Heuristic Method

- In a typical linear solution of (9)-(19) there will be several fractional paths, instead of one path.
- The main idea is to choose the fractional path with the highest value in the x variables, which would probably yields a good duty.
- Maybe it was not integral because it is not feasible to include all the trips, or there is a path with similar cost.

Santos and Mateus (2007)

Proposed Heuristic Method

- The Heuristic Method

1. Solve the linear relaxation of model (9)-(19)
2. Current vertex $v = s$
3. New duty $D = \{\}$
4. $k = \{w \mid x_{(v,w)} \geq x_{(v,t)}, \forall (v,i) \in A\}$
5. if $k = t$, then stop, end of the duty
6. else

   6.1 if $D \cup \{k\}$ is feasible: $D = D \cup \{k\}$, $v = k$

   6.2 else: $x_{(v,\ w)} = 0$
7. Go back to step 4

Santos and Mateus (2007)

Proposed Heuristic Method

- Step 4 chooses the next vertex to be included in the path, representing the next trip to be in the duty. It chooses the one with the highest value of the arc variable coming from the last vertex included.
- If the duty remains feasible the corresponding trip is included in the duty (step 6.1), and the algorithm chooses another trip following this one.
- If, instead, the duty becomes infeasible with the inclusion of this trip, the corresponding variable is set to zero, avoiding choosing this arc again (step 6.2).

Santos and Mateus (2007)

Proposed Heuristic Method

- As the size of a duty is generally very small, the steps (2)-(7) of the heuristic are very fast.
- But we still have to solve the linear relaxation of the subproblem model.
- And each time it is solved, only one new duty is added to the master problem.
- These drawbacks give the opportunity to use a totally different approach: a genetic algorithm to solve the subproblem, building not one but a set of duties without using the ILP model (9)-(19).

Santos and Mateus (2007)

Proposed Genetic Algorithm

Santos and Mateus (2007)

Proposed Genetic Algorithm

- Proposed genetic algorithm components:
  - Chromosome
  - Selection
  - Crossover
  - Mutation
  - Replacement
  - External Population

Santos and Mateus (2007)

Chromosome

- A chromosome is an ordered set of integers, representing the trips of a duty.
- The trips may be previously ordered by start time, and then a feasible duty always gives a chromosome with a set of increasing numbers.
- During the whole algorithm, the trips inside a chromosome will never overlap in time, but it may violate some other constraints, like:
  - overtime above the limit,
  - forbidden changes of vehicles,
  - etc.

Santos and Mateus (2007)

Chromosome

- This facilitates the crossover operator, turning possible the combination of almost any pair of chromosomes, increasing the diversity of the population.
- However the fitness function includes some penalties for this kind of chromosomes, avoiding their continuation in the final population.

Santos and Mateus (2007)

Chromosome

- The fitness value of each chromosome is:
  - the cost of the duty it represents,
  - discounting the dual prices of the trips it includes
  - plus some penalties for disobeying a constraint.
- The greater the dual price provided by the master problem, smaller is the cost of the chromosome.

Santos and Mateus (2007)

Selection

- Chromosomes from the current population are randomly selected creating a new population.

Santos and Mateus (2007)

Crossover

- Two chromosomes, A and B, are randomly selected from the current population.
- Let the trips of:
  - chromosome A: A1, A2, …, Aa
  - chromosome B: B1, B2, …, Bb
- A random trip Ai , $1 < i \leq a$, is chosen from chromosome A.
- The first trip Bj in B with start time greater than the final time of Ai is selected.

Santos and Mateus (2007)

Crossover

- The trips are interchanged, and the final chromosomes have the trips:
  - A1, …, Ai, Bj, …, Bb
  - B1, …, Bj-1, Ai+1, …, Aa
- The trips of the first chromosome for sure do not overlap, as the trips are ordered, and the start time of Bj is greater than the end time of Ai.
- In the second chromosome, if Bj-1and Ai+1 overlaps in time, one of them is dropped.
- In any case, if the second chromosome has just one trip, than the original B chromosome is kept.

Santos and Mateus (2007)

Mutation

- Suppose A is a chromosome to be mutated.
- One of the trips A1, A2, …, Aa of A is randomly selected, lets say Ai.
- The whole set T of trips is scanned, and if there is a trip X whose start time is greater than the end time of trip Ai-1, and whose end time is earlier than the start time of trip Ai+1, the trip Ai of A is substituted by X.
- The mutated chromosome is A1, …, Ai-1, X, Ai+1…, Aa.
- A mutation step is used to avoid premature convergence, bringing new trips in the process.

Santos and Mateus (2007)

Replacement

- The new offspring is selected using a roulette rank.
- For each chromosome is given a part in a roulette, according to its fitness value.
- The lower the fitness the larger the part in the roulette, the higher the probability to be selected.
- The chromosomes with trips of greater dual prices are more likely to be chosen, and to survive for the next generation.

Santos and Mateus (2007)

External Population

- There is an external population that keeps the best solutions found.
- The size of this population is fixed, and it is updated after each cycle of crossover, mutation, and replacement of the genetic algorithm.
- At the end of the genetic algorithm, the chromosomes within the external population that corresponds to feasible duties, and with negative reduced cost, are added to the master problem.

Santos and Mateus (2007)

Parameters

- The population size: 3 times the number of trips to be covered
- The external population size: 10
- The crossover probability: 90%
- The mutation probability: 10%
  - The mutation probability is higher than usual because we need variety on the set of duties generated by the genetic algorithm, not just a single solution.
- Penalties:
  - A penalty for more than one interval of 2 hours between trips
  - A penalty for exceeding the allowed overtime.

Santos and Mateus (2007)

Computational Results

Santos and Mateus (2007)

Computational Results

- All algorithms were coded in C programming language, using the Xpress-MP Builder Component Library to model and to solve the LP and ILP models.
- The tests were run on a 3.2Ghz Asus notebook, with 512Mb memory.

Computational Results

- Input data used to test the algorithms

Data	Bus Line	# Trips	# Vehicles	Average Work Time (in minutes)
A	321	19	2	34,4
B	1170	23	2	65,2
C	2152	43	6	110,5
D	4150	63	9	142,0
E	5201	75	12	155,7
F	2104	94	17	163,0
G	8207	101	18	154,6
H	2152 + 4150	106	15	129,2
I	4150 + 5201	138	21	149,5

Santos and Mateus (2007)

Computational Results

- The details of data A

Trip	Assigned Vehicle	Start Time	End Time	Depart Station	Arrival Station	Idle time
1	7	365	405	0	3	0
2	7	408	438	3	3	0
3	7	444	484	3	0	0
4	7	650	688	0	3	0
5	7	690	718	3	3	0
6	7	720	748	3	3	0
7	7	750	778	3	3	0
8	7	780	818	3	0	0
9	7	1022	1062	0	3	0
10	7	1068	1098	3	3	0
11	7	1104	1134	3	3	0
12	7	1140	1180	3	0	0
13	8	386	426	0	3	0
14	8	432	462	3	3	0
15	8	468	502	3	0	0
16	8	1034	1074	0	3	0
17	8	1080	1110	3	3	0
18	8	1116	1146	3	3	0
19	8	1155	1195	3	0	0

Santos and Mateus (2007)

Computational Results

- The difficult of the problem increases with the number of trips, and with the number of trips that can be part of a duty.
- The lower average work time of the trips, the greater is the number of trips that may be part of a duty, increasing the combinatorial number of feasible duties.
- The instances H and I are combination of instances C and D, and instances D and E, respectively.

Santos and Mateus (2007)

Computational Results

- Some computational results:

	Exact ILP model		Linear Relaxation		Genetic Algorithm	
Data	#Duties	Time (s)	#Duties	Time (s)	#Duties	Time (s)
A	137	2,55	113	1,25	205	1,78
B	144	19,7	255	19,6	386	18,2
C	221	76,6	265	37,3	469	37,7
D	168	490,4	172	354,2	507	71,0
E	190	1277,6	256	671,2	782	201,3
F	304	1259,5	415	554,97	867	394,8
G	?	> 1 hour	421	1409,4	1141	603,3
H	492	1692,0	512	1246,0	1009	637,3
I	?	> 1 hour	?	> 1 hour	1121	953,2

Santos and Mateus (2007)

Computational Results

- The table shows the total number of duties inserted in the master problem (numbers of columns generated), and the time spent to reach the solution.
- The execution was interrupted after 1 hour.

Santos and Mateus (2007)

Methods Comparisons

- Effectiveness (solution value): The solution value is not reported
  - because the column generation algorithm reaches the same solution with all heuristics
  - whenever the heuristic fails to generate a new column with negative reduced cost, the ILP method is called.
  - So, the optimal solution is always assured.

Santos and Mateus (2007)

Methods Comparisons

- The number of duties are generated in the methods:
  - Exact ILP model
    - This approach generates a lower number of duties.
    - It gets the best column to be added in the master problem.
  - Heuristic method
    - It adding more duties.
  - Genetic algorithm
    - It generates twice or more the number of duties, in comparing to the heuristic method.
    - Some of duties may be useless, it can add more than one good duty a time

Santos and Mateus (2007)

Methods Comparisons

- The efficiency of methods (speed)
  - Exact ILP model
    - It takes higher than two other methods
    - For the instance G & I, the execution was interrupted after 1 hour.
  - Heuristic method
    - In this method the subproblem is called more times, the number of iterations is higher, but as they are faster then solving the ILP model, the total time is generally lower.
    - For some instances the time is almost half the time spent when only the exact ILP method is used.

Santos and Mateus (2007)

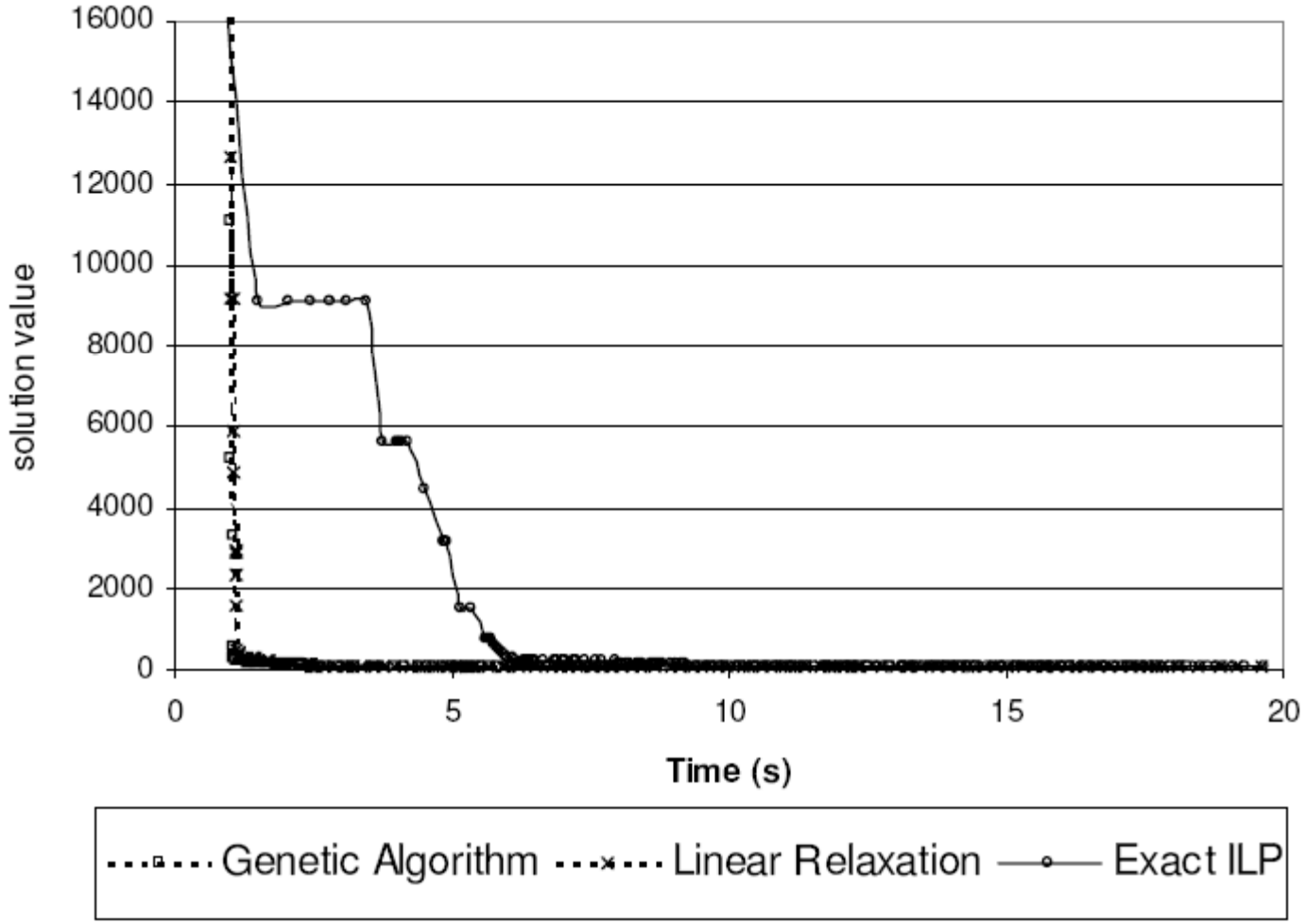
Methods Comparisons

- The efficiency of methods (speed)
  - Genetic algorithm
    - The genetic algorithm improves the column generation even spent lower time
    - It takes almost the same time as the heuristic when used for small instances, but half or even lower time for the big instances.

Santos and Mateus (2007)

Convergence

- Convergence of the solution value for input data B:



Santos and Mateus (2007)

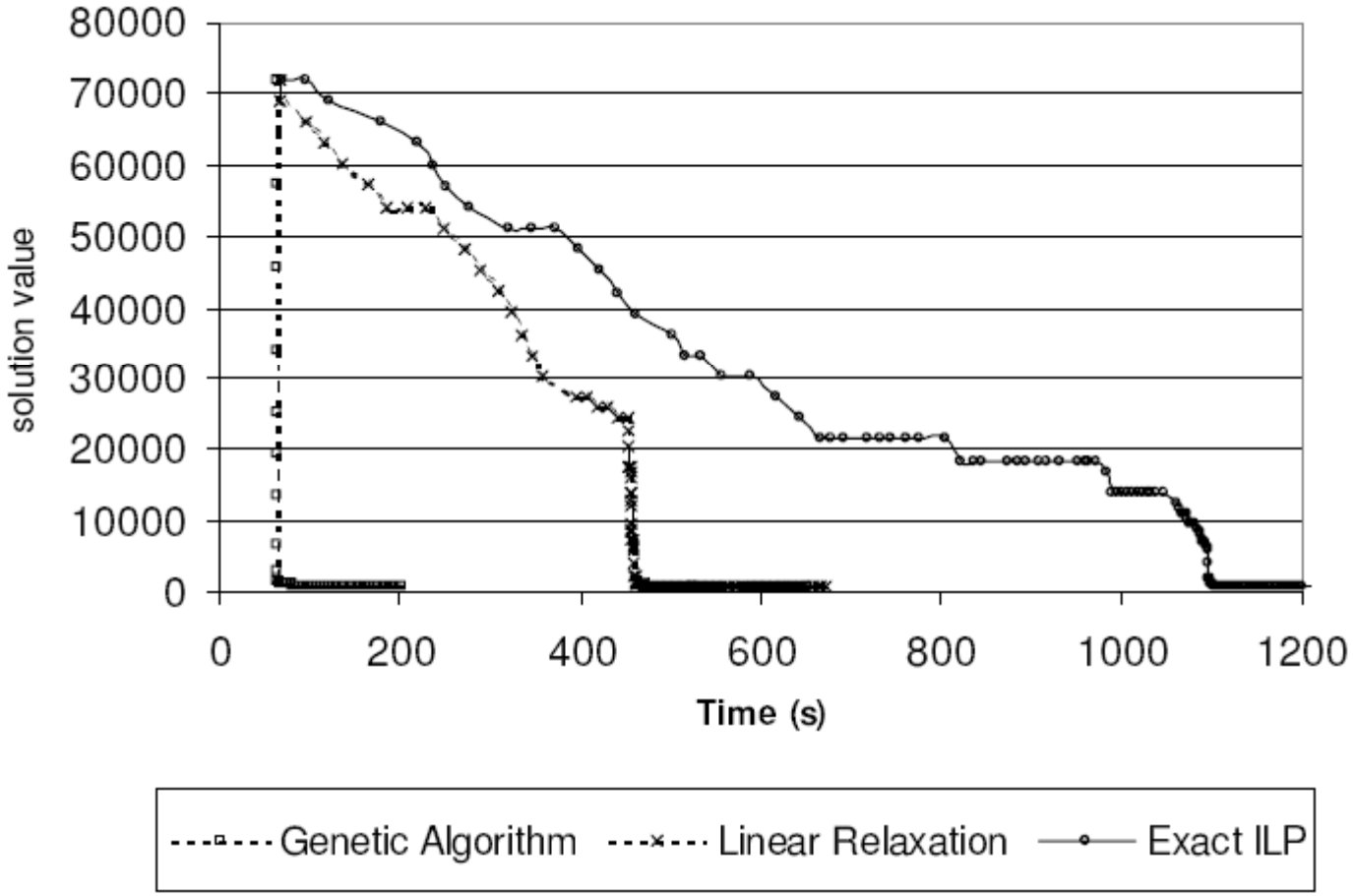
Convergence

- Convergence of the solution value for input data B:
  - These data were chosen because all three algorithms spent almost the same time to find the optimal solution.
  - The heuristic and the genetic algorithm have the same behavior, their curves overlap in the figure.
  - The important behavior of the methods in this figure is what happens in the beginning of the algorithm.
  - Solving the subproblem using the genetic algorithm or the heuristic based on the linear relaxation, the master problem has a substantial improvement in its solution value in less than 2 seconds.
  - Using the exact ILP, instead, it spends almost 10 seconds to reach the same level.

Santos and Mateus (2007)

Convergence

- The convergence of solution values for input data E:



Santos and Mateus (2007)

Convergence

- The convergence of solution values for input data E:
  - Speed of methods reach the optimal solution:
    - Genetic algorithm: 200 seconds
    - Heuristic method: 670 seconds
    - Exact ILP model: 1280 seconds
  - Although the genetic algorithm generates more columns, many of them useless, it quickly adds the best columns that take part of the optimal solution.

Santos and Mateus (2007)

Convergence

- The convergence of solution values for input data I:
  - Imposing a time limit of one hour, the exact column generation algorithm & heuristic method could not find the solution
  - But genetic algorithm takes about 15 minutes to reach the solution
  - The genetic algorithm could not generate all the duties, from the 1121 duties included in the master problem, 220 were generated by the ILP method, in the iterations that the genetic could not find a good duty.
  - This happens more often at the end of the algorithm, when the solution value is already near the best value.

Santos and Mateus (2007)

Conclusions

Santos and Mateus (2007)

Conclusions

- The genetic algorithm is used to solve the subproblem of the column generation algorithm, accelerating the process, but the whole algorithm still remains exact.
- Integration of genetic algorithms and column generation, in the way it is done in this work, could not be found in the literature.

Santos and Mateus (2007)

Future Works

- The results show that it works very well, opening the way for at least two future works:
  - (i) improve the genetic algorithm
    - Using special operators created for similar problems, which could generate better duties each time, decreasing the total number of duties, and then the execution time.
    - Improving the genetic algorithm may also decrease the number of times that the exact method is called.
  - (ii) apply the same idea in other combinatorial problems
  - (iii) improve column generation with other metaheuristic algorithms
    - Using ant colony optimization, tabu search and etc.

References

Santos and Mateus (2007)

References

- A. G. Santos, G. R. Mateus, **Crew scheduling urban problem: an exact column generation approach improved by a genetic algorithm,** IEEE Congress on Evolutionary Computation, 1725-1731, 2007.

The end