Lower bound computations for routing problems in LocalSolver

Alexandre Bontems
LocalSolver, 36 avenue Hoche, 75008 Paris
e-mail : abontems@localsolver.com

1 Introduction

LocalSolver is a model-and-run mathematical optimization solver combining different operations research techniques [1]. Problems can be modeled using classical decision variables (booleans, integers, and floats) as well as variables representing collections of integers. A `list` variable is described by the maximum number of elements n it can contain and takes any ordered subset of $\{0, \ldots, n-1\}$ for value. This type of variable is especially useful to model routing problems and the solver can take advantage of this structure to improve its performance. In order to measure the quality of solutions produced for routing problems, methods dedicated to `list` variables were implemented in LocalSolver.

2 Lower bound computations

2.1 Held & Karp Relaxation

Let $G = (V, E)$ be a complete undirected graph and c_{ij} a cost function defined for each edge (i, j) of G. The *Traveling Salesman Problem* (TSP) consists in finding the minimum cost hamiltonian cycle in G. To compute a lower bound for this problem, we can use the lagrangean relaxation proposed by Held & Karp [2] which consists in iteratively computing a series of 1-trees.

Given i, a vertex of G, we call 1-tree a spanning tree of the subgraph induced by $V \setminus \{i\}$, together with two edges incident with i. Note that a tour is actually a 1-tree in which every vertex is of degree 2 and a minimum 1-tree is then a lower bound of the optimal tour in G.

Given π a weight vector which associates a weight π_i to every vertex $i \in V$, we can redefine the cost function for every edge as $\bar{c}_{ij} = c_{ij} + \pi_i + \pi_j$. The order of TSP solutions remains unchanged but the solution to the minimum 1-tree problem will be affected by perturbations of π. Hence, we can use a sub-gradient algorithm, in which we modify the weight vector iteratively to try and produce 1-trees that resemble tours.

In LocalSolver, modifications of the original algorithm were added to reduce execution time, handle the asymetric case and compute a bound for the minimum hamiltonian path problem as well. Starting from version 8.5, when the solver detects a TSP structure using a `list` variable, a lower bound of the problem will be automatically provided.

2.2 Using primal solutions produced by the solver

In order to improve the lower bounds computed, the use of primal solutions produced by LocalSolver was studied. The idea was to leverage the structure of good solutions to compute lower bounds during the resolution process. Focusing on the subtour elimination constraints, we can observe that binding constraints of this type correspond to strings of vertices adjacent in the optimal tour. A number of methods taking advantage of this property were developed (such as greedy algorithms and optimization of linear programs).

3 Results

Thanks to this work, LocalSolver is able to compute lower bounds during its *preprocessing*, and measure the quality of its solutions as soon as the resolution process starts. The table 1 shows the bounds computed for some instances of the TSPLib. The instances shown in the lower part of the table represent asymetrical instances.

Instance	**Opt**	**Bound**	**Gap (%)**	**Time (secs)**
gr48	5 046	4 950	1.9	0.020
st70	675	670	0.7	0.027
ch130	6 110	6 065	0.7	0.054
si175	21 407	21 321	0.4	0.130
a280	2 579	2 562	0.6	0.155
rat575	6 773	6 713	0.8	0.353
d1291	50 801	49 136	3.2	0.989
br17	39	39	0.0	0.005
ftv70	1 950	1 906	2.2	0.037
kro124p	36 230	35 924	0.8	0.053
ftv170	2 755	2 703	1.8	0.072

TABLE 1 – Lower bounds obtained during LocalSolver's preprocessing

Références

[1] F. Gardi, T. Benoist, J. Darlay, B. Estellon, et R. Megel. *Mathematical Programming Solver Based on Local Search.* Wiley, 2014.

[2] M. Held et R. M. Karp. *The Traveling-Salesman Problem and Minimum Spanning Trees.* Operations Research 18.6 (1970) p. 1138-1162.