Creating gradient to help local search algorithm
Application to tabu search for the simple Job-Shop-Scheduling Problem

Philippe Preux, David Duvivier
Laboratoire d'Informatique du Littoral, BP 719, 62228 Calais Cedex, France
Philippe.Preux@lil.univ-littoral.fr

February 2, 1999

Basically, local search algorithms examine the neighborhing points of the current point, evaluate their quality, and select one on which they iterate as long as a certain criterion is fulfilled. In other words, they follow some kind of gradient to reach extremal points, the highest point when maximizing, the lowest when minimizing. This line of work is all very nice as long as a gradient does exist but it breaks down when a "flat" area is reached (in the sequel, we call "plateau" such a flat area in the research space). In this case, one can let the algorithm take a neighbor at random until a gradient is found again, hoping it goes towards a good direction. Using a taboo list can help eschewing cycling. An other way is to use extra information that is not used by the algorithm when quietly cruising along a gradient. Within a gradient, using this information might be found not to improve the search and somehow costly to obtain, thus its cost being not its worth. However, when stucked within a flat area, paying the price to compute this information and using it to guess a sequence of moves to escape the plateau may become tremendously efficient. Obviously, it is crucial that the extra information, though different from the objective to optimize, should direct the algorithm towards a good direction, that is leads to points that, when the plateau is left, are better than the point where the plateau was entered. It is even more obvious that this extra information should not be constant for the points belonging to the plateau; so it should discriminate between seemingly equivalent points, that is, points that are identical with regards to the objective to optimize.

On the simple job-shop scheduling problem, we use tabu search based on an operator that swaps two operations of a planning and reschedule it afterward to obtain an active planning. We optimize the makespan. Using this operator, plateaus are found which average size scales with the instance size. Thus, we have embedded a mechanism that uses extra information to create a gradient, in a good direction, and helps the tabu escapes plateaus efficiently.

We have investigated several sources of information about a planning. We have observed that the discrimination power of these measures is rather high, that is, they do not remain constant on makespan plateaus. We have also measured the correlation between the makespan of points and their quality according to these measures and among these measures themselves. This tells us wether using a certain measure makes sense with regards to finding optimal points. Then, we have studied various combinations of these criteria to obtain the best combination and tuned their relative weight.

Finally, we have shown that using this technique, tabu search basically reaches better points within less iterations. We have also obtained the same kinds of result when this technique has been embedded within a genetic algorithm (actually, a hybrid, or memetic, algorithm that carefully combines the GA/GT recombination operator [NY92] and hill-climbing, see [DPT96]).

Furthermore, this technique can be brought onto a wide spectrum of combinatorial optimization problems where it is common to have some intuition about features that makes a solution "good", even though we do not always use it within objective function.

References

[DPT96] D. Duvivier, Ph. Preux, and E-G. Talbi. Climbing-Up $\mathcal{NP}$-Hard Hills. In Hans-Michael Voigt, Werner Ebeling, Ingo Rechenberg, and Hans-Paul Schwefel, editors, *Parallel Problem Solving from Nature (PPSN'96)*, pages 574–583. Springer-Verlag, Lecture Notes in Computer Science vol. 1141, September 1996.

[MM92] R. Männer and B. Manderick, editors. *Proc. of the Second Conf. on Parallel Problem Solving in Nature*. Elsevier Science Publishers, Amsterdam, 1992.

[NY92] R. Nakano and T. Yamada. A genetic algorithm applicable to large-scale job-shop problem. In *[MM92]*, pages 281–290, 1992.