An efficient POPMUSIC based approach to the Point Feature Label Placement Problem

Adriana C.F. Alvim[†], Éric D. Taillard[‡]

[†]Federal University of the State of Rio de Janeiro (UNIRIO), Brazil
[‡]School of Business and Engineering Vaud, SiM-TIC Institute, University of Applied Sciences of Western Switzerland, Switzerland
[†]adriana at uniriotec.br, [‡]eric.taillard at heig-vd.ch

This work, outlined in Alvim and Taillard [1], address the point feature label placement problem (PFLP) which is the problem of placing text labels adjacent to point features on a map so as to maximize legibility. We consider a set of n points, each one with p candidate label positions. A solution S is a list of n labels. For any S, we denote by $f(S)$ the function that counts the number of point features labeled with one or more overlaps (in other words, the number of labels with conflicts) and by $c(S)$ the function that counts the number of overlaps. The goal is to minimize $c(S)$. Cartographic preferences also can be taken into account. For $p \geq 4$, the PFLP is NP-hard. With increasing use of electronic maps, fast and good labeling algorithms must be designed. The `POPMUSIC` approach proposed in [1] is analyzed under a practical complexity point of view. Computational time measures confirm that our `POPMUSIC` approach typically runs in $O(p \cdot n\ log(n))$ while producing solution of quality higher than any other heuristic approaches previously proposed.

POPMUSIC for the PFLP: Let us suppose that a solution S can be represented as a set of *parts* $s_1, \ldots, s_q$. Let us also suppose that a distance measure can be defined between two parts. The central idea of `POPMUSIC` introduced by Taillard and Voss [4], is to select a part s_i, called *seed part*, and a number $r < q$ of the closest parts from the seed part s_i to form a sub-problem called R_i. If parts and sub-problems are defined in an appropriate way, to every improvement of the sub-problem corresponds an improvement of the solution of the whole problem. The particular choices for our `POPMUSIC` implementation for the `PFLP` are: An initial solution is obtained with a fast constructive procedure (first two steps of procedure FALP [3]). Each of the n point defines a part $s_1, \ldots, s_n$. The choice of the next *seed* part is arbitrary. Two parts are at distance 1 if their correponding labels can overlap. Otherwise the distance corresponds to the shortest path using intermediate parts Finally, we use a Tabu Search Procedure [2], as sub-problem optimizer. The procedure ends either when a solution without overlaps is found or when there is no seed part available to create a sub-problem.

Evaluating the practical complexity of our approach: We applied `POPMUSIC` for the `PFLP` with $r = 10$ (`Pop(10)`) and $10r$ tabu iterations on increasingly large problem instances with the same label surface ratio and instances with increasingly large number of positions for each label. The overall complexity of `Pop(10)` cannot be deduced in the worst case since, a priori, the number of times a seed part is selected is unknown. However, the best case complexity is: $\Omega(nlog(n) + nrlog(r))$. In order to evaluate the practical time complexity of our approach, a new class of instances was generated as follows: Labels are of equal size (12×4). The points are distributed uniformly on a square of size $D\sqrt{n} \times D\sqrt{n}$, where $D = 7$

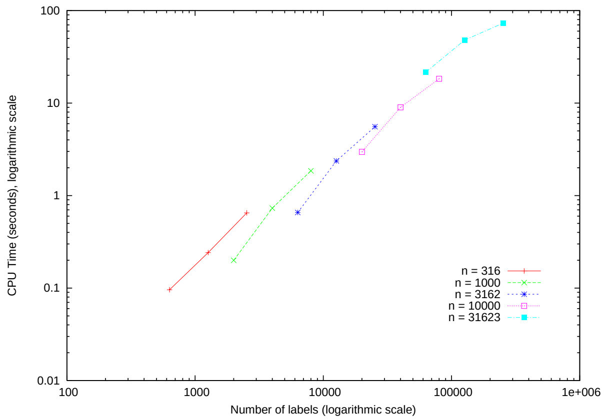

Figure 1: CPU time as a function of the number of labels (np). Average results for 20 instances running `Pop(10)`.

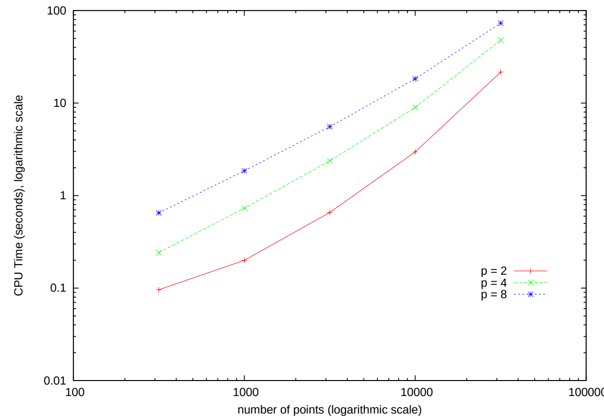

Figure 2: CPU time as a function of the number of points (n). Average results for 20 instances running `Pop(10)`.

is a constant. For each $n \in \{316, 1000, 3162, 10000, 31623\}$ we generated 20 instances and for each one we run variant `Pop(10)` with $p \in \{2, 4, 8\}$, totalizing 300 runs. Figures 1 and 2 shows these results. Figure 1 shows that the computational time does not increase too much with p (apparently less than linearly) and Figure 2 shows that the time increases almost linearly for $p = 4$ and $p = 8$ and $n \leq 10000$. Considering the increase of the computational effort for $n = 10000$ and $n = 31623$, we see that the time is multiplied by about 7 (respectively 5 and 4) for $p = 2$ (respectively $p = 4$ and $p = 8$). The clear over-linear increase for $p = 2$ is explained by the fact that the labeling problem is much harder (only 43% of the points can be labeled without conflicts) than for larger values of p (77% and 90% of point labeled without conflict), implying an increase of `POPMUSIC` iterations.

References

[1] A.C.F Alvim and É.D Taillard. POPMUSIC for the Point Feature Label Placement. *Extended Abstracts of the VI Metaheuristics International Conference*, Vienna, 39–44, 2005.

[2] M. Yamamoto, G. Camara, and L.A.N. Lorena. Tabu Search Heuristic for Point-Feature Cartographic Label Placement. *GeoInformatica* 6, 77–90, 2002.

[3] M. Yamamoto, G. Camara, and L.A.N. Lorena. Fast Point-Feature Label Placement Algorithm for Real Time Screen Maps. Presented at *VII Brazilian Symposium on GeoInformatics* (GEOINFO 2005), Campos do Jordão, Brazil, November, 2005.

[4] É.D Taillard and S. Voss. POPMUSIC: Partial Optimization Metaheuristic Under Special Intensification Conditions. Ribeiro, C. and Hansen, P. (eds.): *Essays and surveys in metaheuristics*. Kluwer Academic Publishers, Boston, USA, 613–629, 2001.

[5] F. Wagner, A. Wolff, V. Kapoor, and T. Strijk. Three rules suffice for good label placement. *Algorithmica* 30, 334–349, 2001.