An optimal algorithm for ressource allocation problem in concave context

A. Le Poupon and O. Rioul
ENST, 46, rue Barrault, 75634 Paris cedex 13
(Axel.LePoupon, Olivier.Rioul)@enst.fr

Keywords : Trees, Discrete Lagrangian Method, Dynamic programming

In this communication, an optimal algorithm is presented for solving the following resource allocation problem

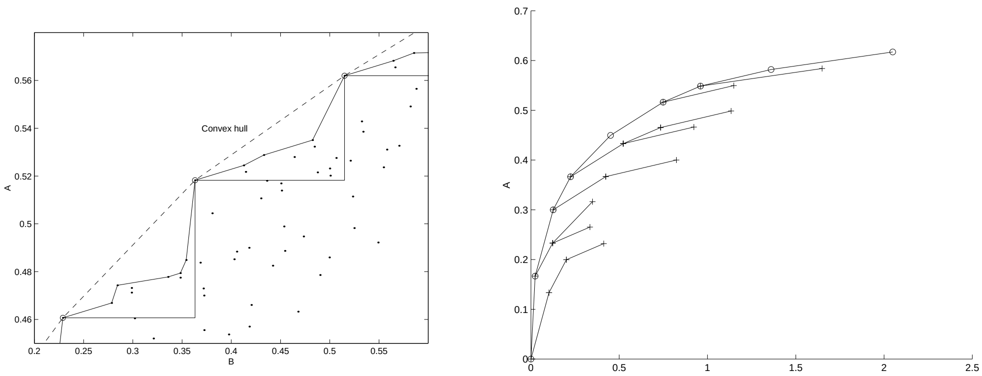
$$\max \left\{ A = \sum_{i=1}^{N} A_i \;\middle|\; B = \sum_{i=1}^{N} B_i \leqslant B_0 \right\} \tag{1}$$

where the (A_i, B_i) are constrained to a finite set of values and where B_0 is the ressource budget. We restrict our study to sets $\{(A_i, B_i)\}$ such that A_i is a concave function of B_i (see [3] for a generalization). Applications include optimal bit allocation procedures for source coding [1] and optimal power allocation for multicarrier channel coding [2].

Up to now, (1) has not been solved completely by an efficient algorithm. Dynamic programming has disastrous computational complexity, while Lagrangian methods [1] give sub-optimal results because the optimal solution does not necessarily lie on the convex hull in the A-B plane (see fig.1a).

Define a critical multiplier λ as one of the values $\lambda = \Delta A/\Delta B$ for which two consecutive points on the convex hull are joigned by a segment of slope λ. The Shoham-Gersho procedure [1] finds a point on the convex hull as the path corresponding to the whole sequence of multipliers $\lambda_1 \geqslant \lambda_2 \cdots \geqslant \lambda_M$. We propose a procedure to find the optimal solution to (1) obtained from a concave path corresponding to a subsequence $\lambda_{i_1} \geqslant \lambda_{i_2} \cdots \geqslant \lambda_{i_m}$ (see fig.1b).The search of an inaccessible solution is made efficient by the means of a test criterion ensuring that all subsequent subpaths passing through a given point consist only of suboptimal points. They are therefore removed from the search which continues with surviving paths.

We obtain the following results : for $N = 16$ sets of 16 values (A_i, B_i), solving (1) with B_0 covering the whole range (1000 test budgets), we find 100% of the optimal points when [1] finds in mean bearly 5%, and we operate in 5.0 seconds when dynamic programming lasts 451.5 seconds (on Pentium4 @2.6GHz).



(a) (b)

Figure 1: (a) Zoom on a A-B set showing inaccessible solutions (inside triangles) between consecutive (circled) convex hull solutions. (b) Examples of concave paths drawn from origin $(A, B) = (0, 0)$.

[1] Y. Shoham, A. Gercho, *ASSP* 36, 1988, 1445-1453.
[2] B. S. Krongold *et al.*, *ITC* 48, 2000, 23-27.
[3] A. Le Poupon, O. Rioul, "On optimal ressource allocation", *Trans. IT*, submited, 2005