RAIRO. RECHERCHE OPÉRATIONNELLE

JEAN-LUC LUTTON
ERNESTO BONOMI

Simulated annealing algorithm for the minimum weighted perfect euclidean matching problem

RAIRO. Recherche opérationnelle, tome 20, n° 3 (1986), p. 177-197

<http://www.numdam.org/item?id=RO_1986__20_3_177_0>

R.A.I.R.O. Recherche opérationnelle/Operations Research
(vol. 20, n° 3, août 1986, p. 177 à 197)

SIMULATED ANNEALING ALGORITHM FOR THE MINIMUM WEIGHTED PERFECT EUCLIDEAN MATCHING PROBLEM (*)

par Jean-Luc LUTTON (1) et Ernesto BONOMI (2)

Abstract. — *In this article, a randomized heuristic derived from the Metropolis procedure is proposed to solve the Minimum Weighted Matching Problem.*

In the limit of large problems, the average behaviour of the minimum cost of the perfect matching in the two dimensional Euclidean space is investigated for different probability distributions of points.

Keywords. — Metropolis algorithm; statistical mechanics; combinatorial optimization; minimum weighted matching problem; simulated annealing.

Résumé. — *Dans cet article une méthode heuristique non-déterministe, dérivée de l'algorithme de Metropolis, est développée pour résoudre le problème de couplage de points, dans l'espace euclidien, de poids minimal.*

Dans la limite de grands problèmes, le comportement asymptotique du coût de la solution optimale est étudié pour différentes fonctions de distribution des points dans le plan.

Mots clés : Algorithme de Metropolis; mécanique statistique; optimisation combinatoire; couplage de points de poids minimal; recuit simulé.

INTRODUCTION

Given a set **G** of N points (N even) inside a bounded domain **A** of Euclidean space, an instance of the minimum weighted matching problem is specified by a $N \times N$ distance matrix $D = (d_{ij})$.

(*) Reçu en février 1986.

(1) Centre National d'Études des Télécommunications, Division PAA/ATR/SST 38, 40 avenue du Général-Leclerc, 92131 Issy-les-Moulineaux.

(2) Centre de Physique Théorique, École Polytechnique, 91128 Palaiseau Cedex.

Adresse actuelle : Institut Physique Théorique, École Polytechnique Fédérale de Lausanne, CH-1015 Lausanne.

R.A.I.R.O. Recherche opérationnelle/Operations Research, 0399-0559/86/03 177 21 $ 4.10,

A perfect matching of **G** is a set of $N/2$ edges such that each point of **G** is an end-point of one and only one of the $N/2$ edges. Each possible perfect matching is represented by an element π of the set $\mathbf{E}_N$ of all admissible solutions. $\mathbf{E}_N$ contains exactly $(N-1)(N-3)\ldots 3.1$ elements.

Let $b_\pi(i)$ and $e_\pi(i)$ denote the end-points of the i^{th} edge, then the total cost of $\pi \in \mathbf{E}_N$, given D, is:

$$l_{\pi,D} = \sum_i d_{b_\pi(i)\, e_\pi(i)} \qquad i = 1, \ldots, N/2 \tag{1}$$

The minimum Euclidean matching problem consists in finding one element π^* such that the sum of the length of the $N/2$ edges is minimal:

$$l_{\pi^*,D} \leqslant l_{\pi,D} \qquad \text{for all} \qquad \pi \in E_N, \quad \pi \neq \pi^*. \tag{2}$$

Despite the typically non-polynomial character of many combinatorial problems, the matching problem can be solved exactly in $O(N^3)$ time by the Edmonds algorithm as implemented by Gabow and Lawler [2], [12], [15].

Obviously, when N increases ($N > 1\,000$), even this method requires a prohibitive computational effort and, consequently, heuristic algorithms running in $O(N^2)$ and $O(N)$ have been developed [1], [9], [16], [17].

As a corresponding drawback, the solution obtained by methods as spiral rack or serpentine algorithm [9] running in $O(N)$ is usually 30% in 50% more costly than the optimal solution (Table 1).

Algorithm	Complexity	Average μ
serpentine [9]	O(N)	0.585
λ-rectangle [17]	O(N)	0.516
spiral-rack [9]	O(N)	0.484
annealing	O(N)*	0.32-0.34
strip [1]	O(NlogN)	0.474
modified strip [1]	O(NlogN)	$\leq$ 0.401
exact [9]	$O(N^3)$	0.32-0.33
* numerical estimate : for large N one arrives within a few percent of the optimal cost in linear time		

TABLE 1. — (Measure of **A**) = 1, $\mu = cost/\sqrt{N}$.

In this article we present a randomized heuristic which seems to give solutions within a few percent of the optimal solution in linear computational time. Of course, our method goes slower than spiral rack and serpentine algorithm but this does not prevent our method having a linear time behaviour.

We use the Metropolis procedure [13] to generate a sequence of admissible solutions $\pi \in \mathbf{E}_N$.

It is well known that a lot of deterministic heuristics based on a local search procedure only allow transitions to lower cost thus trapping, after a finite number of iterations, the current solution in a local minimum. The introduction of the Metropolis schedule permits transitions to higher cost, so that, it is possible to escape out of local minima.

This ergodic process is controlled by a parameter T called "temperature". As the temperature is lowered, this sequence approaches the solution of least cost.

The same method has already been applied by the authors to investigate the asymptotic behaviour of the N-city travelling salesman problem [4] and of the quadratic sum assignment problems [5].

A large part of this article concerns the study of the average cost behaviour of the optimal solution π^* in the limit $N \to \infty$. We numerically verify the result shown by Papadimitriou [14]: the average minimum length of the perfect Euclidean matching is, up to a multiplicative constant, $\mu \simeq 0.321$, precisely the result rigorously proven by Beardwood et al. for the N-city travelling salesman problem [3].

1. STATISTICAL MECHANICS AND THE METROPOLIS PROCEDURE

The practical way to approach hard problems is to design approximate-solution algorithms whose running-time is proportional to small powers of N. A simple and natural heuristic method is based on a local search inside the set $\mathbf{E}$ of all admissible solutions. Of course, the local search is related to the concept of neighbourhood. This is defined, for each element π of $\mathbf{E}$, as the set $\mathbf{v}(\pi)$ of elements of $\mathbf{E}$ close in some sense to π. For example, let us consider a transformation $\boldsymbol{\Theta} : \mathbf{E} \to \mathbf{E}$, then we may define $\mathbf{v}_{\boldsymbol{\Theta}}(\pi)$ as the set of all elements which are accessible from π given $\boldsymbol{\Theta}$.

To illustrate this point, let us consider in the matching problem the **2-OPT** transformation rule [15]: "remove 2 edges from π and replace them with 2 other edges" (fig. 1).

Given the set $\mathbf{E}$ of all possible realisations of an optimization problem, given the cost function l_π and a transformation $\boldsymbol{\Theta}$, we call local minimum an admissible solution α which is not globally optimal, such that $\Delta l = l_\pi - l_\alpha > 0$ for all $\pi \in \mathbf{v}_{\boldsymbol{\Theta}}(\alpha)$.

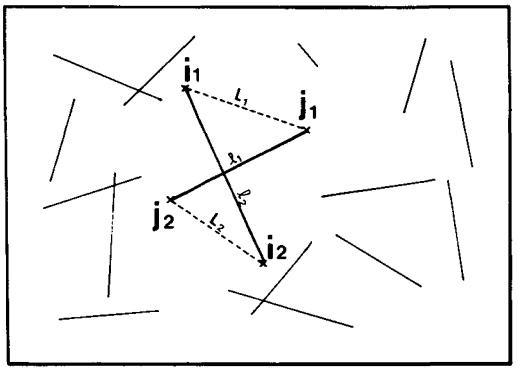

Figure 1. — Elementary transformation giving the trial solution from the current one. In the current solution, points i_1 and i_2 (resp. j_1 and j_2) are matched together. The trial solution is obtained by matching i_1 with j_1 (resp. i_2 with j_2). The cost difference between these two solutions is $\Delta l = L_1 + L_2 - l_1 - l_2$.

Obviously, the choice of Θ selects the set of all local minima of the problem.

The *local search procedure* is formally written as follows:

STEP 0: Select an initial solution $t \in \mathbf{E}$, $\pi := t$.

STEP 1: Select (in a deterministic way or at random) one element $s \in \mathbf{v}_\Theta(\pi)$.

STEP 2: Compute $\Delta l = l_\pi - l_s$, if $\Delta l \leqslant 0$ then $\pi := s$, goto 1.

In this approach, so long as an improved solution exists, we adopt it and repeat the procedure from the new solution until the algorithm is trapped in one local minimum α (or, by chance, in one globally optimal solution π^*). To release the algorithm from these trapping situations, the idea suggested by Kirkpatrick et al [10] [11] is to introduce fluctuations towards higher cost solutions.

By analogy with statistical mechanics, we modify the local search in the following manner:

STEP 1a: Select at random one element $s \in \mathbf{v}_\Theta(\pi)$, with a probability $q_{\pi,s}$ such that $q_{\pi,s} = q_{s,\pi}$.

STEP 2a: Compute Δl, set $\pi := s$ with a probability $p = \min(1, e^{-\Delta l/T})$, goto 1a.

This procedure, called the Metropolis algorithm [13], generates an irreducible Markov chain controlled by the "temperature" $T > 0$; hence eventually, the set $\mathbf{E}$ is covered with probability one. The iteration of this procedure leads to a unique stationary distribution proportionnal to the Boltzman factor [8]: $P_T(\pi) \sim \exp(-l_\pi/T)$.

The Metropolis algorithm allows us to sample the set $\mathbf{E}$ allocating to each admissible solution π a visiting probability $P_T(\pi) \leqslant P_T(\pi^*)$. From the expression

of the Boltzman factor, assuming the existence of n globally optimal solutions, we find:

$$\lim_{T \to 0} P_T(\pi) = \begin{cases} 1/n & \text{if } \pi \text{ is globally optimal} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

This implies that decreasing T (i.e. exponentially), the random walk generated by the Metropolis algorithm will converge to one globally optimal configuration π^* bringing out a sequence of solutions corresponding to lower and lower average values of the cost function $\langle l_\pi \rangle (T)$.

This sequence, and the time spent at each temperature T, is called an "annealing schedule". Fig. 2 shows some typical configurations of the minimum perfect matching problem resulting from an annealing schedule for given values of T.

Obviously, if we look for the configuration which maximizes the cost (maximum perfect matching), we must modify the annealing schedule by inverting the sign of T in *STEP* 2a.

Let us consider the average cost function of the minimum matching problem in the Boltzman formalism:

$$\langle l_{\pi,D} \rangle (T) = \sum_{\pi \in E} [l_{\pi,D} \exp(-l_{\pi,D}/T)]/Q(T)_D \tag{4}$$

where $Q(T)_D = \sum_{\pi \in E} \exp(-l_{\pi,D}/T)$, the normalization constant, is called the partition function.

It is clear that for $T \to \infty$, expression (4) becomes the arithmetic average of $l_{\pi,D}$ over the set of all solutions $\pi \in E_N$. In this limit, an elementary calculation gives:

$$\lim_{N \to \infty} \langle l_{\pi,D}/N \rangle (T \to \infty) = \langle d \rangle /2 \tag{5}$$

where $\langle d \rangle$ is the average distance between two points belonging to the domain **A**.

On the other hand, when $T \to 0$, according to (3), we obtain:

$$\langle l_{\pi,D} \rangle (T \to 0) = l_{\pi^*,D}. \tag{6}$$

Figure 3 illustrates the average behaviour of $l_{\pi,D}$ for an instance of the matching problem in which 400 points are uniformly distributed inside the unit-square **A**, $\langle d \rangle \simeq 0.521$ [4]. Crosses are numerical results obtained by a simulated annealing. The solid line represents an approximation of (4) developped in ref. [4]:

$$\langle l \rangle = [N \langle d \exp(-d/T) \rangle]/[2 \langle \exp(-d/T) \rangle] \tag{7}$$

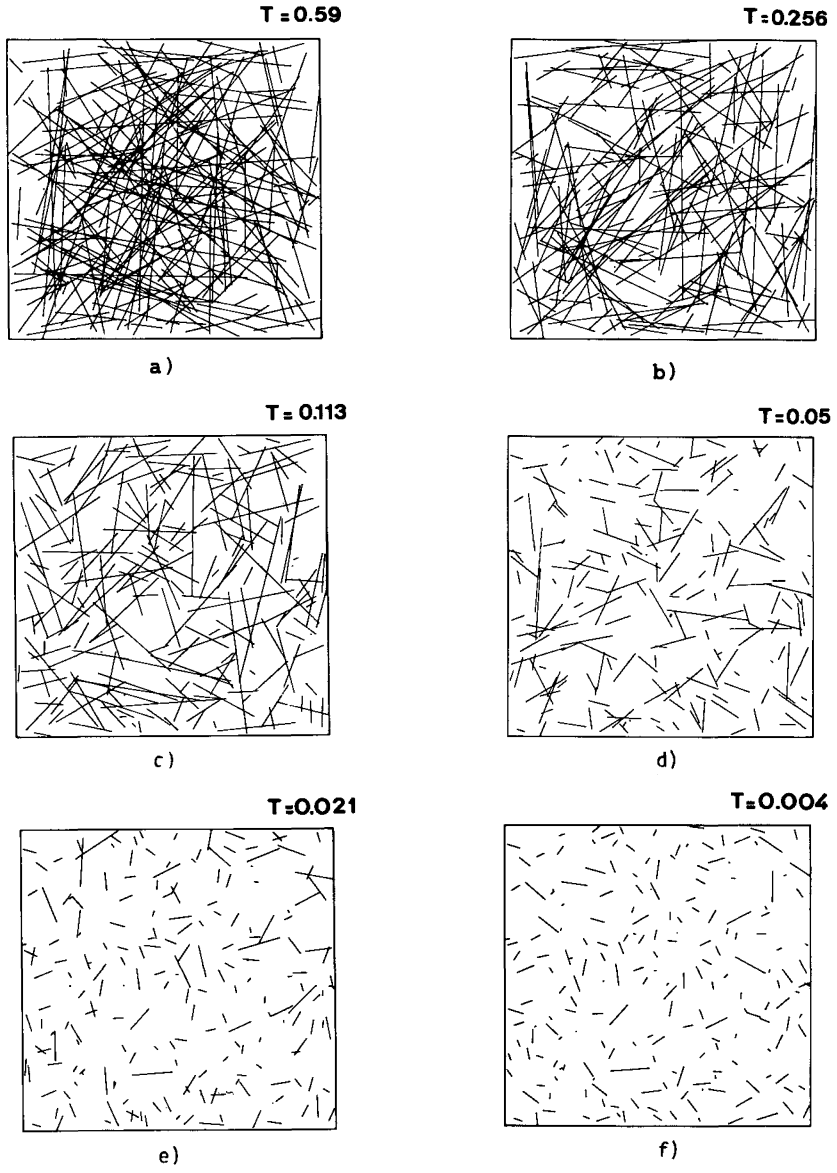

Figure 2. — Typical result from the aplication of an annealing schedule to an instance of the minimum weighted matching problem with $N = 400$. An initial configuration was chosen at random. Then the Metropolis algorithm with $T = 1.25$ was allowed to run for a predetermined number of steps. This process was repeated at lower temperatures using, as the new initial configuration, the final configuration at the previous temperature. The configurations appearing are typical configurations at six different temperatures:

a) $l = 83.67$; ***b)*** $l = 62.25$; ***c)*** $l = 37.48$; ***d)*** $l = 17.80$; ***e)*** $l = 9.16$; ***f)*** $l = 6.85$.

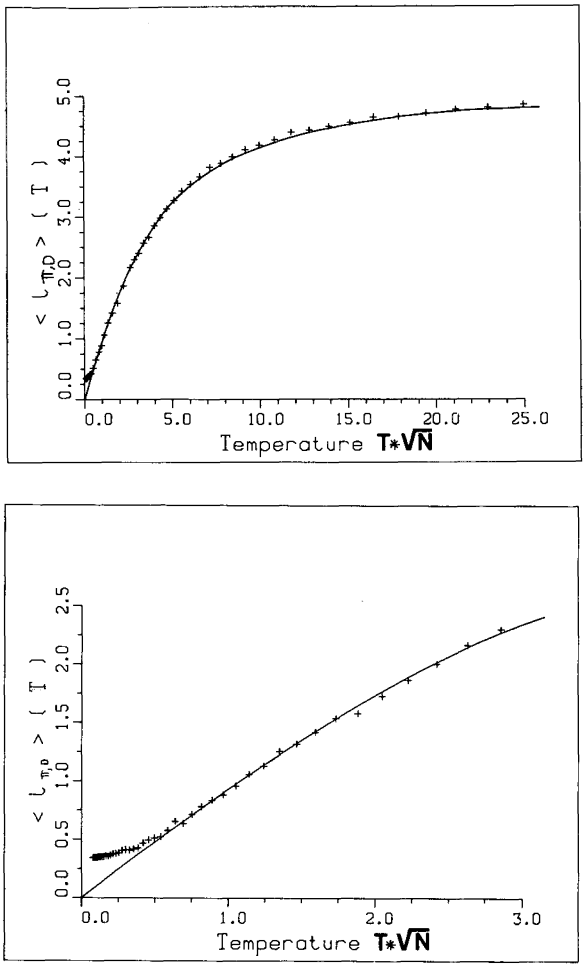

Figure 3. — Average matching cost $\langle l_\pi \rangle(T)/\sqrt{N}$, $N = 400$. The solid curve is given by equation (7). The crosses are the results of a single annealing schedule applied to one instance chosen at random (using a uniform distribution inside the unit square).

The right-hand side average in (7) covers the set of all possible values of the distance variable d inside **A**.

Figure 3 shows the agreement between expressions (7) and (4) for a range of temperature $T\sqrt{N} > 1$. At these temperatures, a lot of connections involve couples of remote points, giving rise to very long steps. On the contrary, at low temperatures, $T\sqrt{N} < 1$, the connections are confined to sub-regions of the domain **A**, giving rise to very short steps.

This last property has been exploited to improve the Metropolis searching procedure (section 2).

For the zero-temperature limit (6), a strong probabilistic result is stated in ref. [14]: the Euclidean perfect matching problem qualitatively behaves as the N-city travelling salesman problem [3]. In fact, it can be shown that, with probability one,

$$\lim_{N \to \infty} l_{\pi^*,D} N^{-1/2} = \mu \int_{\mathbf{A}} \rho(x)^{1/2} \, dx \tag{8}$$

$$1/4 \leqslant \mu \leqslant 0.401{,}06$$

where $\rho(x)$, $x \in \mathbf{A}$, is the probability density function representing the distribution of points inside the domain $\mathbf{A}$. Here, μ is a universal constant. Equation (8) becomes:

$$\lim_{N \to \infty} l_{\pi^*,D} N^{-1/2} = \mu \, |\mathbf{A}|^{1/2} \tag{9}$$

when $\rho(x) = |\mathbf{A}|^{-1}$; $|\mathbf{A}|$ is the area of the region.

Our estimation by means of the improved Metropolis procedure gives in average $\mu \simeq 0.321$ (section 3) which is, according to ref. [9] (exact algorithm), less than the value conjectured in ref. [14], $\mu \simeq 0.35$ (Table 1).

The value of the corresponding constant for the travelling salesman problem, β, is unkown as well; we estimate β to be 0.75 [4] as it is conjectured in [3].

The extension of the above probabilistic result (8) to other Euclidean problems is possible (minimum spanning tree problem, Steiner's problem, K-median problem), provided that the cost function satisfies four conditions stated in ref. [14].

These conditions assure, for sufficiently large problems, the optimal cost to be extensive. Roughly speaking, this means that, if the optimal configuration inside the union of two neighbouring domains $\mathbf{A}_1$ and $\mathbf{A}_2$ (fig. 4b) is compared with the optimal configurations obtained by considering both the domains separately (fig. 4a), then, when the number of points N belonging to the entire domain is large enough, the "global" cost and the sum of each partial cost computed independently differ by a factor which vanishes when $N \to \infty$ (fig. 4c).

A counter-example to this property is the maximum weigthed perfect Euclidean matching problem [18] whose cost indeed depends on the shape of the domain in which points are distributed. The result of an application of an annealing schedule to an instance of this problem is shown in figure 5.

Notice that for the maximum weighted matching problem, the crude version of the algorithm is surprisingly fast.

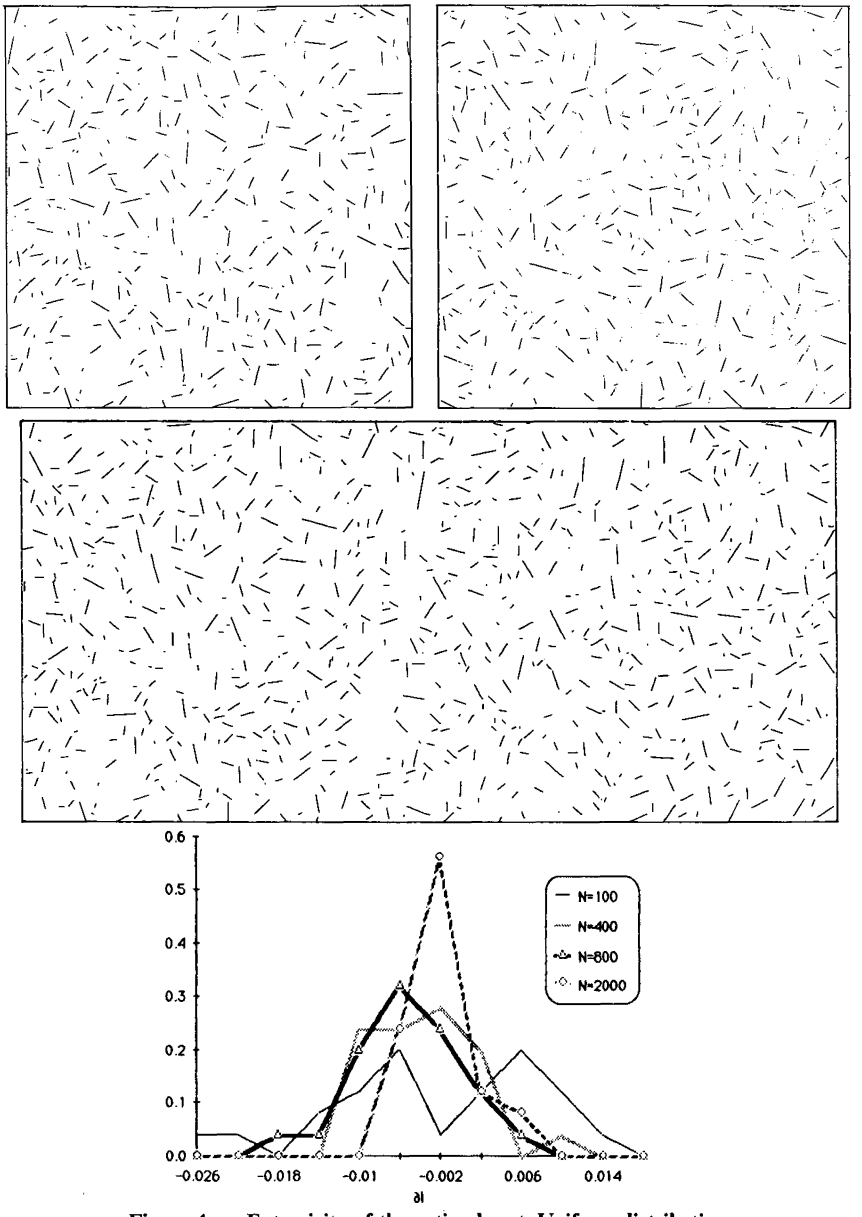

Figure 4. — Extensivity of the optimal cost. Uniform distribution.

a) **Both the domains are disjoint and each one contains $N = 400$ points: $\mu_1 = 0.3399$, $\mu_2 = 0.3417$ (simulated annealing).**

b) **The union of both the domain gives $\mu_{12} = 0.3405$.**

c) **Empirical probability distribution of $\partial l^{(N)}$:**

$$\partial l^{(N)} = (\mu_1^{(N)} + \mu_2^{(N)})/2 - \mu_{12}^{(2N)}.$$

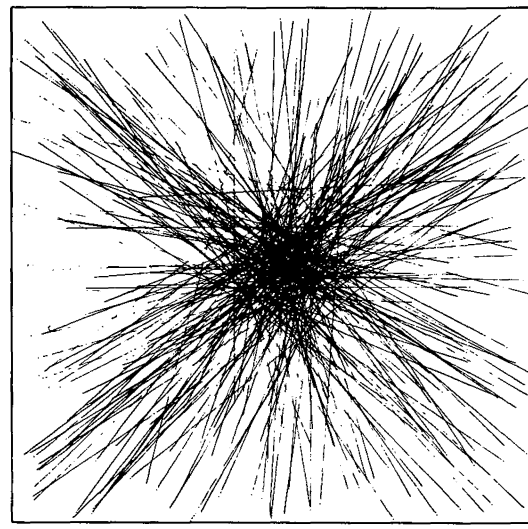

Figure 5. – Maximum weighted perfect Euclidean matching problem. Uniform distribution of points, $N = 400$. Configuration obtained by simulated annealing: $l/N = 0.3735$.

2. EXTENSION OF THE PROCEDURE

As has already been mentioned in section 1, the Metropolis procedure produces a sequence of current solutions $\pi \in \mathbf{E}_N$. The strategy is simple: by decreasing the temperature T (annealing schedule), one gets a set of solutions corresponding to lower and lower average values of the cost function (fig. 3).

However, computational experience shows that at low temperature the probability of accepting a trial configuration is small as opposed to the high temperature situation in which the ratio of accepted configurations is large.

According to *STEP* 2a of the algorithm, we have, as is shown in figure 1, that the change of the cost induced by the choice of the trial configuration is: $\Delta l = (L_1 + L_2) - (l_1 + l_2)$.

End-points of the edges with length L_1 and L_2 are identical random variables taking values inside the domain **A** (*STEP* 1a); their mean length, $\langle d \rangle$, is as the same order of magnitude as $|\mathbf{A}|^{1/2}$.

l_1 and l_2 are lengths of edges which characterized a matching at a given temperature T. Consequently, averaging Δl, we obtain:

$$\langle \Delta l \rangle (T) = 2[\langle d \rangle - 2 \langle l_{\pi,D} \rangle (T)/N] \tag{10}$$

In the limit $T \to \infty$, using (5), $\langle \Delta l \rangle (T)$ vanishes:

$$\langle \Delta l \rangle (T \to \infty) = 0. \tag{11}$$

On the other hand, when $T \to 0$, using (6) and (8), expression (10) gives:

$$\langle \Delta l \rangle (T \to 0) = 2 \langle d \rangle - o(\sqrt{N}). \tag{12}$$

Expression (12) implies that at low temperature any trial solution, obtained by transforming a current solution, will give on average a large value of $\Delta l/T$, so that, the probability of accepting a new configuration (*STEP* 2a) will be small and, globally, the searching procedure will slow down considerably. In figure 6 (uniform distribution of points inside the unit square, $\langle d \rangle \simeq 0.521$), the right hand side of (10) is compared with the measure of $\langle \Delta l \rangle (T)$ obtained by computer simulation. The agreement is good.

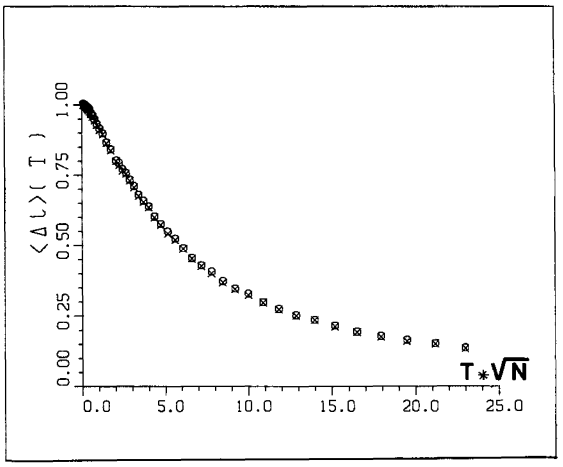

Figure 6. — Average $\langle \Delta l \rangle (T)$. $N = 1000$. The domain is not partitioned. The crosses are given by (10) where $\langle l_{\pi,D} \rangle (T)$ is the average matching cost experimentally measured at a given temperature. The circles are the measured average values of $\Delta l(T)$.

To improve the procedure, especially for large N, we have to keep in mind that at low temperature $T\sqrt{N} < 1$ (fig. 2) the connections between pairs of points are confined to sub-regions whose size is small compared to the entire domain in which points are distributed. For this reason, from the beginning of the procedure, we must force the algorithm to work on trial configurations which already exhibit this structural property.

The problem is to select trial configurations exchanging, according to *STEP* 1a, couples of points belonging to the same neighbourhood.

For Euclidean metric combinatorial problems, this is not a very difficult task. The solution is to divide the domain **A** into a set $\{\mathbf{R}_i\}_{i=1}^{n}$ of n disjoint sub-regions whose size is of the same order of magniture, $(|\mathbf{A}|/N)^{1/2}$, as the average step of the optimal configuration $\pi^* \in \mathbf{E}_N$.

The initial configuration is executed as follows. The sub-regions are considered in turn in the order given (for example) by the heavy line in figure 7. The points in the first sub-region are matched randomly. If there are an odd number of points in the first sub-region, the remaining point is matched with a point chosen at random from those in the next sub-region.

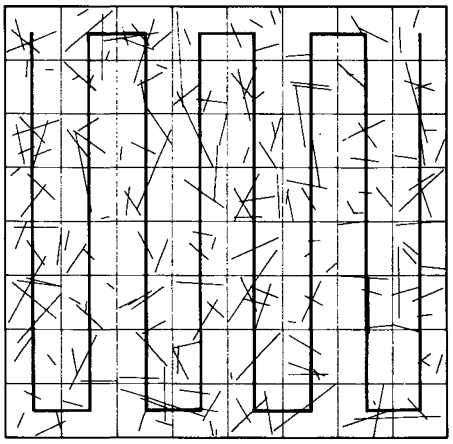

Figure 7. — $N = 400$. **Initial configuration with 64 sub-regions.**

The points in the next sub-region are matched at random and so on. This procedure is simply the serpentine algorithm [9].

The initial solution represents a low-temperature configuration and makes it possible to choose the initial temperature at $\Theta_0 = T\sqrt{N} < 1$.

The improved Metropolis algorithm is formally written as follows:

STEP 0: Divide the domain **A** into a set of disjoint sub-regions $[\mathbf{R}_i]$ and match points together to obtain an initial low-temperature solution.

STEP 1: Select a point i_1 inside a cell $\mathbf{R}_k$ (k is randomly chosen). Let i_2 be the point matched with i_1. Select at random a point j_1 (j_1 is matched with j_2) inside $\mathbf{R}_k$ or inside one of its nearest neighbour sub-regions. Execute the transformation described in figure 1 and evaluate the change in the cost function Δl.

If $\Delta l < 0$ go to *STEP* 3.

STEP 2: Select a random number $p \in [0,1]$.

If $p \geqslant \exp(-\Delta l \sqrt{N}/\Theta_n)$ reject the trial solution and go to *STEP* 4.

STEP 3: Accept the trial solution. Compute the value of the new cost function.

STEP 4: When statistical equilibrium is reached, then decrease the temperature Θ_n according to the annealing schedule. Go to *STEP* 1.

We remark that the effect to the partitioning is to narrow the search space and introduce a change of scale in Δl. This change of scale modifies substantially the probability of accepting a trial solution in *STEP* 2. This effect may be seen in figure 8.

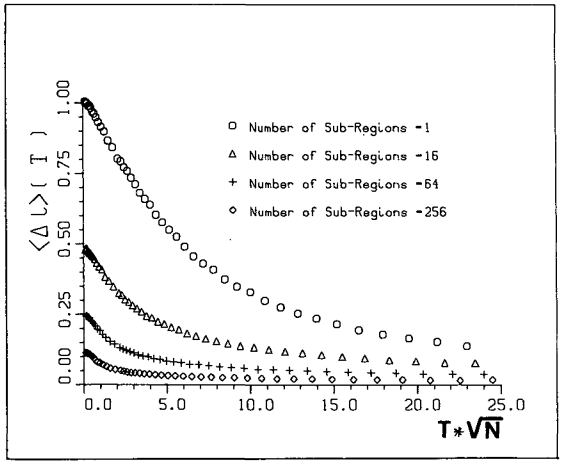

Figure 8. — Average $\langle \Delta l \rangle (T)$. $N = 1000$. Comparison of four partitioning schemes (number of sub-regions).

Another problem is the quantification of the "cooling" in *STEP* 4. An interesting result is given in ref. [6] and [7]: it is shown that the annealing schedule converges to the globally optimal solution if the temperatures Θ goes to zero no faster than $(C/\log m)$ as the number of iteration $m \to \infty$. C is a constant depending on the distribution of $(l_{\alpha,D} - l_{\beta,D})$, α and β are solutions belonging to the set of all local minima. Consequently, C is closely related to the transformation rule (section 1). However, the weak convergence of Θ and the difficulty to compute C make such a procedure impratical.

Our annealing schedule has been chosen in the following way:

$$\Theta_0 = 0.8 \tag{13}$$

$$\Theta_n = \gamma \Theta_{n-1} \qquad \gamma = 0.925 \qquad n = 1, 2, \ldots, 35.$$

the time spent at each temperature depends on the problem size.

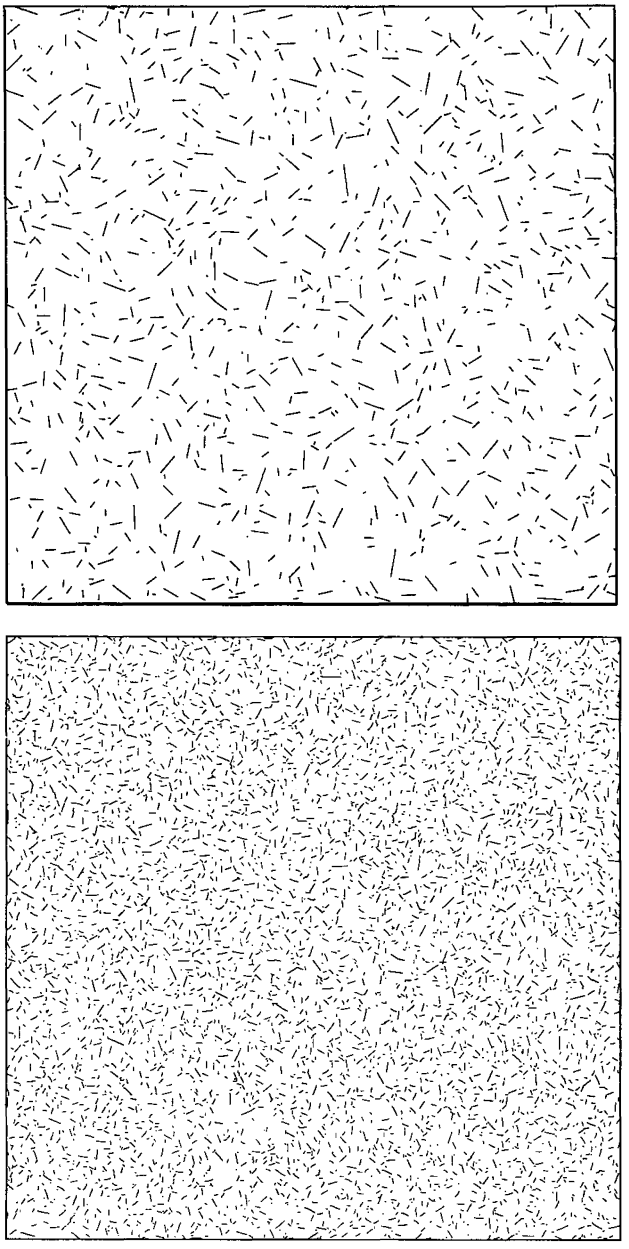

Figure 9. — Minimum weight matching configuration given by the improved simulated annealing.

a) $N = 2{,}000$, **Uniform distribution inside the unit square, 1024 sub-regions,** $l = 0.3326N^{1/2}$.

b) $N = 10{,}000$, **Uniform distribution, 4096 sub-regions,** $l = 0.3345N^{1/2}$.

The choice of (13) is merely empirical but it reflects an efficient compromise between a too abrupt annealing (the searching procedure may be trapped far from the global minimum) and a too slow one requiring a large computational time.

The application of the improved algorithm is illustrated in figure 9 where $N = 2{,}000$ and $N = 10{,}000$.

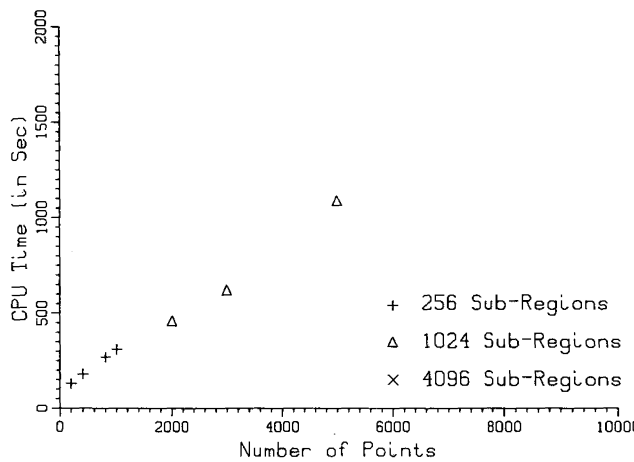

Figure 10. — Average CPU time required for the extended simulated annealing to obtain solutions a few percent of the optimal one.

In figure 10, we show the average CPU time (DPS8 Honeywell Bull computer) necessary to obtain a solution which is not more than 5% above the optimal one, as a function of the number of points N (problem size). For each problem size, the annealing schedule is described in (13). Only the number of iterations at each temperature step is different. It varies from 4,000 for $N = 100$ to 50,000 for $N = 10{,}000$ (10,000 for $N = 800$).

How do we choose these numbers? It is mainly the result of a lot of experiments. With these numbers of iterations per temperature step, we insure the obtention of a solution which is not 5% more costly than the optimal one (asymptotic limit). In the limit of great problems ($N \gg 100$), we observe that the mean number of transformation attempts per point must be at least 5 when the number of points per sub-region is about 2. If this number is greater, we have to increase the number of iterations (about 10 iterations per point for a number of points per sub-region close to 4).

We advise against taking more than 10 points per sub-region because the number of iterations then become prohibitive. Empirically we determine that the algorithm is most efficient when the number of points per sub-region is close to 4.

COMPUTATIONAL RESULTS

In this section, we use the improved procedure to exhibit a numerical estimation of the constant μ appearing in expression (8). We consider three different kinds of matching problem. Points are randomly drawn: each coordinate x and y is a random variable distributed according to a:

i) uniform law $x, y \in [0 ; 1]$: $\lim_{N \to \infty} l_{\pi^*,D} N^{-1/2} = \mu$

ii) Gaussian law $x, y \in \mathbb{R}$: $\lim_{N \to \infty} l_{\pi^*,D} N^{-1/2} = \mu(8\pi)^{1/2}$

iii) triangular law $x, y \in [0 ; 1]$: $\lim_{N \to \infty} l_{\pi^*,D} N^{-1/2} = \mu 16/9.$

Notice that in cases ii) and iii), another partitioning scheme is taken. We use polar coordinates and we divide the domain as shown in figure 11. The dimension of rings are chosen in such a way that the average number of points in each cell is the same. The initial configuration is determined by the serpentine cell order as described in figure 11.

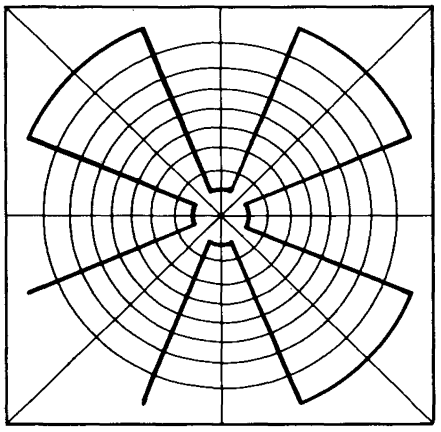

Figure 11. – Domain partitioning with polar coordinates.
The heavy line gives the cell order which generates the initial configuration.

In figure 12, we see two annealed configurations corresponding to situations ii) and iii).

For each case i), ii), iii) and for different values of N ($100 \leqslant N \leqslant 2{,}000$), we applied the simulated annealing algorithm to 50 different problems. The annealing schedule is given by (13).

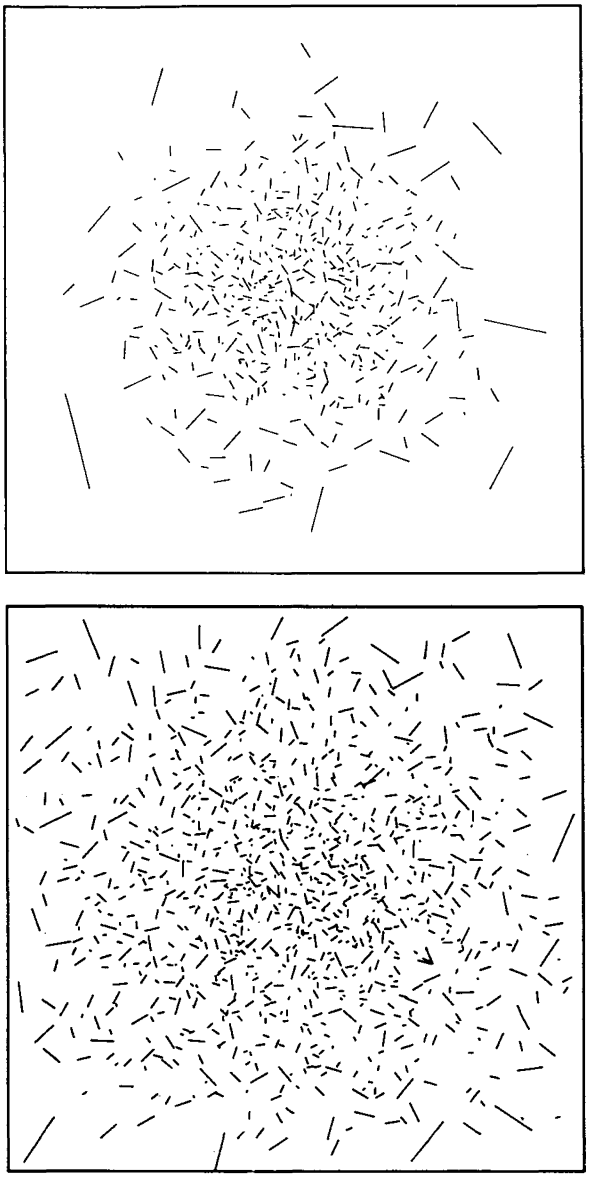

Figure 12. – **Minimum weight configurations (simulated annealing).**

a) $N = 1{,}000$, **Gaussian distribution inside the plane,**

$$l = 0.3248(2N^{1/2}).$$

b) $N = 2{,}000$, **Trianglular distribution inside the unit-square,**

$$l = 0.3341(16N^{1/2}/9).$$

For these three families of problems, figure 13 illustrates the empirical probability distribution of μ. We see that almost always the algorithm gives an "optimal" solution such that $0.32 < \mu < 0.35$. Notice that these results classify the simulated annealing procedure, applied to the Euclidean matching problem, as one of the best heuristics.

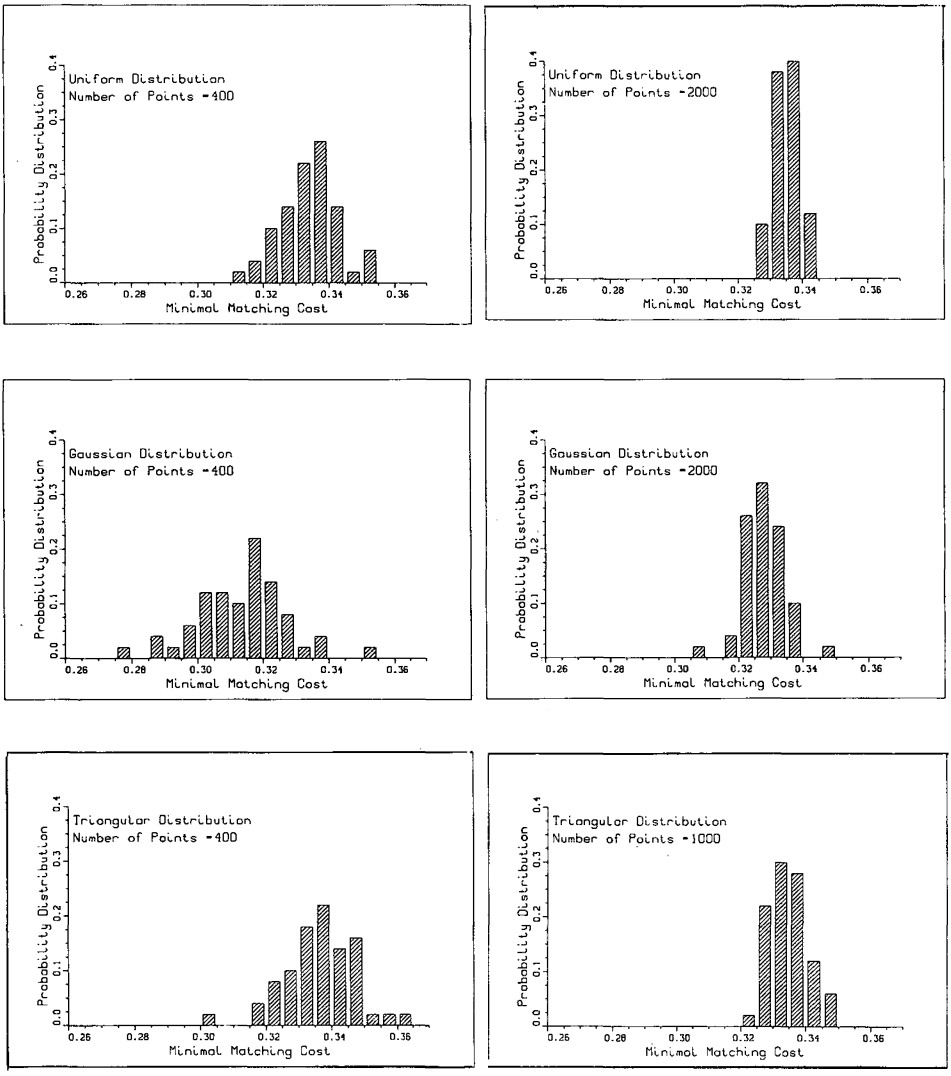

Figure 13. — Empirical probability distribution of μ for different values of N and different point distributions (simulated annealing).

Moreover, we observe that increasing the number of points N, corresponds to sharpening the empirical probability distribution of μ around a mean value μ^*. This is a consequence of the limit exhibited in expression (8). Assuming that, for large N, the procedure comes within a few percent (5%) of the optimal solution, we can estimate that the range of μ^* is [0.32; 0.33] which is in full agreement with the prediction given in [1] (exact algorithm).

In order to obtain a better accuracy in the determination of μ^*, we consider expression (10). For small values of the temperature $T\sqrt{N}$ (0.06, 0.04, 0.03), we numerically compute $\langle \Delta l \rangle (T)$ running the procedure without partitioning from a sub-optimal solution ($5\ 10^4$ steps).

From this result, we deduce $\langle l_{\pi,D} \rangle (T)$ and μ (uniform distribution of points), $N = 1000$:

$$2\mu = [\langle d \rangle - \langle \Delta l \rangle (T)/2] N^{1/2} \tag{14}$$

we average over 40 different instances and we estimate the value of μ^*.

For each temperature, we run 10 Metropolis computer simulations and we determine the 95% confidence interval. We obtain:

$$T\sqrt{N} = 0.03 \qquad \mu^* = 0.321 \pm 0.006$$
$$T\sqrt{N} = 0.04 \qquad \mu^* = 0.321 \pm 0.006$$
$$T\sqrt{N} = 0.06 \qquad \mu^* = 0.322 \pm 0.006$$

so, we deduce $\mu^* = 0.321 \pm 0.006$.

CONCLUSION

On the base of this work, as in [4], we feel the simulated annealing procedure to be a powerful numerical tool for solving combinatorial problems which call for the construction of some kind of shortest possible network of given a set of points (minimum spanning tree problem, Steiner's problem, K-median problem).

The temperature must be regarded as an external parameter which controls fluctuations of the random walk generated inside the set of all admissible solutions: it prevents the local search algorithm from being trapped by local minima. Decreasing the temperature corresponds to freezing the random searching process in a neighbourhood of the optimal solution.

Despite the optimism of some authors [19], we have noticed that choosing trial configurations inside the set of all possible solutions leads to excessive running time. Consequently, we have combined the "pure" Metropolis

algorithm with a "neighbouring" trial state selecting procedure, which favours the choice of "important" configurations in such a way that the change of cost Δl is reasonably small.

The efficiency of the algorithm is intimately related to the choice of the annealing schedule. The optimal choice is determined by the competition between the following two effects:

– if the temperature is lowered too abruptly then the searching procedure may end up in a local minimum quite far from the optimal state (more than 5% above),

– if, on the other hand, the temperature is lowered too slowly then the procedure will indeed converge to the ground state, but will do so extremely slowly.

However, we have seen how to reduce the importance of the choice of the annealing schedule by introducing a partitioning of the problem which narrows the search space. In this way, it has clearly appeared that a large improvement to the simple version of the Metropolis algorithm has been possible: for large N, the improved version gives a solution within a few percent of the optimal one in a linear time.

ACKNOWLEDGMENT

The authors wish to appreciate helpful discussions with H. Cameron, F. Horn and J. Roberts. We are particularly grateful to Prof. M. R. Feix for many valuable comments and suggestions.

REFERENCES

[1] D. Avis, *A survey of heuristics for the weighted matching problem*, Network, vol. 13, n° 4, 1983, 475-493.

[2] M. O. Ball, U. Derigs, *An analysis of alternative strategies for implementing matching algorithms*, Network, vol. 13, n° 4, 1983, 517-550.

[3] J. Beardwood, J. H. Halton, J. M. Hammersley, *The shortest path through many points*, Proc. of the Cambridge Phil. Society, vol. 55, 1959, 299-327.

[4] E. Bonomi, J. L. Lutton, *The N-city travelling salesman problem: Statistical Mechanics and the Metropolis Algorithm*, SIAM Review, vol. 26, n° 4, 1984.

[5] E. Bonomi, J. L. Lutton, *The Asymptotic behaviour of quadratic sum assignment problems: a statistical mechanics approach*, to be published in the Europ. J. Op. Res.

[6] S. Geman, D. Geman, *Stochastic relaxation, Gibbs distribution and the Bayesian restoration of images*, IEEE Trans, Pattern Anal. Machine Intell., vol. PAMI-6, 1984, 721-741.

[7] B. GIDAS, *Non-stationnary Markov Chains and Convergence of the annealing algorithm*, J. Stat. Phys., vol. 39, n° 1/2, 1985.

[8] J. M. HAMMERSLEY, D. C. HANDCOMB, *Monte-Carlo methods*, Chapman and Hall, London, 1964.

[9] M. IRI, K. MUROTA, S. MATSUI, *Heuristic for Planar Minimum-Weight Perfect Matching*, Network, vol. 13, 1983, 67-92.

[10] S. KIRKPATRICK, C. D. GELATT, M. P. VECCHI, *Optimization by simulated annealing*, Science, vol. 220, n° 4598, 1983, 671-680.

[11] S. KIRKPATRICK, *Optimization by simulated annealing, quantitative studies*, J. Stat. Phys., vol. 34, n° 516, 1984, 975-987.

[12] E. L. LAWLER, *Combinatorial optimization: networks and matroids*, Holt, Rinehart and Winston, New York, 1976.

[13] N. METROPOLIS, A. ROSENBLUTH, M. ROSENBLUTH, A. TELLER, E. TELLER, *Equation of state calculations by fast computing machines*, J. Chem. Phys., vol. 21, 1953, 1087-1092.

[14] C. H. PAPADIMITRIOU, *The probabilistic analysis of matching heuristics*, in Proc. 15th Ann. Allerton Conf. on Communication, Control and Computing, 1977, p. 368-378.

[15] C. H. PAPADIMITRIOU, K. STREIGHTZ, *Combinatorial Optimization Algorithms and Complexity*, Prentice Hall, Englewood Cliffs, N-Y, 1982.

[16] E. M. REINGOLD, R. E. TAYAN, *On a Greedy heuristic for complete matching*, SIAM J. Comput., vol. 10, n° 4, 1981, 676-681.

[17] E. M. REINGOLD, K. J. SUPOWIT, *Divide and Conquer heuristics for minimum weighted Euclidean matching*, SIAM J. Comput., vol. 12, n° 1 1983, 118-143.

[18] M. WEBER, Th. M. LIEBLING, *The Euclidean matching problem and the Metropolis algorithm*, private communication.

[19] *Problem solving*, The Economist, July 28, 1984, 70-71.