Recasting Deterministic Annealing as Constrained Optimization

Paul Stolorz

SFI WORKING PAPER: 1992-04-019

SANTA FE INSTITUTE

Recasting Deterministic Annealing as Constrained Optimisation

Paul Stolorz

Theoretical Division and Center for Nonlinear Studies

MS B213, Los Alamos National Laboratory

Los Alamos, New Mexico, 87545, USA

and

Santa Fe Institute

1660 Old Pecos Trail, Suite A

Santa Fe, New Mexico, 87501, USA

Email: paul@hopi.santafe.edu

March 31, 1992

Abstract

Several parallel analogue algorithms, based upon mean field theory approximations to an underlying statistical mechanics formulation, now exist for finding approximate solutions to difficult combinatorial optimisation problems such as the Travelling Salesman (TSP). These methods have also found application in such areas as speech and vision processing, as well as in adaptive learning and clustering algorithms. However, they all suffer from the substantial drawback of requiring an externally imposed "annealing" schedule similar to that used in simulated annealing. I show in this paper that any given "deterministic" (or "mean field theory") annealing algorithm can be combined in an extremely natural way with notions from the areas of constrained optimisation (Lagrange multipliers) and adaptive simulated annealing to yield a single homogeneous and parallel relaxation technique for optimisation. In particular, an externally prescribed annealing schedule is no longer required, which gives rise to somewhat more efficient procedures. The results of numerical simulations on 50-city TSP problems are presented, which show that the ensuing algorithms are typically an order of magnitude faster than the mean field algorithms alone. An analysis of the methods is presented which shows how their efficiency arises, and which also displays a mechanism allowing some unwanted local minima in the mean field theory methods to be avoided, thus leading, on occasion, to *qualitatively* superior solutions as well. This behaviour is illustrated by the ability of the new algorithms to locate a higher quality solution than deterministic annealing for a well-known 100-city instance of the TSP.

1 Introduction

The term "deterministic annealing" , or "mean field theory" (MFT) annealing, can be used to describe loosely several promising parallel analogue algorithms which have recently been proposed as heuristics for difficult combinatorial optimisation problems [1, 2, 3, 4, 5, 6]. The general approach has been applied to many target areas of great practical interest, ranging widely from the notorious Travelling Salesman Problem (TSP) [1, 2, 7, 8], to speech processing and computational vision [3, 4]. It has also been applied in the context of adaptive learning [9], and to the construction of automatic clustering algorithms for image processing problems [10]. The methods are labelled by a variety of terms reflecting the diverse fields in which they arose; for example, the Hopfield/Tank neural network [1], the elastic net method [2] and mean field annealing [6]. However, they all have a major common feature in that each attempts to locate the global minimum of a suitably crafted analogue "objective" function which has its roots in the quite successful simulated annealing heuristic [11, 12]. The latter is in contrast a stochastic procedure, requiring greater computational resources.

Several authors [13, 14, 15] have considered the alternative analogue approach of Lagrangian relaxation, a form of constrained optimisation due originally to Arrow [16], as a different means of tackling these problems. The various alternatives require the introduction of a new set of variables, the Lagrange multipliers. Unfortunately, these lead in turn to either the inclusion of computationally intensive "penalty" terms, or the consideration of restricted classes of problem constraints. The penalty terms also tend to introduce unwanted local minima in the objective function, and they must be included even when the algorithms are "exact" [17, 14]. These drawbacks severely limit their applicability when a problem is large in scale, containing say 100 or more variables.

In this paper I show that the technical features of analogue mean field approximations can be merged with both Lagrangian relaxation methods, and with the broad philosophy of adaptive simulated annealing without, importantly, requiring the large computational resources that typically accompany the Lagrangian methods. The result is a systematic procedure for starting with any given MFT algorithm, and crafting from it a single parallel homogeneous relaxation technique which needs no externally prescribed annealing schedule. In this way the computational power of the analogue heuristics is greatly enhanced. The basic idea is to simply recast as a relaxation variable the annealing, or temperature, parameter which already appears in the MFT formulations. In conjunction with this, the characteristic structure of the deterministic annealing objective (or energy) functions can be exploited to show that the most straightforward Lagrangian relaxation adaptations will suffice for these recast functions. It is not necessary to introduce expensive and/or complicated penalty terms to the procedure in the form of exact "augmented" Lagrangians. The sole stipulation is the introduction of an extremely simple and *inexpensive* penalty term. In fact, this term leads directly to a natural dynamics for the former annealing parameter, now a Lagrange multiplier variable, which roughly mimics the notion of

an adaptive annealing schedule in the simulated annealing heuristic. Some of the resulting candidate methods must be tested to ensure that they converge. However, in general the effective convergence of the methods is shown to be highly plausible. Furthermore, there are good reasons to expect the new algorithms to avoid some of the unwanted local minima that occur inevitably in the deterministic annealing procedures, and I present the results of numerical experiments which support this assertion.

The basic procedure is quite general in scope, and can be used to construct "candidate methods" from any of the deterministic annealing algorithms mentioned earlier. However, I have concentrated in this paper on its application to the well-known TSP benchmark. In particular, I show that the Lagrangian framework can be used to construct an efficient adaptation of the elastic net algorithm [2], which is perhaps the most promising of the analogue heuristics. This method actually has a structure which at first sight seems to preclude the straightforward use of Lagrangian relaxation, since the annealing parameter appears awkwardly in the constraint itself. However, this proves not to be a handicap for the elastic net, and it is demonstrated why this is so. This slightly surprising result is extremely encouraging, because it suggests that the simplest and most efficient Lagrangian relaxation adaptation can be attempted, with some chance of success, upon any deterministic annealing energy function, even one which does not have a true Lagrange multiplier structure. In addition, the apparatus can be generalised naturally to a procedure which uses several multipliers, in a manner that roughly parallels the notion of different temperatures at different physical locations in the simulated annealing heuristic.

The paper is laid out as follows. In Section 2, deterministic annealing will be reviewed, together with its relationship to simulated annealing. Lagrange multiplier methods are discussed in Section 3. The merged algorithms are then introduced in Section 4, and their convergence properties discussed in Section 5. In Section 6 the TSP problem is introduced, and Lagrangian relaxations of the Hopfield/Tank and elastic net MFT approximations are discussed. Section 7 presents the results of numerical experiments with the elastic net adaptations. The results display both increased computational efficiency, and qualitatively better solutions (avoidance of some local minima) over deterministic annealing. Section 8 presents a schematic view of the basic mechanism underlying the new algorithms, and contrasts them with both deterministic annealing, and with conventional constrained optimisation. After some concluding remarks, a fuller discussion of convergence properties, including some rigorous results, is given in Appendix 1. Appendix 2 contains a detailed discussion of the elastic net adaptation: its implementation, why it represents a sensible algorithm, and its convergence properties.

2 Deterministic ("mean field") annealing

To briefly review, each deterministic annealing procedure embeds the chosen discrete problem in a space of analogue variables. These variables obey a set of non-

linear ordinary differential equations, whose coefficients are determined by the data parameters of the problem, with the addition of an arbitrary "annealing", or "temperature", parameter. The common thread of the methods is that in each case, the relaxation of these equations to a fixed point can be identified with the location of a local minimum for some "energy function" $F(\underline{x})$. Although different for each method, this energy (or "objective") function is always of the form

$$F(\underline{x}) = U(\underline{x}) - TS(\underline{x}) \tag{1}$$

where $\underline{x}$ represents the analogue variables used to describe the particular problem at hand, and $T \geq 0$ is the adjustable annealing/temperature parameter. The form of the functions $U(\underline{x})$ and $S(\underline{x})$ is determined by the particular method of interest, and by the original discrete problem.

The object of the computation is to locate the global minimum of the function $F(\underline{x})$ when $T = 0$. In order to do this, the appropriate set of differential equations is initially relaxed to a fixed point for some value of the parameter T, which is chosen large enough so that the corresponding energy function (1) is convex. This guarantees that the local minimum obtained is in fact the global minimum at that value of T. The local minimum is then tracked as T is gradually lowered. The energy function becomes non-convex during this process (in general). Nevertheless, it is hoped that the local minimum being tracked remains a good approximation to the global minimum as T approaches zero, and can therefore be accepted as the "solution" to the original problem when $T \approx 0$. The algorithms are thus particular examples of continuation methods [18], and are also quite similar to the graduated non-convexity methods described in [19]. The general procedure is illustrated schematically in Figure (1).

The efficiency of these algorithms is due in large part to the fact that they can be considered to supply estimates, in relatively short computational time, of several important thermodynamic quantities of an underlying statistical mechanics system which encodes the original discrete problem. This underlying system is defined by the partition function

$$Z = \sum_{\underline{\mu}} e^{-E(\underline{\mu})/T} \tag{2}$$

In this expression, the vector $\underline{\mu}$ represents a set of discrete variables which describe legal solutions (henceforth referred to as configurations) to the original problem, and $E(\underline{\mu})$ denotes a "cost function" of these variables which must be minimised to find the solution: e.g. for the Travelling Salesman Problem, each $\underline{\mu}$ defines a possible tour through a given set of cities, and $E(\underline{\mu})$ represents the total length of each such tour. The required solution is the tour which minimises this length.

The successful heuristic based upon this discrete formalism is known as simulated annealing [11, 12]. It proceeds by beginning at high temperature T, stochastically generating configurations distributed according to the Boltzmann distribution $e^{-E(\underline{\mu})/T}$, and estimating the internal energy

$$U = \frac{1}{Z} \sum_{\underline{\mu}} E(\underline{\mu}) e^{-E(\underline{\mu})/T} \tag{3}$$

As the temperature is lowered, this average tends to become dominated by the configuration $\underline{\mu}^*$ which minimises the cost function $E(\underline{\mu})$ (or at least, by a configuration with a very low cost function). It thereby supplies an approximate solution to the original problem. It is customary to re-express the partition function Z as an equally fundamental function, namely the free energy F. This function is defined by $F = -T \ln Z$, which in turn can be written as

$$F = U - TS(U) \tag{4}$$

where U is the internal energy of the system as defined above, and where $S(U)$ is the entropy (logarithm of the number of configurations of energy U). The Boltzmann distribution, which is the basic object generated by simulated annealing at any given temperature, can be viewed as the distribution which minimises this free energy function at any given temperature.

Returning now to the analogue deterministic annealing heuristics, it has been shown that the values of the functions $F(\underline{x})$, $U(\underline{x})$, and $S(\underline{x})$ at minima of $F(\underline{x})$ constitute analogue "mean field theory approximations" for the free energy, internal energy and entropy respectively of the underlying discrete statistical physics system defined by (2) [4, 8, 6] (hence the choice of notation, especially the minus sign in (1)). In particular, the minimisation of the continuous function $F(\underline{x})$ can be thought of as an approximation to the minimisation of the free energy F of the discrete system. It therefore supplies an approximate solution to the original problem as the temperature is lowered to zero. In this way the deterministic analogue techniques make contact with simulated annealing. However, they have the advantage of requiring far less computational investment then the stochastic simulated annealing procedure. The reason for this is that, in very crude terms, a single configuration of continuous variables in one of the analogue heuristics represents an average over many different discrete states which must be looked at individually in simulated annealing. The deterministic heuristics have an obvious appeal in that they offer a systematic procedure for generating relatively simple yet effective analogue objective functions from a general and powerful statistical physics formalism.

However, a major drawback of the analogue heuristics is the requirement for an *ad hoc* external schedule by which the parameter T must be lowered. This schedule is usually quite slow, due to the combinatorial complexity of the original problem. It is a difficulty shared to some degree by simulated annealing, although in the latter case *adaptive* annealing schedules have been developed which substantially increase the efficiency of the algorithm [20, 21]. It would obviously be highly desirable to graft some of the attractive features of simulated annealing, such as the use of adaptive annealing schedules, onto the more computationally efficient analogue methods. In this regard, notice that one consequence of the statistical physics embedding of the analogue heuristics is that, in each case,

$$S(\underline{x}_{min}) \to 0 \text{ as } T \to 0 \tag{5}$$

where $\underline{x}_{min}$ is the local minimum of $F(\underline{x})$ obtained for the parameter value T. This deceptively simple observation allows the consideration of the somewhat different

approach of Lagrange multiplier methods to automatically determine the various values of T for the analogue heuristics, using as a constraint the vanishing of the entropy function at zero temperature. I note in passing that this particular fact has not been explicitly used in any previous work based on Lagrange multipliers, although it is implicit in the work of [13]. Most authors have focussed instead on the syntactic constraints contained in the function $U(\underline{x})$ when incorporating Lagrange multipliers. As a result the issue of eliminating an external annealing schedule has not been directly confronted.

3 Lagrange multipliers

Multiplier methods seek the critical points of a "Lagrangian" function

$$L(\underline{x}, \lambda) = f(\underline{x}) - \lambda g(\underline{x}) \tag{6}$$

where $f(\underline{x})$ is some function to be minimised, subject to the constraint $g(\underline{x}) = 0$, λ being a Lagrange multiplier. The minus sign is chosen for later notational convenience. By definition, the critical points of $L(\underline{x}, \lambda)$ obey the so-called Kuhn-Tucker conditions

$$\begin{aligned} \nabla_{\underline{x}} L(\underline{x}, \lambda) &= 0 = \nabla_{\underline{x}} f(\underline{x}) - \lambda \nabla_{\underline{x}} g(\underline{x}) \\ \nabla_{\lambda} L(\underline{x}, \lambda) &= 0 = -g(\underline{x}) \end{aligned} \tag{7}$$

Thus, at any critical point of this function, the constraint $g(\underline{x}) = 0$ is satisfied. Hopefully, in addition, $f(\underline{x})$ is minimised, subject to the constraint. There are several iterative methods for locating these critical points [17], in all which λ is given some kind of dynamical behaviour.

In accordance with the philosophy discussed above, we would like to identify the function $f(\underline{x})$ with the internal energy $U(\underline{x})$ from a deterministic annealing heuristic, the function $g(\underline{x})$ with the entropy $S(\underline{x})$, $L(\underline{x}, \lambda)$ with the free energy $F(\underline{x}, T)$ and the λ variable with the temperature T. The difficulty with this approach when used in isolation is that finding the critical points of $L(\underline{x}, \lambda)$ entails, in general, the minimisation of a transformed "unconstrained" function, whose set of local minima contains the critical points of L as a subset. This transformed function is required in order to ensure an algorithm which is convergent, because the critical points of L are saddle points, not local minima. One well-known way to do this is to add a term $g^2(\underline{x})$ to (6), giving an augmented Lagrangian with the same fixed points as (6), but hopefully with better convergence properties. Unfortunately, the transformed function is invariably more complicated than $L(\underline{x}, \lambda)$, typically containing extra quadratic penalty terms (as in the above case), which tend to convert harmless saddle points into unwanted local minima. It also leads to greater computational overhead, usually in the form of either second derivatives of the functions $f(\underline{x})$ and $g(\underline{x})$, or of matrix inversions [17, 14]. For large-scale combinatorial problems such as the TSP these disadvantages become prohibitive. In addition, the entropic constraint functions occurring in deterministic annealing tend to be quite complicated

nonlinear functions of the variables involved, often with peculiar behaviour near the constraint condition. In these cases (the Hopfield /Tank method is an example) a term quadratic in the entropy cannot simply be added to (6) in a straightforward way to produce a suitable augmented Lagrangian (of course, such a procedure *is* possible with several of the terms in the internal energy $U(\underline{x})$).

An alternative Lagrange multiplier scheme which avoids these penalty term problems has been developed in the context of the TSP [13]. This scheme is strongly in the spirit of the original Lagrangian relaxation method of Arrow [16] (described below), and is quite efficient. However, it requires a constraint function of no higher order than linear in $\underline{x}$, and indeed in the TSP case the constraint function is piecewise linear, resulting in a less than homogeneous technique. We would like to have less restrictive conditions on the constraints, in order to take full advantage of mean field theory techniques as a way of determining the best entropy constraint function $S(\underline{x})$. Another scheme containing very little penalty term overhead has also been constructed [15]. However, while this method comes close in philosophical spirit to the present work, it still does not directly address the question of avoiding an external annealing schedule. A method which does feature an automatic adaptive schedule in an analogue context has been developed as an improvement on the elastic net [22]. It is also very efficient, although it is not a Lagrange multiplier procedure, and is specific to the particular form of the elastic net method.

4 Combining Lagrange multipliers with deterministic annealing

To see how to proceed, we begin by making the Lagrange multiplier identifications discussed above. Since T will now be considered as a variable, denote it by λ. Now consider the efficient first-order algorithm (i.e. containing only first derivatives of $F(\underline{x}, \lambda)$) originally proposed by Arrow [16] as a means of locating the critical points of $F(\underline{x}, \lambda)$. It is specified by the equations

$$\begin{aligned} \dot{x}_i &= -\nabla_{x_i} F(\underline{x}, \lambda) &&= -\nabla_{x_i} U(\underline{x}) + \lambda \nabla_{x_i} S(\underline{x}) \\ \dot{\lambda} &= +\nabla_{\lambda} F(\underline{x}, \lambda) &&= -S(\underline{x}) \end{aligned} \tag{8}$$

Notice the change of sign between the two parts of (8). This is due to the saddle point nature of the critical points of $F(\underline{x}, \lambda)$, which means that regular gradient descent upon the free energy surface does not work. The Arrow algorithm is known to be convergent provided $F(\underline{x}, \lambda)$ is convex with respect to the variables $\underline{x}$ [17, 13], i.e. provided the matrix

$$\nabla^2_{xx} F(\underline{x}, \lambda) = \nabla^2_{xx} U(\underline{x}) - \lambda \nabla^2_{xx} S(\underline{x}) \tag{9}$$

is positive definite. This requirement is quite restrictive. It is clearly not satisfied in general by the non-convex deterministic annealing free energy functions (the convex function of [13] is an exception). Sophisticated procedures have been developed to

deal with situations such as this by grafting the above methods, based upon local convexity, onto relatively inefficient algorithms which can be proven to be globally convergent [17]. Unfortunately, the resulting hybrids are not single homogeneous relaxation techniques, and are therefore difficult to parallelise, as well as to make serially efficient.

However, we now make the central observation that for fixed $\lambda > 0$, not only are the deterministic annealing free energy functions $F(\underline{x}, \lambda)$ *locally* convex with respect to $\underline{x}$ about desired fixed points (after all, this is how the methods work to begin with), but that also, these locally convex regions tend to dominate the respective free energy landscapes. The convexity condition (9) thus holds over a large volume of the solution space. Hence, provided we insist upon the condition $\lambda > 0$, a simple Arrow-style dynamics which allows λ to vary can be expected to converge to a fixed point, since the small non-convex regions are unlikely to have a large effect upon the long-term dynamics. Therefore, given a deterministic annealing procedure as a starting point, it ought not be necessary to graft the locally convergent first-order Arrow algorithms onto less efficient procedures in order to ensure global convergence, despite the lack of strict convexity of the annealing free energy functions. Notice that the typical free energy landscape involving large convex regions is of some importance here. Pathological landscapes are possible in principle which contain relatively small convex regions, but whose various local minima control the same volume of solution space. If such cases were generic, it would no longer be possible to construct an efficient and convergent first-order Lagrangian relaxation adaptation by relying on the essential convexity of $F(\underline{x}, \lambda)$.

The requirement that λ be positive can easily be satisfied by adding to (8) a simple extra "barrier" term in the relevant equation of motion, so a typical proposed algorithm is given by the equations

$$\begin{aligned} \dot{x}_i &= -\nabla_{x_i}\hat{F}(\underline{x}, \lambda) &&= -\nabla_{x_i}U(\underline{x}) + \lambda\nabla_{x_i}S(\underline{x}) \\ \dot{\lambda} &= +\nabla_{\lambda}\hat{F}(\underline{x}, \lambda) &&= -S(\underline{x}) + c/\lambda \end{aligned} \tag{10}$$

where $\hat{F}(\underline{x}, \lambda)$ is a slightly modified free energy function given by

$$\hat{F}(\underline{x}, \lambda) = U(\underline{x}) - \lambda S(\underline{x}) + c\ln\lambda \tag{11}$$

In these expressions, $c > 0$ is a constant, chosen small on the scale of the other parameters. The particular form of the extra term that it characterises is not very important. The main criteria for choosing it should be that it interfere as little as possible with the dynamics of λ specified by the entropy term, while at the same time ensuring that λ remain positive. In this way the structure of the original annealing Lagrangian dominates the behaviour of the new homogeneous system. Indeed, in the numerical experiments that will be presented, a penalty term for λ was not even used - the algorithm was simply terminated at a suitably small value of λ.

The algorithms described by (10) above have several features which distinguish them from previous work. The first distinctive aspect is that the entropy estimate $S(\underline{x})$ has been chosen as the constraint function determining the rate of alteration

of λ. One consequence is that since this function is usually positive for the mean field theory heuristics, λ (the only new variable) decreases monotonically in a manner roughly similar to the temperature decrease schedule used in simulated and deterministic annealing, but with the *adhoc* drawback now removed (it also remains positive, as does the temperature in annealing methods). Moreover, there is no requirement that the system be at or near a fixed point each time λ is altered, which is a requirement for the annealing heuristics - there is simply a single homogeneous dynamical system which must approach a fixed point only once at the very end of the simulation. Finally, the algorithms do not require extra structure in the form of quadratic penalty terms, second derivatives or inverses, in contrast to the usual Lagrangian relaxation techniques [14, 17]. All of these features can be seen to be due to the statistical physics setting of the annealing "Lagrangian", and the use of an entropic constraint instead of the more usual syntactic constraints.

To elaborate, most Lagrangian relaxation methods allow both the multiplier and the constraint function to oscillate around some equilibrium value, provided there is eventual convergence consistent with the constraint condition. The equilibrium *value* of the multiplier is determined indirectly by the form of the objective function, including especially quadratic penalty terms. In contrast, the statistical physics structure of the annealing free energy functions specifies the final equilibrium value to assign to the multiplier (zero), and also demands that it must not oscillate about this value, but rather must approach it from above ($\lambda > 0$). This information is used to directly set a barrier penalty for λ. Hence an extra indirect and usually costly quadratic penalty term is not needed in the Lagrangian to perform this task, and an efficient first-order algorithm can be constructed. The efficiency derives both from the lack of these extra terms, and the fact that the objective function is essentially linear in λ except near the end of the procedure, when the simple barrier term makes its presence felt. This juxtaposition of the technical setting of constrained optimisation with the philosophical thrust of adaptive simulated annealing appears to be a crucial ingredient in the efficiency of the technique, and is a major point of departure from previous algorithms. It allows a rather complicated, but computationally powerful, entropy function to serve as a perfectly good constraint function for an effective Lagrangian relaxation scheme.

5 Convergence properties

How can the convergence of the above algorithm be made plausible mathematically? Consider a new "energy function", similar to the one used by [13], given by

$$E = \frac{1}{2}\sum_i \dot{x}_i{}^2 + \frac{1}{2}\dot{\lambda}^2 \tag{12}$$

Its time derivative, using equations (10), can easily be shown to be

$$dE/dt = -\sum_i \sum_j \dot{x}_i \, \nabla^2_{x_i x_j} \hat{F}(\underline{x}, \lambda) \, \dot{x}_j - c(\dot{\lambda}\,/\lambda)^2 \tag{13}$$

(See Appendix 1 for more details). Thus E would always be decreasing if $\hat{F}(\underline{x}, \lambda)$ were always convex with respect to $\underline{x}$: it would be a so-called Lyapunov function for the system of equations. Although this is not the case, the dominance of the locally convex regions of $\hat{F}(\underline{x}, \lambda)$ should mean that E is *usually* decreasing for the given algorithm, i.e. that E is "almost" a Lyapunov function for the method, provided the small regions of non-convexity of $\hat{F}(\underline{x}, \lambda)$ do not interfere too strongly. The behaviour is a reflection of the fact that in the original deterministic annealing methods, where $F(\underline{x})$ given by (1) is a true Lyapunov function for the dynamics, the alternative function

$$E = \frac{1}{2} \sum_i \dot{x}_i{}^2 \tag{14}$$

is "almost" a Lyapunov function also.

It can be expected then, that the function E will eventually reach its lower bound of zero, which of course describes a fixed point of the dynamics satisfying the Kuhn-Tucker constraints (7). This expectation is indeed born out by numerical investigations. Shown in Figure (2) is the time evolution of E for a typical TSP problem. The figure shows E, together with its "kinetic energy" components associated with λ and $\underline{x}$. Notice that apart from an initial region in which E increases (this is to be expected if the algorithm begins near a local maximum of $\hat{F}(\underline{x}, \lambda)$), the function decreases monotonically, so the constraints are gradually fulfilled. Equally important, however, is the fact that the individual kinetic components exchange energy during this process, allowing the $\underline{x}$ velocity to occasionally increase by slowing down the λ velocity. This behaviour can be viewed roughly as the presence of "inertial" influences on the variables, although this inertia is somewhat illusory - the true equations are first order, not second order as the inertial allusion implies (this issue is also discussed further in Appendix 1). In any case, this exchange of energy due to the extra degree of freedom supplies a mechanism by which the algorithm can occasionally overcome energy barriers between different regions of the solution space. An equivalent deterministic annealing procedure, lacking an inertial aspect, would in constrast become trapped in the first region it encountered. Of course, this behaviour by no means guarantees that the solutions found by the new algorithms will be global optima, but it does offer a way of obtaining qualitatively improved solutions over deterministic annealing, in addition to improved efficiency.

The Lyapunov function viewpoint can be pursued even further to display the efficiency of the algorithm(s) described by (10). Observe that at the fixed points, where $E = 0$, the K-T conditions (7) are precisely satisfied (setting $c = 0$). An alternative algorithm could therefore be constructed which concentrates on directly satisfying these conditions by regarding the function E simply as a function of λ and $\underline{x}$, and performing strict gradient descent with respect to λ and $\underline{x}$ upon this rewritten function

$$D(\underline{x}, \lambda) = \sum_i [\nabla_{x_i} U(\underline{x}) - \lambda \nabla_{x_i} S(\underline{x})]^2 + S^2(\underline{x}). \tag{15}$$

Thus, we would have the dynamics

$$\dot{x}_i = -\nabla_{x_i} D(\underline{x}, \lambda) \tag{16}$$

$$\dot{\lambda} = -\nabla_\lambda D(\underline{x}, \lambda)$$

for which $D(\underline{x}, \lambda)$ would be a true Lyapunov function. This algorithm is closely related to an exact Lagrangian procedure due to Di Pillo and Grippo [23]. Its disadvantages are that the minimisation of $D(\underline{x}, \lambda)$ requires second derivatives of $U(\underline{x})$ and $S(\underline{x})$, and that the sum-of-quadratic form also introduces extra local minima, a point which has been stressed by Bertsekas [17]. The first-order algorithm (10) can be thought of as taking a somewhat different trajectory than (16) in the space of variables, paying the price of not *monotonically* minimising the function $D(\underline{x}, \lambda)$, but in return receiving the twin rewards of much less computational investment per variable update, and higher quality solutions due to the presence of fewer local minima.

6 Application to the TSP

In order to illustrate the utility of the formalism outlined above, I now describe its application to a concrete case, the Travelling Salesman Problem (TSP), although it should be stressed that it applies to a much broader class of problems. The TSP consists of finding the shortest complete tour, beginning and ending at the same city, and visiting every city once, around a given set of cities with known fixed distances between each and every pair. Its ease of formulation hides a formidable computational task (in general), and it is considered a prototypical example of a combinatorially difficult problem. I will consider in some detail the elastic net heuristic for tackling the TSP, but before doing so, outline very briefly the application of the apparatus described by (10) to another analogue heuristic, the Hopfield/Tank algorithm [1].

The Hopfield/Tank method can be described as the minimisation with respect to $\underline{x}$, in conjunction with annealing in the parameter λ, of the function

$$F(\underline{x}, \lambda) = U(\underline{x}) - \lambda S(\underline{x}) \tag{17}$$

where

$$U(\underline{x}) = A/2 \sum_{a,i \neq j} x_{ai} x_{aj} + B/2 \sum_{a \neq b,i} x_{ai} x_{bi} \tag{18}$$
$$+C/2[\sum_{a,i} x_{ai} - n^2]^2 + D/2 \sum_{a \neq b,i} d_{ab} x_{ai}(x_{b,i+1} + x_{b,i-1})$$

and

$$S(\underline{x}) = -F/2 \sum_{a,i} [x_{ai} \ln x_{ai} + (1 - x_{ai}) \ln(1 - x_{ai})] \tag{19}$$

The indices a and b label the different cities in a TSP problem, while i and j label the order in which the n cities are visited. The analogue variables x_{ai}, which are kept in the range [0,1] by the form of $S(\underline{x})$, represent the probability of city a occurring at the i^{th} step of a tour. The distances between cities a and b are specified by the parameters d_{ab}, while A, B, C, D and F are arbitrary positive parameters which

must be fine-tuned to obtain an optimal algorithm (alternatively, some of them may be converted into Lagrange multipliers themselves [24]). If the algorithm is modelled by an analogue electronic circuit, the parameter $\lambda > 0$ can be viewed as the inverse of an adjustable "gain" parameter.

The procedure described by (10) can be implemented immediately to convert this algorithm into a Lagrangian relaxation procedure. The Lagrange multiplier in this case is the inverse of the gain parameter. Since the entropy function $S(\underline{x})$ is never negative, λ will decrease monotonically to zero, at which point the entropy will also be zero. This is exactly what is expected of the Hopfield/Tank algorithm: at high initial values of λ (low gain), the variables $\underline{x}$ occupy the interior of an n^2-dimensional hypercube, with a corresponding positive entropy, then as λ is lowered (i.e. the gain is increased) the variables move towards the corners of the hypercube, lowering the entropy until finally it reaches the value zero at the corners themselves (infinite gain, or zero temperature).

The focus of attention will now be switched to the elastic net method [2], which is known to be a somewhat superior algorithm to the Hopfield/Tank method. The reason for this is that it supplies a substantially more precise estimate of the thermodynamic quantities of the relevant underlying statistical physics system than does Hopfield/Tank [4, 8, 6]. This manifests itself as a formalism with fewer variables and fewer sub-optimal minima. The numerical experiments reported here therefore concentrated on using the elastic net as the initial deterministic annealing procedure. It can be described as the minimisation by simple gradient descent of the function

$$F(\underline{x}, \lambda) = U(\underline{x}) - \lambda S(\underline{x}, \lambda) \tag{20}$$

where

$$U(\underline{x}) = \gamma \sum_{i}^{M} |x_i - x_{i+1}|^2 \tag{21}$$

and

$$S(\underline{x}, \lambda) = \alpha \sum_{a}^{N} \ln \sum_{i}^{M} e^{-|x_i - x_a|^2/2\lambda^2} \tag{22}$$

In these expressions, x_a is a (fixed) 2-dimensional vector specifying the position of city a, x_i is a (variable) 2-dimensional vector describing the position of tour point i, and of course λ is the annealing/temperature parameter. The remaining tunable parameters are α and γ. The tour points represent a deformable curve whose shape is manipulated by the algorithm until it eventually passes through each city once. In general, the number of tour points M is chosen to be somewhat larger than the number of cities N. Several other mean field algorithms developed in the context of image-processing have a very similar free energy structure [5, 4, 3].

The elastic net works by minimising the "internal energy" term $U(\underline{x})$, which forces different tour points to be closer together. This tendency is balanced by the entropic term $S(\underline{x}, \lambda)$, which at high temperatures tries to pull tour points apart. As the temperature λ is lowered, $F(\underline{x}, \lambda)$ is best minimised by allowing $S(\underline{x}, \lambda)$ to approach zero, while minimising $U(\underline{x})$. Inspection of the form of $S(\underline{x}, \lambda)$ reveals that

at low values of λ, the entropy can only be zero if there is associated with each city x_a some tour point $x_i = x_a$, so at low temperatures the cities attract tour points. This situation guarantees that at low enough temperature the syntax of the original discrete problem will be satisfied, with the network specifying that there be at least one tour point located at each city.

There is an obvious impediment to the straightforward conversion of this algorithm into a Lagrangian relaxation procedure: the structure of $F(\underline{x}, \lambda)$ precludes the use of a true Lagrange multiplier, since λ now appears non-trivially in the constraint function $S(\underline{x}, \lambda)$ itself! Nevertheless, applying the Lagrangian relaxation apparatus regardless leads to the following simple first-order set of equations

$$\begin{aligned} \dot{x}_i &= -\nabla_{x_i} F(\underline{x}, \lambda) &&= -\nabla_{x_i} U(\underline{x}) + \lambda \nabla_{x_i} S(\underline{x}, \lambda) \\ \dot{\lambda} &= +\epsilon \nabla_\lambda F(\underline{x}, \lambda) &&= -\epsilon [S(\underline{x}, \lambda) + \lambda \nabla_\lambda S(\underline{x}, \lambda)]. \end{aligned} \tag{23}$$

where $U(\underline{x})$ and $S(\underline{x}, \lambda)$ are given by (21) and (22) respectively. The first line is simply the gradient descent term occurring in the original elastic net procedure, and the second describes the new "annealing dynamics". In this expression, the constant $\epsilon > 0$ appears in order to slow the convergence rate of λ. It functions as both an alternative to introducing an extra additive barrier term (this change of convergence rate does not of course alter the location of the fixed points), and as the sole remaining means of controlling the annealing schedule and avoiding sub-optimal solutions. There are two distinct possible sources of difficulty which could prevent this set of equations from being a useful optimisation algorithm. One is the issue of whether it can be expected to converge. The other arises from the fact that the simple constraint condition $S(\underline{x}, \lambda) = 0$, which is known to satisfy the syntax of the original problem, is not obviously satisfied at the fixed points of the new system. This is because λ is no longer a true multiplier. The question is, does the new constraint produce a sensible final configuration which satisfies the original constraint $S(\underline{x}, \lambda) = 0$ at $\lambda = 0$?

Convergence can be dealt with by considering the function

$$E = \frac{1}{2} \sum_i \dot{x}_i^2 + \frac{1}{2\epsilon} \dot{\lambda}^2 \tag{24}$$

In a similar manner as before, we find that

$$dE/dt = -\sum_i \sum_j \dot{x}_i \, \nabla^2_{x_i x_j} F(\underline{x}, \lambda) \, \dot{x}_j - [2\nabla_\lambda S(\underline{x}, \lambda) + \lambda \nabla^2_\lambda S(\underline{x}, \lambda)] \dot{\lambda}^2 \tag{25}$$

All the components of this expression have the correct sign for eventually lowering E, with the exception of the second derivative of $S(\underline{x}, \lambda)$ with respect to λ. It is negative, and becomes appreciable for small values of λ. In the numerical experiments, this problem was avoided by choosing ϵ small enough so that the first term dominated at all except small values of λ. The algorithm was then terminated before this regime was reached. If desired, a formal barrier penalty term could be used instead, in the manner of (10).

The remaining concern is to ensure that the new term $\lambda \nabla_\lambda S(\underline{x}, \lambda)$ appearing in the constraint condition does not interfere with $S(\underline{x}, \lambda)$ in such a way as to lead to a nonsense final configuration. It is possible, for example, for $S(\underline{x}, \lambda)$ to take on negative values (unlike the corresponding functions in the Hopfield/Tank procedure). In principle, these could cancel out $\lambda \nabla_\lambda S(\underline{x}, \lambda)$ at awkward places to produce a fixed point at an illegal solution. Fortunately, a little algebra settles this issue by showing, firstly, that the constraint function obeys

$$S(\underline{x}, \lambda) + \lambda \nabla_\lambda S(\underline{x}, \lambda) > 0 \quad \text{for } \lambda > 0, \tag{26}$$

and secondly, that it is zero only when both $\lambda = 0$ and when there is a tour point located at each city (for each a, $x_a = x_i$ for one value of i). More details concerning this issue and convergence can be found in Appendix 2. This last condition ensures that any fixed point of the algorithm satisfies the syntax of the original problem. The general entropic structure which gives rise to (26) above is not limited to the elastic net mean field theory approximation to the TSP. The mean field theory approximations to a variety of problems discussed in [3, 4, 5, 10] all satisfy (26), and can therefore be adapted successfully to Lagrangian relaxation methods.

The form of the elastic net entropy function suggests a further natural generalisation of the procedure. A different "multiplier" λ_a can be assigned to each city a, each variable being responsible for satisfying a different additive component of the entropy constraint. The required dynamics and the various new free energy functions are all straightforward extensions of those discussed above, and the same arguments concerning convergence and satisfaction of syntactic constraints apply. A detailed description of the implementation pf the algorithm, including the use of several multipliers, is given in Appendix 2. The same notion could also be applied, of course, to the adaptation of the Hopfield/Tank methods, by either assigning a multiplier to the entropy for each variable x_{ai} individually, or to groups of variables associated with a given city, etc. In general each of the resulting multipliers will converge towards zero at different rates, supplying more degrees of freedom to the problem in a natural way, and hopefully enhancing trap-avoidance. The idea has an obvious parallel to the notion in simulated annealing of lowering the temperature in different geographical regions at different rates in response to the behaviour of the system. The number of extra variables required is a modest computational investment, since there are typically many more tour points than city points for a given implementation.

7 Results for the elastic net Lagrangian relaxation

Numerical simulations were performed using three different algorithms: the elastic net method, its Lagrangian adaptation with a single global Lagrange multiplier, and the modification discussed above involving one Lagrange multiplier for each

	Elastic Net		Local Multipliers		Global Multiplier	
City set	Length	CPU (# its)	Length	CPU (# its)	Length	CPU (# its)
1	5.93	219s (400)	5.93	66s (100)	5.93	34s (60)
2	6.03	215s (390)	6.03	79s (120)	6.03	35s (60)
3	5.74	226s (410)	5.75	65s (100)	5.75	34s (60)
4	5.90	322s (580)	5.90	105s (160)	5.90	45s (80)
5	6.49	274s (510)	6.50	90s (140)	6.50	69s (120)

Table 1: Performance of the 3 heuristics described in the text on a set of 5 randomly distributed 50-city instances of the TSP generated by [2]. CPU times quoted are for a SUN SPARC Station 1+. Solutions were obtained by running the algorithm a single time from a standard starting loop.

city. The results of running the various algorithms on the five 50-city instances used by [2] are shown in Table 1.

The table displays the final tour lengths obtained, the total running time on a SUN Sparc Station 1+, and the number of fundamental "iterations" this required, for each of the methods and each of the 50-city instances. It can be seen that the tour lengths obtained by the Lagrangian methods are essentially the same as those found by the elastic net. However, the fundamental number of iterative steps required, and the total running time, is substantially lower for the Lagrangian procedures, especially for the global Lagrange multiplier. The latter method appears to be roughly an order of magnitude faster than the elastic net. Care was taken to compare the algorithms fairly by refraining from applying special numerical improvements to any of the methods to improve convergence properties. Several possible improvements, such as building up the number of tour points only as they are required, were indicated with by [2] in their original elastic net discussion. These improvements can typically be applied to all three methods quoted above - the main purpose of the table is to show the effect upon CPU time of the adaptive annealing notion itself, which can then be combined with other numerical improvements as desired.

The Lagrangian algorithms obtained solutions no more accurate than the elastic net on the problem instances shown above. However, the form of the algorithm appears to offer the possibility, discussed earlier, of generating qualitatively better solutions. The methods were therefore compared on a larger set of 40 randomly chosen 50-city problems, the results of which are displayed in Table 2. For this larger set of instances, the Lagrangian algorithms show a trend towards better solutions. On average, they are 0.5% shorter than the elastic net tours. Of course, this difference is well within a single standard deviation of tour lengths over the various instances, so the results are suggestive rather than conclusive. They do, however, show that the procedures hold promise for the TSP, because the problem is notorious for requiring substantially greater investments of computer resources to improve overall accuracies by small amounts as the optimal tour lengths are approached. Even small improvements over the elastic net results, occurring as they do in conjunction with

	Elastic Net	Local Multipliers	Global Multiplier
Tour Length	5.95 ± 0.10	5.92 ± 0.08	5.92 ± 0.09
CPU time (secs)	260 ± 33	82 ± 12	49 ± 5
α	0.2	0.4	0.4
γ	2.5	2.5	2.5
ϵ	0.02	0.012	0.010

Table 2: Performance of the 3 heuristics described in the text on a set of 40 randomly distributed 50-city instances of the TSP within the unit square. CPU times quoted are for a SUN SPARC Station 1+. The value of ϵ quoted for the elastic net represents the proportional decrease in the temperature applied after every 5 iterations.

significantly shorter CPU times, are somewhat encouraging.

The Lagrangian methods were also run on the 100-city problem used by [2]. For this instance, the global Lagrangian procedure succeeded in finding a superior solution to the annealed elastic net (the local Lagrangian relaxation produced the same solution as the elastic net). The annealed solution, of length 7.783, is shown in Figure (3a). The best solution, of length 7.746, found by the global Lagrangian relaxation is shown in Figure (3b). This represents a 0.5% improvement, and supplies a tour within 0.6% of the shortest simulated annealing solution found by [2], which was of length 7.70. At this level of accuracy, such an improvement is extremely encouraging. The CPU time taken by the global Lagrangian relaxation was about half that required for the elastic net.

The differences in the displayed solutions can be traced to a few changes in the south-west portion of the tour. Notice that while these changes consider rearrangements of *physically* nearby cities, they are highly *non*-local rearrangements in terms of the topological neighbours of the various cities in a tour. Furthermore, the two solutions are only distinguished by the algorithm near the end of its run. Hence other net-based algorithms which operate by severely reducing the topological search neighbourhood as the method proceeds, or by considering local city rearrangements, will fail to find the shorter solution, which seems to require the presence of global information contained in the entropy term right up to the end. On the other hand, comparison with the best simulated annealing solution shown in [2] reveals that the global topology of Figure (3b) is still completely wrong in several important respects. The entire course-featured "basin" into which the algorithm settles early in its evolution is simply misguided, and cannot be corrected later in a major way. Although the algorithm uses global information about the problem, it is unfortunately still not global enough! It's clear from this kind of behaviour that the whole class of deterministic mean-field based procedures are best viewed as efficient local optimisers, and that stochastic rearrangements must be superimposed at some later stage if further progress is to be made.

It would be of some interest to correlate the more obvious plateaux in the time evolution of the λ variables (see Figure (2)) with distinctive behaviour in the an-

nealing of the elastic net. The annealing schedule for the elastic net is determined by the occasional occurrence of narrow "fingers" in the net [2], and the subsequent need to anneal slowly at these temperatures in order to prevent the net from self-intersecting. This behaviour is related to the occurrence of multiple phase transitions as the temperature is lowered through a series of critical temperatures. These transitions manifest themselves as bifurcations in the minima of the free energy surface, which occur along the principle axes of the covariance matrix of the distribution of cities. Comprehensive discussions of this mechanism may be found in [25, 10]. It seems possible that the adaptive schedule specified automatically by a Lagrangian relaxation algorithm is able to rapidly "sense" the need to slow down the schedule at these critical temperatures, just as the mean field algorithms themselves are able to rapidly "sense" the effect of many different discrete configurations in the form of a single configuration of analogue variables. The relationship between this adaptive schedule and the cascade of phase transitions in deterministic annealing is currently under investigation.

It should be pointed out that neural net-based algorithms, including this one, do not yet outperform more classically-based heuristics and exact procedures for the TSP [26, 27], despite occasional claims to the contrary. Their main drawback is the frequent inability to obtain solutions within 1-2% of optimal. They do however, offer at least two appealing features. One is the fact that they can be decomposed into parallel algorithms with great ease. This indicates the possibility of regarding network methods as extremely efficient generators of locally-optimal TSP solutions on parallel computers. The other promising aspect is the simplicity and generality of their formulation and implementation. With minor modifications, the general approach can be applied with relatively little algorithmic investment to a wide range of important problems. The developments reported here have incorporated this philosophy as a major ingredient, and show that it can be extended further than previously anticipated.

8 Why it works - computation by valley ascent

The results discussed above show that the Lagrangian relaxation adaptations do indeed seem to offer improved computational capabilities over pure deterministic annealing methods. In this section, a schematic view of the new algorithms will be presented, which displays in a straightforward, if crude, form the main differences between the two techniques.

Recall the sequence of free energy functions shown in Figure (1), each parametrised by a different value of temperature. At any given temperature, the absolute values of the free energy minima are of no great importance in these diagrams. It is simply the ratio of the internal energy to the entropic term which determines the all-important location of minima. The new feature of the algorithms described in the previous sections can be thought of as the assignment of computational significance to the actual *value* of the free energy function as the temperature is altered.

It is therefore useful to consider the rough features of a free energy "surface" in the enlarged space $(\underline{x}, \lambda)$, as shown in Figure (4).

The foreground of Figure (4) shows schematically the form of the free energy at high values of temperature. There is of course a single global minimum in this regime. As one moves towards the background along the axis of decreasing λ, the free energy increases. Bifurcations in the free energy (i.e. phase transitions) show up as the appearance of multiple valleys in the enlarged free energy surface. Finally, when the temperature is zero, one reaches a series of saddles, or passes, each of which describes a possible solution, with entropy zero, satisfying the constraints of the original discrete problem. The internal energy term places different saddle points at different heights. Now, a deterministic annealing algorithm can be thought of as starting at the global minimum in the foreground, and ascending just one of these valleys. Every time there is a bifurcation of valleys, it is hoped that the mechanism chooses the deepest and broadest valley, and that this in turn will lead to the lowest saddle as the final answer. Two possible trajectories are displayed in the figure.

By way of contrast, a typical trajectory on this surface for one of the Lagrangian relaxation methods is also shown. The overall trajectory consists roughly of the ascent of one of the valley floors. However, oscillations about this floor now occur on the way to the final saddle point, a reflection of the interplay between the different kinetic energy components displayed in Figure (2). It is hoped that the extra degrees of freedom are a more effective way of locating the deepest valley than simple ascent along a valley floor, especially near bifurcation points. Although this is by no means guaranteed, it is certainly plausible for the following reason. Increasing the number of degrees of freedom will have the generic effect of allowing the depth and broadness of different valleys to be more fully explored. This improvement would be nullified if, in the process, too many new sub-optimal valleys and/or saddle points were to appear in the enlarged system. However, we know from (26) that no *extra* local minima are created by the Lagrangian relaxation for the elastic net, and the same is trivially true for the Hopfield/Tank style free energy functions. We can therefore be sure that the most immediate potential drawback of enlarging the number of degrees of freedom is not a problem. The worst that can happen is that one settles into a less optimal local minimum which already existed in the original deterministic annealing formulation.

Figure (4) can also be used to illustrate other issues briefly mentioned earlier. One such issue is the comparision with more usual methods of constrained optimisation, as exemplified by [13]. An important point in this regard is that the landscape depicted in Figure (4) becomes discontinuous at $\lambda = 0$. The whole world must be restricted to $\lambda > 0$, because it is only in this region that the free energy function behaves in the correct manner. In contrast, more conventional schemes allow one to pass back and forth at will over the equivalent background saddle regions, while still ensuring eventual convergence to one of the saddles. The price that must be paid for this is the use of much simpler and more restrictive constraint functions, which ultimately lead to less optimal saddle points. Another point is that the λ variable(s) are monotonically decreasing in the new algorithms. All oscillations are restricted

to the $\underline{x}$ variables. This situation is also unlike more usual constrained optimisation methods. It is, however, exactly like standard annealing in this respect, although the use of several multipliers certain allows for different *rates* of annealing. It might be interesting and worthwhile to try to develop a Lagrangian-style method which did allow occasional increases in temperature in an automatic way, thus mimicing the occasional *adhoc* use of such expedients in simulated annealing.

9 Conclusions

Both deterministic and simulated annealing methods work by constructing artificial statistical physics models to represent a chosen optimisation problem at different non-zero temperatures, with associated non-zero "entropies", in order to make the problem as "convex" as possible. However, the original problem is only truly represented by the final system at zero temperature, with an entropy of zero. By viewing the vanishing of the entropy function as a constraint upon possible solutions, it has been shown how a bridge can be formed to powerful methods of constrained optimisation, especially Lagrangian relaxation. The basic idea of Lagrangian relaxation is to construct a "soft" mechanism which allows a constraint to be fulfilled gradually, whilst ensuring that it ultimately be fulfilled exactly. In essence, this is the task performed somewhat awkwardly by the temperature parameter, with respect to the entropy constraint, in both deterministic and stochastic annealing schemes. However, the deterministic mean field theory approximations allow this process to be taken a step further by offering complicated, but closed form, entropic expressions which can serve as perfectly reasonable constraint functions within an enlarged relaxation scheme.

The result of this point of view is a simple yet effective framework for systematically generalising any deterministic annealing algorithm described by a mean field theory approximation of the form (1) into a single homogeneous parallel analogue relaxation procedure. The same framework generates a rational guess for an optimisation algorithm when the appropriate approximation is of the form (20), in which the natural candidate for a "Lagrange variable" is no longer even a true multiplier of the constraint function. The need to conduct several relaxations of a dynamical system at different temperature values can therefore be replaced in these cases by the relaxation of a single dynamical system, with resulting improvements in efficiency and accuracy. Furthermore, the resulting algorithms now mimic in a rough way the notion of a temperature being lowered adaptively in simulated annealing [28].

In the case of the elastic net solution to the TSP, the ensuing adaptation can be shown to provide syntactically correct solutions, and I find in fact that it substantially improves the speed (and to a smaller extent the accuracy) of that method. However, it is important to stress that although the TSP was chosen as the test problem for the algorithms described above, there is nothing special about this particular problem as far as the structure of the algorithms is concerned, other than its status as an important bench-mark. Its main relevant feature is simply the ability

to embed the problem in a general statistical physics framework in such a way that self-consistent mean-field theory approximations can be made to describe the configuration space at low temperatures. This characteristic is shared by many different combinatorial problems [5, 9, 4, 3, 10], and in each case a candidate algorithm of the kind described above may be constructed. Problems of this type are particularly prevalent in the field of computational vision, and a comprehensive discussion of them in relation to deterministic annealing can be found in [4]. The methods described above should also be applicable to other techniques in computational vision, such as graduated non-convexity [19], which are similar in spirit to deterministic annealing, albeit without the same technical grounding in statistical mechanics.

Of course, it is by no means certain that every mean field approximation of the form (20) will enjoy the same providential properties as the elastic net. Nevertheless, the procedure generates candidate algorithms whose properties ought to be worth investigation in general. The procedure is not limited to functions derived from mean field theory, but this seems to be a particularly promising avenue of inquiry, as it frees us from the need to artificially guess objective functions for problems that are somewhat opaque to begin with. As a final observation, the idea of using entropy as a constraint function can also be applied in a somewhat wider context. For example, it can be used in the design of objective functions and architectures for neural networks and other learning mechanisms which seek to generalise, a question which touches on the related issues of regularisation and the solution of ill-posed problems in general [29].

Acknowledgements

I would like to thank Mark Muldoon and Jeff Davitz for many useful discussions, Alan Lapedes and David Sharp for their interest and encouragement, and Geoffrey Fox for helpful comments upon an earlier version of the manuscript.

Appendix 1 - Convergence Properties of Lagrangian Relaxations Based Upon Hopfield/Tank Style Objective Functions

In this Appendix I discuss in some detail the convergence properties of the Lagrangian relaxation mechanisms obtained from deterministic annealing objective functions of the form

$$F(\underline{x}) = U(\underline{x}) - TS(\underline{x}) \tag{27}$$

Examples of such functions are the Hopfield/Tank method for solving the TSP [1], and the mean field theory learning algorithms derived as deterministic approximations to Boltzmann machines [9]. The more complicated case of objective functions related to the elastic net will be discussed in Appendix 2.

As these functions stand, the temperature parameter T can be interpreted as a true Lagrange multiplier. However, since we want to endow this multiplier with dynamics, we consider the slightly modified objective function

$$\hat{F}(\underline{x}, \lambda) = U(\underline{x}) - \lambda S(\underline{x}) + c \ln \lambda \tag{28}$$

where $c > 0$. The algorithm itself is given by

$$\begin{aligned} \dot{x}_i &= -\nabla_{x_i} \hat{F}(\underline{x}, \lambda) &&= -\nabla_{x_i} U(\underline{x}) + \lambda \nabla_{x_i} S(\underline{x}) \\ \dot{\lambda} &= +\nabla_{\lambda} \hat{F}(\underline{x}, \lambda) &&= -S(\underline{x}) + c/\lambda \end{aligned} \tag{29}$$

It is perhaps worth stressing that this Appendix considers only the issue of local convergence. In order for the method to be genuinely useful, it is necessary to ensure that the value of c be chosen small enough that the entropic constraint is satisfied to an appropriate accuracy at the fixed point, but this is a separate issue from that of convergence itself.

Now consider the energy function

$$E = \frac{1}{2} \sum_i \dot{x}_i{}^2 + \frac{1}{2} \dot{\lambda}^2 \tag{30}$$

The time derivative of this function is given by

$$\begin{aligned} dE/dt &= \sum_i \dot{x}_i \ddot{x}_i + \dot{\lambda} \ddot{\lambda} \\ &= \sum_i \dot{x}_i \left[\sum_j \frac{\partial \dot{x}_i}{\partial x_j} \dot{x}_j + \frac{\partial \dot{x}_i}{\partial \lambda} \dot{\lambda} \right] + \dot{\lambda} \left[\sum_j \frac{\partial \dot{\lambda}}{\partial x_j} \dot{x}_j + \frac{\partial \dot{\lambda}}{\partial \lambda} \dot{\lambda} \right] \\ &= \sum_i \dot{x}_i \left[\sum_j -\nabla^2_{x_i x_j} \hat{F}(\underline{x}, \lambda) \dot{x}_j - \nabla^2_{x_i \lambda} \hat{F}(\underline{x}, \lambda) \dot{\lambda} \right] \\ &\quad + \dot{\lambda} \left[\sum_j \nabla^2_{\lambda x_j} \hat{F}(\underline{x}, \lambda) \dot{x}_j + \nabla^2_{\lambda} \hat{F}(\underline{x}, \lambda) \dot{\lambda} \right] \\ &= -\sum_i \sum_j \dot{x}_i \nabla^2_{x_i x_j} \hat{F}(\underline{x}, \lambda) \dot{x}_j - c(\dot{\lambda}/\lambda)^2 \end{aligned} \tag{31}$$

The central point in the whole analysis is contained in the last step above. The two terms involving mixed derivatives of $\hat{F}(\underline{x}, \lambda)$ with respect to $\underline{x}$ and λ are equal in magnitude, and cancel each other out. This situation arises directly because of the opposite signs of the two gradients in the algorithm described by (29). In addition, the function $\hat{F}(\underline{x}, \lambda)$ has been constructed to be concave in λ, so that the dynamics of λ also contributes to the lowering of E. As a result of this concavity, and the cancellation of mixed derivatives, the only regions of the solution space in which E is not guaranteed to be decreasing with time are those in which the function $\hat{F}(\underline{x}, \lambda)$ fails to be convex with respect to $\underline{x}$. This convexity structure is of course independent of the new term $c \ln \lambda$ in $\hat{F}$. Because these regions of non-convexity are typically small, and in addition, do not seem to occur near desired solutions in the deterministic annealing free energy functions, it can be expected that E will decrease monotonically with time, for large enough times, in almost all cases.

The formalism can easily be altered to accomodate several different constraints, each with a different multiplier λ_a. This comes about because the entropy is nearly always a sum of additive components. A typical free energy function can therefore be written as

$$\hat{F}(\underline{x}, \underline{\lambda}) = U(\underline{x}) - \sum_a^N \lambda_a S_a(\underline{x}) + c \sum_a^N \ln \lambda_a \tag{32}$$

where N is the number of groups one chooses to split the entropy into. The algorithm itself is now given by

$$\begin{aligned} \dot{x}_i &= -\nabla_{x_i} \hat{F}(\underline{x}, \underline{\lambda}) &&= -\nabla_{x_i} U(\underline{x}) + \sum_a \lambda_a \nabla_{x_i} S_a(\underline{x}) \\ \dot{\lambda}_a &= +\nabla_{\lambda_a} \hat{F}(\underline{x}, \underline{\lambda}) &&= -S_a(\underline{x}) + c/\lambda_a \end{aligned} \tag{33}$$

The appropriate energy function is now

$$E = \frac{1}{2} \sum_i \dot{x}_i^{\,2} + \frac{1}{2} \sum_a \dot{\lambda}_a^{\,2} \tag{34}$$

for which

$$dE/dt = -\sum_i \sum_j \dot{x}_i \, \nabla^2_{x_i x_j} \hat{F}(\underline{x}, \underline{\lambda}) \, \dot{x}_j - c \sum_a (\dot{\lambda}_a / \lambda_a)^2 \tag{35}$$

Although the convergence of the algorithm does not require it, the dynamics of λ may also be scaled with respect to the $\underline{x}$ dynamics by a positive factor ϵ. The energy function E can easily be altered to take account of this change also. The reason for this factor is to prevent the algorithm from annealing too quickly. It is a tunable parameter which must be set at the beginning of the algorithm. It represents the remaining arbitrary component of the former annealing process.

The function E has been interpreted in Section 5, and in Figure (2), as the sum of two different "kinetic energies" associated with $\underline{x}$ and λ respectively. It was noted by [13] that the λ component of their algorithm could be interpreted in a natural way as a "potential energy" function, in conjunction with a second-order differential equation describing the $\underline{x}$ dynamics. Such an interpretation is also possible here.

Consider for example the case of a single global λ. By combining the two parts of (29), one obtains the following second-order equation in $\underline{x}$ for given λ:

$$\ddot{x}_i + \sum_j \nabla^2_{x_i x_j} \hat{F}(\underline{x}, \lambda)\, \dot{x}_j + \nabla_{x_i} V(\underline{x}, \lambda) = 0 \tag{36}$$

where

$$V(\underline{x}, \lambda) = \frac{1}{2}(-S(\underline{x}) + c/\lambda)^2 \tag{37}$$

This is the equation for a damped mass system with inertia, a damping term, and an external force given by the negative gradient of the "potential energy" $V(\underline{x}, \lambda)$. The function E can now be written as the sum of a kinetic term T and a potential term V:

$$E = T + V = \frac{1}{2}\sum_i \dot{x}_i{}^2 + \frac{1}{2}(-S(\underline{x}) + c/\lambda)^2 \tag{38}$$

This modification of the elegant construction of [13] shows how the extra degrees of freedom in the new algorithms can emulate the effects of inertia on the variables $\underline{x}$. In the current case, this interpretation might perhaps be considered a little strained because of the appearance of λ in the potential energy function. In the elastic net procedures, λ plays an even more complicated role. From this point of view, the dual kinetic energy interpretation, treating $\underline{x}$ and λ on a single homogeneous footing, perhaps shows more of the relevant computational flavour of an interplay between kinetic components, and has therefore been stressed in the body of the paper.

A limited amount can be proved concerning the convergence of (29) or (33). I describe here a result which ensures local convergence. Consider the case of several multipliers. Suppose there are n variables x_i and m variables λ_i. Assemble them into vectors, denoted simply by x and λ. The basic approach, closely following the treatment of [17], consists of regarding the algorithm (33) as an iterative process specified by

$$\begin{aligned} x_{k+1} &= x_k - \eta \nabla_x \hat{F}(x, \lambda) \\ \lambda_{k+1} &= \lambda_k + \eta \nabla_\lambda \hat{F}(x, \lambda) \end{aligned} \tag{39}$$

where $\eta > 0$ is a scalar stepsize. Suppose that (x^*, λ^*) is a fixed point of (33), and that $\hat{F}(x, \lambda)$ is twice-differentiable in an open set around this fixed point. Then the following proposition, a generalisation of Proposition 4.23 in [17], holds.

Proposition : If the matrix $\nabla^2_{xx} \hat{F}(x^*, \lambda^*)$ is positive definite (i.e. if the fixed point is a local minimum of $\hat{F}$ with respect to x), then one can find small enough η and some initial pair (x_0, λ_0) so that the sequence (x_k, λ_k) generated by (39) converges to (x^*, λ^*). Furthermore, the rate of convergence is at least linear.

Proof : The proof proceeds by considering the mapping $G : R^{n+m} \rightarrow R^{n+m}$ defined by the above iterations. This map can be represented as a column vector

$$G(x, \lambda) = \begin{bmatrix} x - \eta \nabla_x \hat{F}(x, \lambda) \\ \lambda + \eta \nabla_\lambda \hat{F}(x, \lambda) \end{bmatrix} \tag{40}$$

The pair (x^*, λ^*) is obviously a fixed point of the map. Now consider the gradient of this map at the fixed point, which can be represented as an $(n+m) \times (n+m)$ matrix:

$$\nabla G(x^*, \lambda^*) = I - \eta B \tag{41}$$

where the matrix B is given by

$$B = \begin{bmatrix} \nabla^2_{xx} \hat{F}(x^*, \lambda^*) & \nabla^2_{x\lambda} \hat{F}(x^*, \lambda^*) \\ -\nabla^2_{\lambda x} \hat{F}(x^*, \lambda^*) & -\nabla^2_{\lambda\lambda} \hat{F}(x^*, \lambda^*) \end{bmatrix} \tag{42}$$

Notice that the bottom right-hand component of this matrix is diagonal for the objective functions (32), and has only positive entries, so it is trivially positive definite.

The important thing about B is that the real parts of each of its eigenvalues are all strictly positive. To see this, let β be an eigenvalue of B, with corresponding eigenvector $b \neq 0$. Denote by $b^\dagger = (z^\dagger, w^\dagger) \neq (0,0)$ the Hermitian conjugate of b. Then by definition, we have

$$Re(b^\dagger B b) = Re(\beta)(|z|^2 + |w|^2) \tag{43}$$

On the other hand, we also know by considering the form of B that

$$Re(b^\dagger B b) = Re(z^\dagger \nabla^2_{xx} \hat{F}(x^*, \lambda^*) z) - Re(w^\dagger \nabla^2_{\lambda\lambda} \hat{F}(x^*, \lambda^*) w) \tag{44}$$

The last line comes about because of the exact cancellation of the mixed derivatives in the non-diagonal components of B. It is precisely the same feature that is used to obtain the suggestive form of dE/dt above. Now, due to the positive definiteness of $\nabla^2_{xx} \hat{F}(x^*, \lambda^*)$ (by assumption) and the negative definiteness of $\nabla^2_{\lambda\lambda} \hat{F}(x^*, \lambda^*)$ (by construction), this last expression must be strictly positive: otherwise, we would violate the assumption that $b \neq 0$. Hence from the first expression, $Re(\beta) > 0$.

The strict positivity of the real parts of the eigenvalues of B implies in turn that η can always be scaled so that all the eigenvalues of $\nabla G(x^*, \lambda^*) = I - \eta B$ lie strictly within the unit circle of the complex plane. By a theorem which Bertsekas attributes to Ostrowski [17], this condition guarantees the existence of an open set S such that if $(x^0, \lambda^0) \epsilon S$, then the sequence generated by (39) remains entirely in S, and converges linearly to the fixed point (x^*, λ^*). Q.E.D.

This result demonstrates the local convergence of the algorithm (39) to a fixed point. In the context of deterministic annealing based objective functions, it means that a region of the solution space always exists for which the non-convex regions of the free energy cannot interfere with the Lagrangian relaxation process to spoil the convergence of the new algorithm. The only requirements are the convexity of the free energy with respect to x at the fixed point and the concavity of the barrier functions with respect to λ at the fixed point (and by continuity, in a neighbourhood about this point). These conditions are enough by themselves to ensure the existence of an open set within which the iterations (39) are bounded, and for which the convex regions of (32) with respect to x enforce convergence to a fixed point. Of course, the proposition does not prove global convergence, i.e. that the algorithm will always find such a region, no matter where it starts from.

Appendix 2 - Convergence Properties of Lagrangian Relaxations Based Upon Elastic Net Style Objective Functions

In this appendix I discuss the computational details of the elastic net Lagrangian relaxations introduced in Section 6, and subsequently used to obtain the TSP results quoted in Section 7. Although I will discuss the TSP implementation, there has been substantial recent work concerned with the use of extremely similar schemes to perform automatic clustering [10], as well as in vision problems [3, 4]. The formalism described here applies with little or no modification to all of these problems. I will deal in detail with the serendipitous constraint-satisfaction property (26) of Section 6, and will outline briefly the formal convergence properties of the method, which are quite similar to those discussed in Appendix 1.

To begin with consider the elastic net objective function, but with a different temperature parameter λ_a associated with each city a. Thus, the objective function now reads

$$F(\underline{x}, \underline{\lambda}) = U(\underline{x}) - \sum_a^N \lambda_a S_a(\underline{x}, \lambda_a) \tag{45}$$

where

$$U(\underline{x}) = \gamma \sum_i^M |x_i - x_{i+1}|^2 \tag{46}$$

and

$$S_a(\underline{x}, \lambda_a) = \alpha \ln \sum_i^M e^{-|x_i - x_a|^2/2\lambda_a^2} \tag{47}$$

The local multiplier Lagrangian adaptation is given by the equations

$$\begin{aligned} \dot{x}_i &= -\nabla_{x_i} F(\underline{x}, \underline{\lambda_a}) = -\nabla_{x_i} U(\underline{x}) + \sum_a \lambda_a \nabla_{x_i} S_a(\underline{x}, \lambda_a) \\ &= -2\gamma(2x_i - x_{i+1} - x_{i-1}) - \alpha \sum_a \frac{(x_i - x_a)}{\lambda_a} \frac{e^{-|x_i - x_a|^2/2\lambda_a^2}}{\sum_j e^{-|x_j - x_a|^2/2\lambda_a^2}} \end{aligned} \tag{48}$$

and

$$\begin{aligned} \dot{\lambda}_a &= +\epsilon \nabla_{\lambda_a} F(\underline{x}, \underline{\lambda}) = -\epsilon[S_a(\underline{x}, \lambda_a) + \lambda_a \nabla_{\lambda_a} S_a(\underline{x}, \lambda_a) \\ &= -\epsilon\alpha \ln \sum_j e^{-|x_j - x_a|^2/2\lambda_a^2} - \epsilon\alpha \sum_i \frac{(x_i - x_a)^2}{\lambda_a^2} \frac{e^{-(x_i - x_a)^2/2\lambda_a^2}}{\sum_j e^{-|x_j - x_a|^2/2\lambda_a^2}} \end{aligned} \tag{49}$$

This was the algorithm used to obtain the results shown in Section 7. Simple gradient descent was used as the fundamental update step. At any given step, numerical stability considerations suggest that the increments of the variables be modified by the temperature parameter. In the local multiplier case, this was accomplished by using λ_{min}, the smallest value of the set of $\lambda's$ at a given time step. Thus

$$\begin{aligned} x_i(t+1) &= x_i(t) + \eta \lambda_{min}(t)\ \dot{x}_i \\ \lambda_a(t+1) &= \lambda_a(t) + \eta \lambda_{min}(t)\ \dot{\lambda}_a \end{aligned} \tag{50}$$

where η is a fixed rate constant, typically set to 1.

Before considering the convergence of this algorithm, I will first show that at a fixed point of the λ_a dynamics, the appropriate constraints for a legal TSP solution are satisfied. Dropping the subscript on λ to avoid notational clutter, and defining

$$b_i = e^{-|x_i - x_a|^2/2\lambda^2} \tag{51}$$

we have

$$-\dot{\lambda} \propto \ln \sum_i b_i - 2\frac{\sum_i b_i \ln b_i}{\sum_i b_i} \tag{52}$$

Now as long as $\lambda > 0$, we know that $\sum_i b_i \neq 0$. Hence the numerator of this expression can be written as

$$\sum_i b_i [\ln \sum_i b_i - 2 \ln b_i] \tag{53}$$

Since $0 < b_i < 1$, we know immediately that as long as $\sum b_i \geq 1$, the expression is strictly positive. On the other hand, if $\sum b_i < 1$, we have

$$-2 \ln b_i > |\ln \sum_i b_i| \quad \forall i \tag{54}$$

so the expression is still positive. This establishes that provided $\lambda > 0$, $\dot{\lambda}$ is always acting to decrease λ. Furthermore, the expression can only be equal to zero, thus specifying a fixed point of the λ dynamics, when some $b_k = 1$, with all other $b_i = 0$. This in turn can only occur if both $x_k = x_a$ and $\lambda = 0$, which establishes that the fixed point satisfies the correct syntax of the TSP.

It's interesting to note that it is perfectly possible to satisfy the entropic constraint $S(\underline{x}, \lambda) = 0$ at some non-zero value of λ, without the correct solution syntax being satisfied. The second term in the λ dynamics therefore plays a vital role in these methods, ensuring that such a sub-optimal situation cannot possibly be a fixed point of the algorithm. This feature highlights the fact that the full mean field theory free energy function is the best starting point for constructing a Lagrangian procedure. In particular, it suggests that every occurrence of the temperature parameter in the free energy should be wholeheartedly converted into a variable λ, even though this means that λ can no longer be truly considered a Lagrange multiplier. If one insisted on updating each λ as a true multiplier, i.e. proportionally only to $S(\underline{x}, \lambda)$, there is no guarantee that unwanted fixed points corresponding to illegal solutions could not arise.

Finally, consider the convergence of the algorithm described above. The appropriate energy function is now

$$E = \frac{1}{2} \sum_i \dot{x}_i^{\,2} + \frac{1}{2\epsilon} \sum_a \dot{\lambda}_a^{\,2} \tag{55}$$

From the rather general discussion of the time derivative of the equivalent function in Appendix 1, it's clear that the important cancellation involving mixed derivatives

of the x_i and λ_a variables does not depend in any way on λ_a appearing merely as a multiplier in $F(\underline{x},\underline{\lambda})$. In fact, one can immediately write down for the current case

$$dE/dt = -\sum_i \sum_j \dot{x}_i \, \nabla^2_{x_i x_j} \hat{F}(\underline{x},\underline{\lambda}) \, \dot{x}_j + \sum_a \sum_b \dot{\lambda}_a \, \nabla^2_{\lambda_a \lambda_b} \hat{F}(\underline{x},\underline{\lambda}) \, \dot{\lambda}_b \qquad (56)$$

Now, because each entropy component depends upon just one λ_a, the second matrix in this expression is diagonal, leading to

$$dE/dt = -\sum_i \sum_j \dot{x}_i \, \nabla^2_{x_i x_j} F(\underline{x},\lambda) \, \dot{x}_j - \sum_a [2\nabla_{\lambda_a} S(\underline{x},\lambda_a) + \lambda_a \nabla^2_{\lambda_a} S(\underline{x},\lambda_a)] \dot{\lambda_a}^2 \qquad (57)$$

The term containing the second derivative of $S(\underline{x},\lambda_a)$ with respect to λ_a is of the incorrect sign for lowering E. The simple expedient of terminating a simulation before this term overwhelmed the simple gradient of $S(\underline{x},\lambda_a)$ was found to yield perfectly good results in the problems tackled. However, if desired a barrier term can also be added to ensure convergence. Consider, for example, a modified objective function of the form

$$F(\underline{x},\underline{\lambda}) = U(\underline{x}) - \sum_a^N \lambda_a S_a(\underline{x},\lambda_a) + \sum_a^N f_a(\lambda_a) \qquad (58)$$

where possible $f_a(\lambda_a)$ will be described shortly. We now have

$$dE/dt = -\sum_i \sum_j \dot{x}_i \, \nabla^2_{x_i x_j} F(\underline{x},\lambda) \, \dot{x}_j - \sum_a [2\nabla_{\lambda_a} S(\underline{x},\lambda_a) + \lambda_a \nabla^2_{\lambda_a} S(\underline{x},\lambda_a) - \nabla^2_{\lambda_a} f_a(\lambda_a)] \dot{\lambda_a}^2 \qquad (59)$$

Hence by choosing each $f_a(\lambda_a)$ to be suitably concave so as to overwhelm $\nabla^2_{\lambda_a} S(\underline{x},\lambda_a)$ at small λ_a, it can be expected that E will decrease monotonically with time at large enough times. More rigorously, suitable concavity ensures that the conditions for the Proposition in Appendix 1 are fulfilled in this case also, hence local convergence is guaranteed. Of course, one must be careful to make sure that this occurs at low enough λ_a to leave the resulting method computationally useful.

References

[1] J.J. Hopfield and D.W. Tank. Neural Computation of Decisions in Optimization Problems. *Biol. Cybern.*, 52:141–152, 1985.

[2] R. Durbin and D. Willshaw. An Analogue Approach to the Travelling Salesman Problem using an Elastic Net Method. *Nature*, 326:689–691, 1987.

[3] D. Geiger and F. Girosi. Coupled Markov Random Fields and Mean Field Theory. In D. Touretzky, editor, *Advances in Neural Information Processing Systems 2*, pages 660–667. Morgan Kaufmann, 1990.

[4] A.L. Yuille. Generalised Deformable Models, Statistical Physics, and Matching Problems. *Neural Comp.*, 2:1–24, 1990.

[5] P.D. Simic. Constrained Nets for Graph Matching and other Quadratic Assignment Problems. *Neural Comp.*, 3:268–281, 1991.

[6] C. Peterson and B. Soderberg. A New Method for Mapping Optimization Problems onto Neural Networks. *Int. J. Neural Syst.*, 1:3–22, 1989.

[7] J.J. Hopfield and D.W. Tank. Computing with Neural Circuits: A Model. *Science*, 233:625–633, 1986.

[8] P.D. Simic. Statistical Mechanics as the Underlying Theory of "Elastic" and "Neural" Optimisations. *NETWORK: Comp. Neural Syst.*, 1:89–103, 1990.

[9] C. Peterson and J.R. Anderson. A Mean Field Learning Algorithm for Neural Networks. *Complex Systems*, 1:995–1019, 1987.

[10] K. Rose, E. Gurewitz, and G.C. Fox. Statistical Mechanics and Phase Transitions in Clustering. *Phys. Rev. Lett.*, 65:945–948, 1990.

[11] S. Kirkpatrick, C.D. Gelatt Jr., and M.P. Vecchi. Optimization by Simulated Annealing. *Science*, 220:671–680, 1983.

[12] S. Geman and D. Geman. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 6:721–741, 1984.

[13] J.C. Platt and A.H. Barr. Constrained Differential Optimization. In D.Z. Anderson, editor, *Neural Information Processing Systems*, pages 612–621. American Institute of Physics, 1988.

[14] A.G. Tsirukis, G.V. Reklaitis, and M.F. Tenorio. Nonlinear Optimization Using Generalised Hopfield Networks. *Neural Comp.*, 1:511–521, 1989.

[15] E. Mjolsness and C. Garrett. Algebraic Transformations of Objective Functions. *Neural Networks*, 3:651–669, 1990.

[16] K.J. Arrow, L. Hurwicz, and H. Uzawa. *Studies in Linear and Nonlinear Programming.* Stanford University Press, 1958.

[17] D.P. Bertsekas. *Constrained Optimization and Lagrange Multiplier Methods.* Academic Press, 1982. See especially Chapter 4.

[18] E. Wasserstrom. Numerical Solutions by the Continuation Method. *SIAM Review*, 15:89–119, 1973.

[19] A. Blake and A. Zisserman. *Visual Reconstruction.* MIT Press, 1987.

[20] S.R. White. Concepts of Scale in Simulated Annealing. In *Proc. International Conf. on Computer Design*, pages 646–651, 1984.

[21] E.H.L. Aarts, J.H.M. Korst, and P.J.M. van Laarhoven. A Quantitative Analysis of the Simulated Annealing Algorithm: A Case Study for the Travelling Salesman Problem. *J. Stat. Phys.*, 50:187–206, 1988.

[22] D.J. Burr. An Improved Elastic Net Method for the Travelling Salesman Problem. In *IEEE Second International Conference on Neural Networks*, pages I–69–76, 1988.

[23] G. Di Pillo and L. Grippo. A New Class of Augmented Lagrangians in Nonlinear Programming. *SIAM J. Control and Optimization*, 17:618–628, 1979.

[24] E. Wacholder, J. Han, and R.C. Mann. An Extension of the Hopfield-Tank Model for Solution of the Multiple Travelling Salesmen Problem. In *IEEE Second International Conference on Neural Networks*, pages II–305–324, 1988.

[25] R. Durbin, R. Szeliski, and A.L. Yuille. An Analysis of the Elastic Net Approach to the Travelling Salesman Problem. *Neural Comp.*, 1:348–358, 1989.

[26] D. Johnson. More Approaches to the Travelling Salesman Guide. *Nature*, 330:525, 1987.

[27] D.L. Miller and J.F. Pekny. Exact Solution of Large Asymmetric Travelling Salesman Problems. *Science*, 251:754–761, 1991.

[28] P. Stolorz. Merging Constrained Optimisation with Deterministic Annealing to "Solve" Combinatorially Hard Problems. In J.E. Moody, S.J. Hanson, and R.P. Lippmann, editors, *Advances in Neural Information Processing Systems 4, in press.* Morgan Kaufmann, 1992.

[29] P. Stolorz. Analog Entropy as a Constraint in Adaptive Learning, Optimization and Regularization. Technical report, Los Alamos National Laboratory and Santa Fe Institute, 1992.

Figure Captions

Figure 1. Schematic representation of a generic deterministic annealing algorithm.

(a) A convex objective function occurring for some large value of annealing/temperature parameter T. An associated dynamical system relaxes to the unique minimum.

(b) As T is lowered, the fixed point solution bifurcates. It is hoped that the system chooses the lefthand local minimum, which is which is wider and deeper than the righthand minumum.

(c) As T is lowered still further, the lefthand minimum becomes substantially deeper than the righthand minimum.

(d) Finally, as T approaches zero, more local minima appear. Once again, it is hoped that the algorithm chooses the deeper of the local minima, which is then taken as the solution to the original discrete problem.

Figure 2. A plot of the evolution of "kinetic energies" associated with the system variables during a typical Lagrangian relaxation algorithm. Shown is the total energy $E = \frac{1}{2}\sum_i \dot{x}_i^2 + \frac{1}{2}\dot{\lambda}^2$ (solid curve), together with the individual kinetic components associated with $\underline{x}$ (dash-dotted curve) and λ (dotted curve). It can be seen that these energies all decrease asymptotically, despite regions in which they may temporarily increase.

Figure 3. Two different solutions to the 100-city TSP problem from [2].

(a) The solution found by the elastic net algorithm. It has length 7.783.

(b) The solution found by the global Lagrangian relaxation of Section 7. It has length 7.746.

Solution (b) is roughly 0.5% shorter than (a), although it is still 0.6% longer than the shortest simulated annealing tour found by [2] (of length 7.70).

Figure 4. Schematic representation of the free energy "surface" of a generic deterministic annealing free energy function, obtained by plotting the various functions in Figure (1) along a temperature axis. Shown on the surface are typical trajectories which might be taken along valley floors by a deterministic annealing algorithm (dash-dotted curves) and by a Lagrangian relaxation adaptation (dotted curve), as discussed in Section 8. The Lagrangian relaxation succeeds in locating the lowest saddle point in the background.

Figure 1

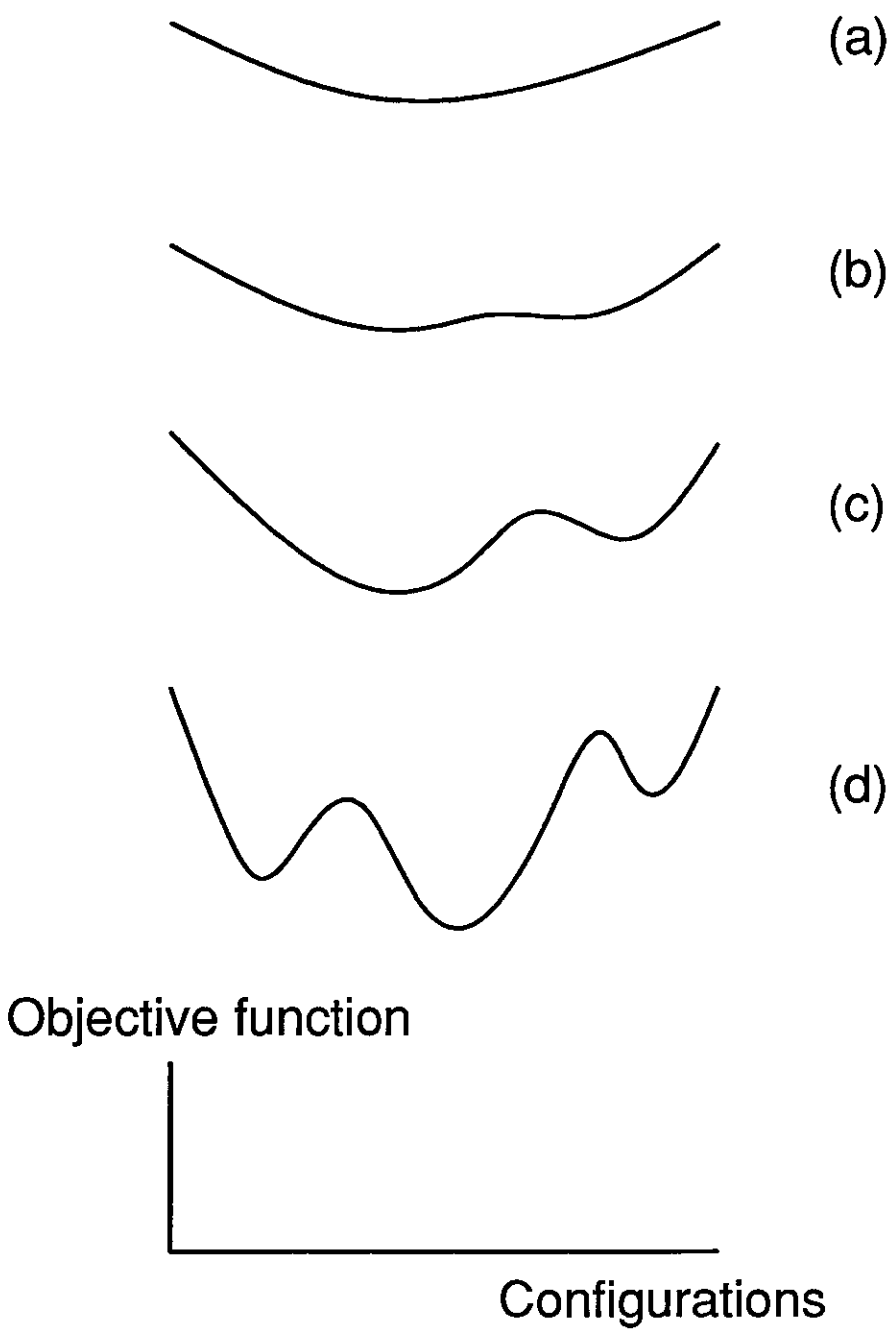

Figure 2

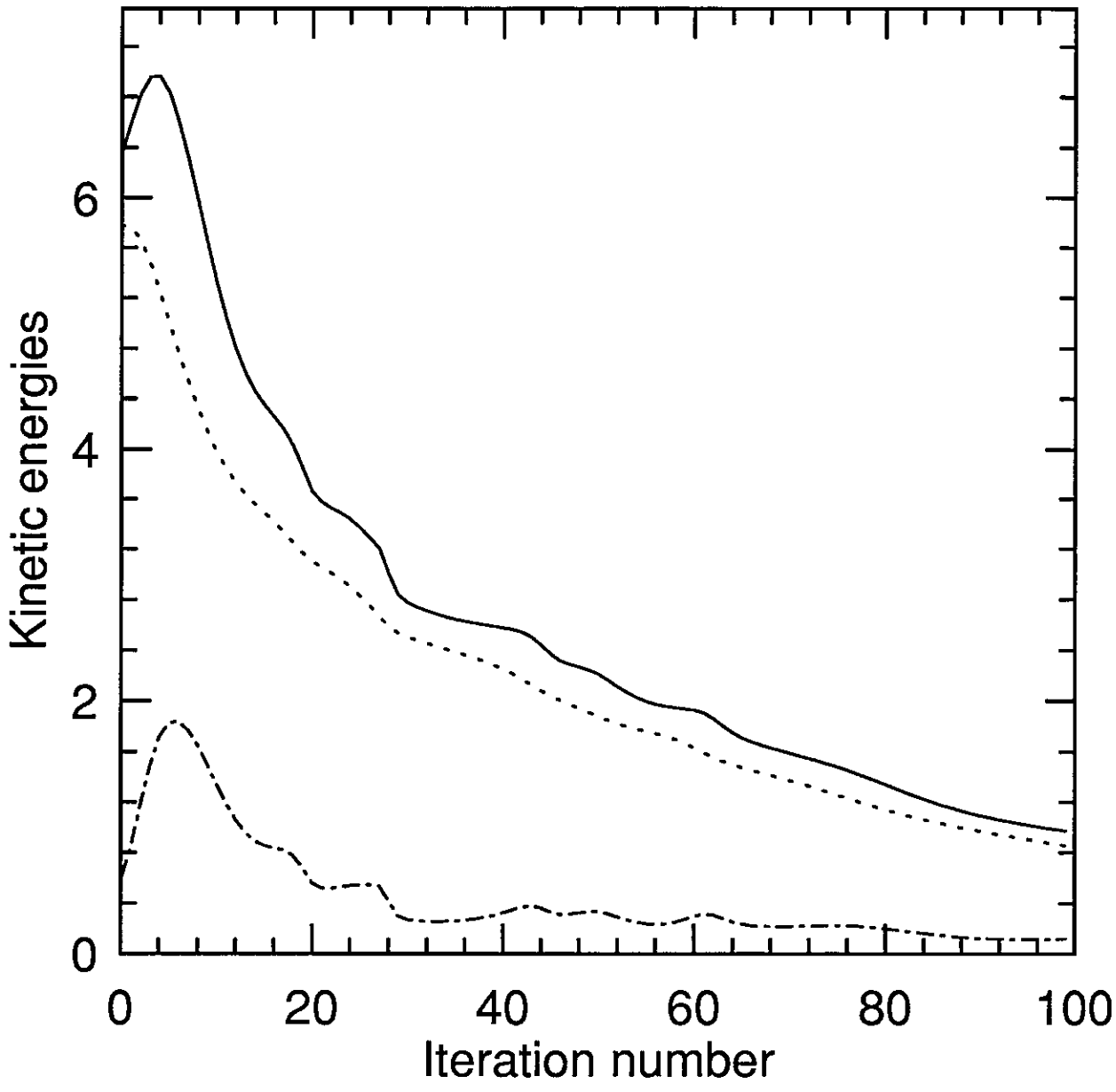

Figure 3(a)

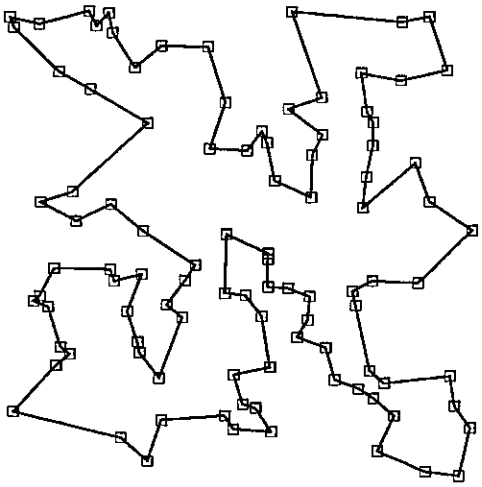

Figure 3(b)

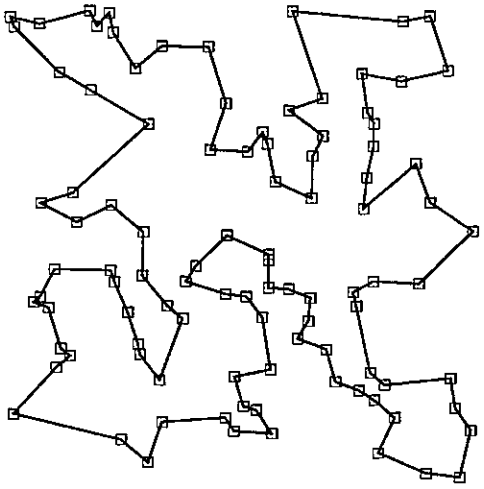

Figure 4