Physical Computation

Geoffrey C. Fox
Northeast Parallel Architectures Center
111 College Place
Syracuse University
Syracuse, New York 13244-4100

August 17, 1995

Abstract

Physical Computation embraces a variety of physical analogies used to tackle non-traditional problems. We describe Monte Carlo and deterministic methods, including simulated annealing and neural networks. Applications include economic change in Eastern Europe, the travelling salesman problem, vehicle navigation, track finding, and parallel computer load balancing. We show how different problems are suitable for the different various approaches to optimization—there is no universally applicable method.

1 Introduction

Physical Computation encompasses a variety of ideas that can be loosely classified as the use of physical analogies or methods from the physical sciences to problems outside their normal domain of applicability. One example is the use of simulated annealing (an idea from physics) to chip routing and placement [Kirkpatrick:83a] (a problem in optimization). Another is the use of neural networks (an idea from biology) in learning and pattern recognition (problems in computer science, robotics, etc.). Again, we will show an example of cellular automata (an idea from physics) applied to social change in Eastern Europe (a problem from economics).

The Santa Fe Institute and its articulate spokesman Gell Mann has defined the concept of a complex system.

> **"The Need for New Options in Education and Research"**
>
> "The transformation of society by the scientific revolution of the 19th and 20th centuries is about to be overshadowed by even more sweeping changes arising from a growing ability to understand the complex mechanisms which are central to human concerns. *The technology base of the new revolution will be provided by almost unimaginably powerful computers together with the mathematical and experimental tools and associated software which are essential to achieving an understanding of complexity. Complex systems contain large numbers of coupled elements. The strength of the interaction between elements varies with time, space, and the nature of the surrounding environment which may also change with time.* Such systems can adapt to their environments. Examples of adaptive, complex systems include biological evolution, learning, and neural processes, intelligent computers, protein chemistry, much of pathology, and medicine, human behavior, and economics."
>
> "It is becoming increasingly evident that understanding complex systems demands mutually supportive research conducted by scholars representative of a broad spectrum of the intellectual community ranging from mathematics and the natural sciences to the humanities. Society must find new ways to nurture the necessary convergences of academic disciplines and other critical resources. Present-day academic institutions are not well designed to meet this increasingly urgent need".

We can view physical computation as the use of physical methods to describe general complex systems. Note that as used here, "physical" means "pertaining to nature" and is broader than just physics. However, this field is particularly relevant as physics has studied large complex systems, albeit those obeying Newton's and other basic laws of physics. For example, in thermodynamics, we find a theory describing large systems in a way that is insensitive to irrelevant microscopic detail. A key feature of physical computation is approaches that naturally tackle large problems; we can anticipate a growing role for physical computation as the growing power of computers allows the simulation of larger and larger systems. Some traditional methods (for discrete optimization) have time complexities that scale exponentially in problem size while physical computation is often essentially linear. The factor of a thousand in computer performance improvement, expected by the year 2000, makes little impact on an exponentially complex algorithm; however, it implies a revolution for an algorithm with linear time complexity.

Optimization is an important applicant of physical computation and Simic originally introduced the term physical optimization [Simic:90a] [Fox:91b]. Indeed, most physics laws can be formulated variationally as an optimization problem while nature is also involved in optimization. Thus, in the long term, the evolution of the human race is perhaps maximizing some combination of survival and happiness. In the short term, we interpret visual and other sensor information optimally according to our prejudices and experience. These last two analogies lead respectively to genetic and neural net approaches to optimization. Simulated annealing minimizes the (free) energy by Monte Carlo methods and later we will see elastic net and deterministic annealing approaches to optimization. These correspond to non-statistical variational methods from physics applied to optimization. Maximum entropy or information theory leads to similar approaches based on analogies from an engineering field.

Above, we listed several ideas that we collectively call physical optimization. They can be contrasted with other methods for optimization. Heuristics can be considered as an approach motivated by the problem; combinatorial optimization as one from mathematics, and expert systems as one from computer science.

There is no universally good approach to optimization. Each method has different trade offs in robustness, accuracy, speed, suitability for parallelization, and problem size dependence. For instance, neural networks do simple things on large data sets and parallelize easily while expert systems do complex things on small data sets and parallelize with difficulty. In nature, we see at least four approaches combining to solve the problem of survival. On the long term, a genetic algorithm is used to evolve people to maximize survival. On the short term, we wish to avoid being eaten by a lion. A relatively simple low-level vision network with largely local connections is used to process the initial image. A learning (learned) back propagation like network may be used to distinguish various animals in the scene. A high level, possibly expert system like, reasoning is used to optimize escape procedures after the lion has been identified.

Also note that physical analogies tend to be fundamentally imprecise; when applied to optimization, they find approximate and hopefully good solutions, but not the best. Combinatorial optimization aims for the exact solution. In practice, approximate solutions to large real world problems are all that is required and, indeed, all that is warranted by imprecise data.

In the next section, we describe a novel cellular automata approach to understanding society. The majority of the paper is devoted to optimization and Section 3 describes the basic ideas of physical optimization. Initially, we describe deterministic annealing for clustering in Section 4; and neural networks and simulated annealing for the travelling salesman problem, computer load balancing, and vehicle navigation in Section 5. In Section 6, we develop the elastic network and its relation to neural networks for the same applications. In Section 7, we show that track finding may be tackled by difference approaches in different regimes differing in track and noise density. The final Section 8 looks to the future.

2 Physical Analogies in Complex System

As mentioned in Section 1, thermodynamics and statistical physics has taught us how to understand large systems built out of many particles interacting with nature's laws. One may speculate that similar behavior may be exhibited by large complex systems composed of members linked in ways distinct from traditional physics. As described in [Fox:88a] (Chapter 3), [Fox:86a], [Fox:85a] and [Fox:88tt], we successfully applied these ideas to decomposing problems onto a multicomputer. We were able to introduce a concept analogous to temperature and observe phase transitions between different types of problem decompositions onto the computer. We are currently experimenting with such an analogy for society viewed as a complex system of interacting people. In particular, we are exploring the idea that the recent changes in Eastern Europe can be viewed as a phase transition. These are seen in physical systems possessing more than one stable state idealized in Figure 1 with two states m_1 and m_2 whose free energy takes a value $F_i(\lambda)$ as a function of parameter λ. For a magnetic material, λ could be the strength of the external magnetic field or it could refer to an internal coupling strength. For one value $\lambda = \lambda_a$, m_1 may be the equilibrium state, but as λ varies one may find $F_1(\lambda_b) > F_2(\lambda_b)$ with a transition λ_T in between λ_a and λ_b so that m_2 is the ground state at $\lambda = \lambda_b$. One also sees the effect of "supercooling" where the system is in the wrong (higher energy) state until a slight disturbance causes it to find its true equilibrium.

We chose to model Eastern Europe not as a bunch of people, but rather in terms of geographical cells as idealized in Figure 2 [Fox:90l]. In a serious simulation, one might cover (Eastern) Europe by a set of some 10^5 cells. Each cell holds a single spin s_i where at the crudest level;

$$s_i = +1 \quad \text{if cell is communist}$$

Figure 1: The value of the free energy for a typical system as a function of system configuration. The two curves correspond to two values of a parameter λ; $\lambda = \lambda_a$ is solid curve, $\lambda = \lambda_b$ is dashed. The index i of $F_i(\lambda)$ labels two equilibrium states m_i.

Figure 2: A Cellular Automata model for Eastern Europe.

$$s_i = -1 \qquad \text{if cell is capitalist} \tag{1}$$

We model society as governed by a Hamiltonian H that is a function of each s_i where i runs over all the cells. One can speculate on suitable terms to include in H. One possible term would be a nearest neighbor interaction:

$$H_1 = -J_1 \sum_{\langle ij \rangle} s_i \, s_j \tag{2}$$

where the interaction between i and j with a positive J_1 tends to force s_i and s_j to line up in order to minimize H_1, i.e., Eastern Europe is a ferromagnetic material. In an unsophisticated culture, only neighboring i and j are linked in Equation 2 but with an ever increasing fashion, the information revolution (phones, TV, networks) links i and j at geographically distant points. Another interesting term would be

$$H_5 = -J_5 \sum_i s_i . \tag{3}$$

This is an external field in the physics analogy with, for instance, the "Voice of America" broadcasts contributing a negative J_5 tending to force the s_i to negative (capitalist) values. The current model is completed with boundary conditions at Western Europe; a set of distactor spins d_α — "the Warsaw Pact" — which couple to each other and to the people s_i; impurity spins "Gorbachev", "Lech Walechsa"; and a temperature representing the size of internal fluctuations allowed by the government.

This model is qualitatively reasonable with different societies (U.S.A. vs present day China vs. Eastern Europe today vs. medieval feudal Europe) showing quite different parameter values.

A different approach to such systems is in terms of chaos and attractors from non-linear dynamical systems [Stein:89a]. The relation and relative merits of the two approaches is not clear. Both neural networks ($\sim$ cellular automata) and dynamical system models can provide good extrapolations of the time series produced by such complex systems.

3 Physical Optimization

Suppose we wish to minimize

$$E = E(\text{parameters } \underline{y}) \tag{4}$$

where the parameters $\underline{y}$ can be continuous, discrete or a mix. We introduce a fake temperature T and set $\beta = 1/T$; in particular cases, we will have a simple physical interpretation of T as, for instance, the scale or granularity at which the problem is formulated. A state of the system is labeled by $\underline{y}$ and to each state we associate a probability

$$\begin{aligned} Pr(\underline{y}) &= \frac{e^{-\beta E(\underline{y})}}{Z} \\ \text{where } Z &= \sum_{\underline{y}} Pr(\underline{y}) \end{aligned} \tag{5}$$

As $T \rightarrow 0$ or $\beta \rightarrow \infty$, the minimum $\underline{y}_{\text{min}}$ of E dominates in Z and the probability of this state tends to unity. The basic idea in annealing is to find the minimum $\underline{y}_{\text{min}}(T)$ of $F(T)$ for $T \sim 0$ by tracking $\underline{y}_{\text{min}}(T)$ from high to low temperatures. Here, $F = E - TS$ is the free energy expressed in terms of original energy E and entropy S. In some cases, Rose has been able to show that $\underline{y}_{\text{min}}(T)$ is continuous in T, and computer experiments indicate this is at least approimxately true in the cases we look at. Then, this continuation technique should avoid local minima, such as the one shown on Figure 3, as it is easy to find true global minima at high temperatures.

We will describe in the following, four different methods for finding $\underline{y}_{\text{min}}(T)$ and tracking it with temperature.

1. **Simulated Annealing:** We find $\underline{y}_{\text{sample}}(T)$ by Monte Carlo as either a mean or a representative of configurations at temperature T. As both the mean and a representative have the same limit as $T \rightarrow 0$, we can use either. This method is the best known and currently most reliable

Figure 3: Schematic of the free energy $F(T)$ at a set of temperatures $0 \sim T_1 < T_2 < T_3 < T_4 \sim \infty$.

physical optimization method [Kirkpatrick:83a]. We view it as the standard of excellence for the three alternative and less well-known methods that we will discuss in this paper. From the point of optimization theory, the other methods are faster as they avoid costly Monte Carlo steps but less reliable, i.e., they are less successful in avoiding local minima. All the methods have the disadvantage that they cannot in practice either guarantee achieving a true global minima or estimate the quality $(F(y_{\min}(\text{found as } T \rightarrow 0)) - F(y_{\min}(\text{true})))$ of solution. Experience has shown that several of these methods give very reliable answers, near to the true minima, for a variety of problems. We remember that these are methods designed to find approximate and not exact minima.

2. **Deterministic Annealing:** Here we just choose a simple heuristic to minimize $F(T)$ at temperature T [Rose:90f] where one starts with an initial guess for temperature T as the minimum $\underline{y}_{\min}(T+\delta T)$ at higher temperature. This is most effective for cases where $\underline{y}$ is low dimensional, i.e., we only have a few degrees of freedom. This can be achieved in some cases by summing over all except a few critical components of $\underline{y}$. We discuss this in Section 4. The deterministic approach is familiar in Chemistry where particle dynamics and Monte Carlo are both used to find the ground state of a complex molecule. The atoms in the molecule are often found experimentally — say from NMR measurements—and the annealing minimizes a potential containing both physical forces and artificial terms representing agreement of the model with the data. This can be viewed as physical computation addressing the optimization problem "what molecule best fits the experimental data and is consistent with Chemistry" [Brooks:83b]. A similar idea underlines the molecular dynamics approach to Quantum Chromodynamics Lattice Simulations [Gottlieb:87a], [Duane:86a], [Duane:87a].

3. **Neural Networks:** [Hopfield:85b], [Hopfield:86a] This could, and probably does, have deep biological significance but here we can "just" view this method as a calculation of an approximation to $\underline{y}_{\min}(T)$ using the mean field approximation in the case where $\underline{y}$ is discrete. We discuss this in Section 5.

4. **Elastic Network:** [Durbin:87a] We can view this as a similar approach to that of neural networks where an improved mean field strategy is used that incorporates some of the constraints which are used as penalty terms in the neural network method. We discuss this in Section 6.

We illustrate these general ideas with examples in the following sections.

4 Deterministic Annealing for Clustering

Consider a set of data points x which we wish to associate into clusters. This type of problem comes up in many applications, but here we will consider physical clusters in a two-dimensional space such as the 360 points shown in Figure 4 and generated by four clusters [Rose:90a], [Rose:90d].

For each data point x, we assign an energy $E_x(j)$ (cost) for it to belong to the cluster j with mean y_j. We sum over the uninteresting variables that specify the assignment of x to one of the N_c clusters. Then the partition function is

$$Z = \prod_x \sum_{k=1}^{N_c} \exp[-\beta E_x(k)] \tag{6}$$

$$\text{and the free energy } F = -\frac{1}{\beta} \log Z \tag{7}$$

If the clusters were due to Gaussian fluctuations then we can take

$$E_x(j) = |x - y_j|^2 \tag{8}$$

and now the cluster centers are determined by the deterministic annealing condition

$$\frac{\partial F}{\partial y_j} = 0 \tag{9}$$

which gives the implicit equation

$$y_j = \frac{\sum\limits_x x \, Pr(x \text{ in cluster } j)}{\sum Pr(x \text{ in cluster } j)} \tag{10}$$

where y_j also appears on the right hand side from the expression for the probability;

$$Pr(x \text{ in cluster } j) = \frac{\exp\left(-\beta|x - y_j|^2\right)}{\sum\limits_{k=1}^{N_c} \exp\left(-\beta|x - y_k|^2\right)} \tag{11}$$

We can solve the implicit equation iteratively starting with

$$y_j(T = \infty) = \text{mean of all } x\text{'s} \tag{12}$$

and gradually reducing the temperature T. At temperature T, we initialize the search for $y_j(T)$ by using $y_j(T + \delta T)$ as a starting point. Note that Equation (10) surely has many local minima but these are avoided by the annealing as we can usually find the global minimum easily at high temperature and track it down with lowering temperature. This particular example has a striking pattern in its temperature dependence. At high temperature, all the clusters are degenerate with the same y_j. As β is increased, one finds a critical temperature (β) at which

Figure 4: Deterministic Annealing Clustering of four clusters. The lines are the decision boundaries. The final beta is 0.1, and the final energy is 30.05. ◯—computed cluster mean. X—center of cluster used to generate points $\underline{x}$ [Rose:90a].

the clusters break into two sets—again the members of each set have identical y_j [Rose:90c]. This is shown in Figure 5 with transitions corresponding to 1–2–3–4 and finally 18 clusters. We can understand this because $T^{1/2}$ is proportional to the distance scale at which the system is observed. Indeed, one can view temperature as the Lagrange multiplier needed when entropy is maximized at fixed cluster variance, i.e., a fixed distance scale. Lowering T corresponds to looking at a finer and finer scale and so we need to specify a minimum interesting T. The 18 clusters found in Figure 5 correspond to looking inside the four real clusters. Figure 4 shows that very good results are obtained by this method.

In this application, annealing is equivalent to a multiscale approach; we initialize the fine scale optimization with the results of a coarse scale analysis. The utility of this approach is known in many fields including the well known multigrid method for particle differential equations. It also has been applied to vision by Terzopoulos [Terzopoulos:83a], [Terzopoulos:86a], [Battiti:91a].

5 Neural Networks for Optimization

These were first introduced for the Travelling Salesman Problem (TSP) by Hopfield and Tank [Hopfield:85b], [Hopfield:86a] and although the method is not very effective in this application [Wilson:88a], the basic ideas are important for a range of problems. We will set up the formalism for the TSP and then show how it can be applied to parallel program decomposition (Section 5.2) and navigation (Section 5.3). The application in Section 5.2 is very successful for reasons we will be able to identify, but for the examples in Section 5.1 and Section 5.3, neural networks do not perform well. However, in Section 6, we will give the elastic network extension which gives good results for the TSP and navigation. In the final Section 7, we show that track finding naturally uses either elastic and neural networks but in different domains of the parameter values.

For the point of view of deterministic annealing, elastic and neural networks are similar. They both use mean field approximations to the free energy and deterministic methods to solve the resultant equations.

5.1 The Traveling Salesman Problem

Consider a set of N cities labeled by the integer $p = 1 \ldots N$ and illustrated in Figure 6. We wish to visit each city once and once only in a tour that minimizes the total distance travelled. We let $i = 1 \ldots N$ label successive steps of the tour with $p = P(i)$ labelling the city visited at the i'th step. Then we need to minimize

$$\sum_{i=1}^{N-1} d_{P(i)\,P(i+1)} \tag{13}$$

Figure 5: The phase structure of Figure 4 as a function of temperature or $\beta = 1/T$ [Rose:90c].

Figure 6: Travelling Salesman Non-Optimal Path — A simple TSP indicating labels p(city) and i(time).

This assignment ($i \rightarrow P(i)$) is the classic NP complete discrete optimization problem [Papadimitriou:82a] which is often used as the standard benchmark for discrete optimization methods. The following examples make it clear that the TSP is not necessarily typical of all such problems and methods that perform poorly on the TSP work well on other (NP complete) optimization problems.

We introduce the neural variables

$$\begin{aligned} \eta_p^i &= 1 \text{ if } p = P(i) \\ &= 0 \text{ if } p \neq P(i) \end{aligned} \tag{14}$$

and we rewrite Equation (13) as

$$E_1 = \sum_i \sum_{p,\, q} d_{pq} \eta_p^i \eta_q^{i+1} \tag{15}$$

where in Equation (13) and Equation (15), d_{pq} is the distance between cities p and q. We now have a nice quadratic form to minimize as a function of the N^2 neural variables η_p^i. Unfortunately, not all choices of η_p^i are allowed; for this to correspond to a true assignment, one needs to satisfy constraints that each i corresponds to one p and vice versa. These can be written as

$$\begin{aligned} \sum_i \eta_p^i \eta_q^i &= \delta_{pq} \\ \sum_p \eta_p^i \eta_p^j &= 0 \qquad i \neq j \end{aligned} \tag{16}$$

These constraints are implemented by minimizing

$$E = E_1(\text{equation 15}) + \sum [\text{constraints — equation 16}] \tag{17a}$$

with penalty terms, simple linear or quadratic functions of the forms (Equation (16)), which are positive when constraints are violated.

A typical example for a penalty term could be,

$$\sum_{i \neq j} \left[\sum_p \eta_p^i \eta_p^j \right]^2 \tag{17b}$$

Combining Equation (17) with the physical optimization framework of Section 3 leads to a traditional statistical physics problem with N^2 "spins" η_p^i governed by an energy function E. The resultant simulated annealing or Monte Carlo approach to this statistical physics formulation does provide an effective approach to the TSP [Martin:89a].

Here we will study a faster, but less reliable, deterministic method. A well-known approximate method for studying such physics systems is the mean field approximation. Consider an equation such as

$$\langle \eta_r^k \rangle = \sum_{\text{states}} \eta_r^k \exp(-\beta E(\eta_1^1 \dots \eta_r^k \dots \eta_N^N))/Z \tag{18}$$

One can calculate this if one linearizes the exponential by approximating any term in E that is quadratic in η by a linear dynamic term multiplied by the "mean field" — the other η's replaced by their mean value. Roughly, one substitutes

$$\underline{\eta} M \underline{\eta} \rightarrow \langle \underline{\eta} \rangle M \underline{\eta} \tag{19}$$

With the approximation of Equation (19), one can sum over the dynamical variables labelling the states and Equation (18) can be converted into a deterministic equation for $\langle \eta_r^k \rangle$. At the desired ground state, $\eta_r^k = \langle \eta_r^k \rangle$ and one finds a deterministic method for finding the minimum of E.

Unfortunately, this method is an approximation and one will find "illegal" solutions which are not only non-optimal in E but also violate the constraint penalty terms in Equation (17). This has made this approach unsatisfactory for even modest ($N \sim 50$) TSP problems [Wilson:88a]. These results are disappointing for the TSP, but we include them as the basic ideas *are* correct and are very successful in related NP complete optimzation problems, as we describe later in this paper.

5.2 Load Balancing Parallel Programs

A similar approach is much more successful in load balancing; although this is also NP complete and formally equivalent to the TSP, there is a natural neural representation that involves no constraints and penalty terms. Here we get non-optimal solutions, but ones that are accurate enough for the problem at hand [Williams:91a]. This is encouraging as it shows that the difficulty with the TSP neural network is not with a deterministic annealing approach, but rather with the choice of variables. In Section 6, we will change these variables and find good TSP results for deterministic annealing.

The load balancing or (automatic) decomposition problem in parallel programming depends on many issues; the application, the software paradigm and the parallel computer architecture. We have discussed these points elsewhere [Fox:86h], [Fox:88f], [Fox:88mm], [Fox:88uu], [Fox:89q] and here we will consider loosely synchronous problems running on a hypercube; we indicate how to generalize to other architectures at the end of this subsection. We can abstract load balancing graph theoretically as illustrated in Figure 7. The application is defined by a graph with M members, labeled by $m = 1 \dots M$, such that $w(m)$ computational units are needed to "update" m while the matrix $C(m, m')$ is a measure of information needed to be transmitted from m' to m to update graph node m. We wish to decompose the graph onto $N = 2^d$ processor nodes so as to minimize total execution time. As in Section 5.1, we again have an association problem—this time of $m \rightarrow P(m)$ where P is processor number to which m is assigned. We could, as in Equation (14), introduce neural variables $\eta(m,p) = 1$ if $p = P(m)$ and 0 otherwise, but this has difficulties already seen for the TSP. Rather, we write

$$P(m) = \sum_{k=0}^{d-1} 2^k \eta_k(m) \tag{20}$$

and the $Md = M \log_2 N$ neural variables $\eta_k(m)$ provide a non-redundant specification of the decomposition. This is to be compared to the MN variables $\eta(m,p)$ in the redundant formulation. Using some technical assumptions, we can now specify the energy of the associated physical system as

$$\begin{aligned} E &= E_{\text{calc}} + E_{\text{comm}} \quad \text{where} \\ E_{\text{calc}} &= \frac{1}{N} \sum_{m,m'} w(m)\, w(m') \prod_{k=0}^{d-1} [1 + s_k(m) s_k(m')] \\ E_{\text{comm}} &= \frac{1}{4} \sum_{m,m'} C(m,m') \sum_{k=0}^{d-1} [1 - s_k(m) s_k(m')] \end{aligned} \tag{21}$$

with spins $s_k(m) = 2\eta_k(m) - 1$ taking values of ± 1 (for $\eta_k(m) = 1$ or 0). The physical analogy is particularly good here with E_{calc} (balancing computation on each node) as a short range repulsion and E_{comm} (minimizing communication) as a long range attractive force.

Equation (21) can now be used in the physical optimization approach; both the simulated annealing and neural network methods can be applied [Williams:91a]. Indeed, even in simulated annealing, the elegant neural network choice of variables is preferred to a direct expression of energy E in terms of $P(m)$. As shown in [Fox:88e], the mean field method developed by Hopfield and Tank for the TSP is directly applicable and gives excellent results. These are comparable in quality to simulated annealing but much faster as one is just solving deterministic equations [Williams:91a]. As already mentioned, neural networks work well here because the objective function E in Equation (21) has no penalty terms.

The neural representation of Equation (20) was originally motivated by the hypercube topology. However, it is generally useful and there is an interesting analogy with the clustering method in Section 4. Thus, as shown in Figure 8, the neurons provide a multiscale representation with $\eta_{d-1}(m)$ the coarsest and $\eta_0(m)$ the finest detail. From this point of view, it is clear how to generalize this approach to non-hypercube topologies by using the appropriate hierarchical neural multiscale representation in "processor space".

5.3 Navigation

The discussion of the last subsection is confined to essentially static or what we like to call adiabatic problems [Fox:89q]. More challenging are dynamic

Figure 7: Decomposition as a Graph Theory Problem.

Figure 8: Multiscale Decomposition of Processor Node Space with Neural Variables. The case $d = 3$ is illustrated.

problems where one needs to determine for each member, m, the time dependent processor location $P(m, t)$. We developed a path or string formalism for this generalized problem [Fox:88f] and realized that the methods were similar to those needed for navigation problems. These include two- and three-dimensional land vehicles and aircraft path planning, and the motion of one or more multi-joint robot arms. In each case, one needs to determine the path of one or more entities in a target space; this path typically involving minimizing a travel time subject to constraints involving avoidance of obstacles and collisions between entities. In the parallel processing application, the target space is formed by the nodes of the computer with a topology defined by the architecture. In the navigation case, the target space is typically a two- or three-dimensional physical space although the higher dimensional configuration space may be used in a robotics example [Barraguard:90a].

One can formulate a physical computation approach to navigation using the direct neural network of Section 5.1, the multiscale representation of Section 5.2 or the path or elastic net formalism of Section 6. We believe the latter to be most promising, but we have only completed the analysis using the simple neural representation introduced in Section 5.1 [Fox:88kk], [Fox:89aa], [Gurewitz:89b], [Gurewitz:90a]. We will briefly discuss this here as it illustrates the main issues for the more powerful approach based on the methods of Section 6 [Gandhi:90b].

Consider a vehicle, V, avoiding a missile, M. Introduce two sets of neural variables $v(\underline{x}, t), m(\underline{x}, t)$

$$\begin{aligned} v(\underline{x}, t) &= 1 \text{ if vehicle at point } \underline{x} \text{ at time } t \\ &= 0 \text{ if vehicle not at } \underline{x}. \end{aligned} \tag{22}$$

$$m(\underline{x}, t) = 1 \text{ if missile at point } \underline{x} \text{ at time } t$$
$$= 0 \text{ if missile not at point } \underline{x}. \tag{23}$$

The motion of the vehicle can be found by minimizing an energy function E embodying its goals and constraints. This includes a term

$$E_1 = \sum_{\underline{x}, t} v(\underline{x}, t) T(\underline{x}) \tag{24}$$

where the terrain function T measures the difficulty of driving vehicle at point $\underline{x}$.

$$E_2 = \sum_{\underline{x}, t} v(\underline{x}, t)[m(\underline{x}, t-1) + m(\underline{x}, t) + m(\underline{x}, t+1)] \tag{25}$$

Minimizing E_2 ensures that vehicle and missile are safely separated!

$$E_3 = \sum_{\underline{x}, t} v(\underline{x}, t) |\underline{x} - \underline{x}_{\text{destination}}| \tag{26}$$

This term "attracts" vehicle to its destination $\underline{x}_{\text{destination}}$. These terms are not realistic but indicative of the ease with which one can express goals and constraints in this formalism. Our publications give more details and we give, in Figure 9, one example where we determine the path of four vehicles. These must avoid each other and reach their destinations by passage through a narrow "mountain pass".

We understand now and show in Section 7 that this neural approach is best for the case of very many vehicles. The traditional combinatorial method is fine for one and perhaps two vehicles; the very interesting case of "several" vehicles is probably best handled by the elastic net method. The redundancy of the neural net $v(\underline{x}, t) = 0$ at all $\underline{x}$'s not occupied by vehicles—is, just as in the TSP, a major difficulty when the number of vehicles is small.

We stress that the essential point of our approaches is that they scale naturally to problems with many vehicles [Jones:80a] or robots with many arms whilst the traditional methods have a time complexity of $O(N^l)$ for l degrees of freedom when each of these is discretized into N cells [Heinzinger:90a].

6 The Elastic Net

6.1 The Travelling Salesman Problem

Durbin and Wilshaw introduced the elastic net approach to the TSP [Durbin:87a], [Durbin:89a] as a physically based model that outperformed the neural network method in this case. One "invents" a physical system whose equilibrium state is the desired minimum path [Rose:90f]. As shown in Figure 10, we consider an elastic string with beads, labeled i, for each (time) step of the journey. The

Figure 9: The path produced by the neural network method for four vehicles. The "cost" (vertical axis) is a measure of travel time and travel in the two-dimensional $x - y$ plane must avoid other vehicles and the shaded obstacles. Circles represent initial positions and the final destinations [Fox:90e].

beads are attracted to each other by a simple elastic force that, in the absence of other constraints, collapses the string to zero length. We start with this at temperature $T = \infty$ when the elastic forces collapse the beads to a point. Each bead i is attracted to each city p by a force

$$F_i^{city,p} = \alpha w_{pi}(\underline{x}_p - \underline{y}_i) \tag{27}$$

where

$$w_{pi} = \frac{\exp -|\underline{x}_p - \underline{y}_i|^2/2K^2}{\sum_j \exp -|\underline{x}_p - \underline{y}_j|^2/2K^2} \tag{28}$$

where city p is at position $\underline{x}_p$ and bead i is at position $\underline{y}_i$.

The deterministic annealing equation is

$$\begin{aligned} \underline{y}_i &\rightarrow \underline{y}_i + \Delta \underline{y}_i \\ \Delta \underline{y}_i &= \sum_p F_i^{city,p} + \beta K(\underline{y}_{j+1} + \underline{y}_{j-1} - 2\underline{y}_j) \end{aligned} \tag{29}$$

including the elastic term together with the force of Equation (27). This corresponds to

$$\Delta y_i = -K \frac{\partial E}{\partial \underline{y}_i} \tag{30}$$

with the energy function E given by

$$\begin{aligned} E &= -\alpha K \textstyle\sum_p \log \sum_i \exp(-|\underline{x}_p - \underline{y}_i|^2/2K^2) \\ &\qquad + \beta \textstyle\sum_j |\underline{y}_j - \underline{y}_{j+1}|^2 \end{aligned} \tag{31}$$

The formulation is now similar to that of Section 4 with K playing the role of temperature or equivalently in this case, as for clustering, a position resolution. For large K, Equation (31) is minimized with all bead's i at the geometric average (center of mass) of the cities

$$\underline{y}_i(K = \infty) = \frac{1}{N_{city}} \sum_p \underline{x}_p \tag{32}$$

We have exactly satisfied the second term in Equation (31) corresponding to the elastic forces between beads and we are surely at a global minimum of E. As K is reduced, we increase the importance of the first term in E which is the constraint that each bead lie near a city (or in the limit $K = T \rightarrow 0$, each bead is on top of). In [Durbin:89a], it is shown that one gets a similar set of bifurcations to that illustrated in Figure 5 as K is lowered and the single global minima splits into several local minima. Empirically, the deterministic annealing ansatz of Equation (29), namely:

- Start at $K_0 = T = \infty$,

Figure 10: The elastic net for the TSP of Figure 6.

- Minimize Energy E at fixed $K = K_0$ starting with minimum for $K = K_0 + \delta K_0$ and using steepest descent, Equation (30),
- reduce K from K_0 to $K_0 - \delta K_0$

tracks the global minimum approximately and gives good solutions to the Travelling Salesman Problem.

6.2 Neural Network Approach

In Section 6.1, we showed that the TSP could be mapped into a physical computation involving a mechanical system of interacting beads. In Section 5.1, a seemingly different physical analogy was presented with the solution of the TSP corresponding to the ground state of a system of interacting spins η_p^i. Simic has shown that these two approaches correspond to different approximation solution methods for finding the ground state of essentially the same physical system. Here, we will just describe the relation between the approximation methods without the detailed mathematical justification which can be found in [Simic:90a].

Simic starts with the analogy of Section 5.1 with the energy function of Equation (17) which includes penalty terms corresponding to the constraints of Equation (16). The degrees of freedom are the redundant set $\{\eta_p^i\}$. Applying the mean field approximation and deterministic annealing to the statistical mechanics based on Equation (17), gives rise to the neural network approach. Alternatively, we may choose a "better" set of degrees of freedom which satisfy exactly some of the constraints of Equation (16). Suppose we only consider those physical states $\{\eta_p^i\}$ which exactly satisfy

$$\begin{aligned} \sum_p \eta_p^i \, \eta_p^j &= 0 \qquad i \neq j \\ \sum_i \eta_p^i \, \eta_p^i &= 1, \quad \text{all } p \end{aligned} \tag{33}$$

We are ensuring that each city p corresponds to one and only one time step i but we allow the constraint

$$\sum_i \eta_p^i \, \eta_q^i = 0 \qquad p \neq q \tag{34}$$

to be violated, i.e., we do not enforce that each time step i be associated with a unique city p. Equation (34) is included as a penalty term in the energy function.

Then Simic derives a new mean field approximation for

$$E = E_1 \text{ (Equation (15)) } + \sum \text{ [constraints – Equation (34)]} \tag{35}$$

with this restricted phase space. Change variables to

$$\underline{y}_i = \sum_p \underline{x}_p \langle \eta_p^i \rangle \tag{36}$$

where $\langle \eta_p^i \rangle$ are the mean values of the fields η_p^i in the mean field approximation to Equation (35). Then apart from a few technical differences, we find that we have derived the elastic net method expressed in terms of the bead positions of Equation (36). The term E_1 gives rise to the elastic forces between beads, and the constraint of Equation (34) leads naturally to the potential which gives the forces of Equation (27) which attract cities p to beads i. This is reasonable as the constraint of Equation (34) is ensuring that each bead i correspond to a single city p.

6.3 Other Applications of the Elastic Net

An important consequence of Simic's results is that we understood better the trade off between the neural network and elastic network approaches; both correspond to deterministic annealing of an energy function, but with different phase spaces, i.e., different degrees of freedom. Secondly, we can more easily generalize the elastic network from the TSP to other optimization problems [Rose:90f]. We now briefly show how the problems of Section 5.2 and Section 5.3 fit into this scheme.

As shown in Section 5.2, the neural network approach was very successful in the load balancing application. We do not need to, and indeed cannot, "improve" it using the ideas of Section 6.2 because there are no constraints in the energy function of Equation (21). The neural variables of Equation (20) are not redundant like those used for the TSP.

On the other hand, the neural variables of Equation (22) and Equation (23) used in the navigation problem are redundant and constraints must be satisfied. Thus, we would expect that the direct (Hopfield-Tank) neural network approach can be improved using the ideas of Section 6.2 This is in fact straightforward as shown in [Fox:90k], [Gandhi:90a], and [Gandhi:90b]. In fact, the situation is easier than the TSP because we do not need to ensure the two sets of constraints — corresponding to Equation (32) and Equation (33). Thus, for navigation, each time step t corresponds to a unique position $\underline{y}_{\mathrm{vehicle}}(t)$ but it is not necessary or generally desirable to ensure that each position be visited once and once only. Thus, one can use Simic's idea directly with the minor technical change that the variables i and p of Section 6.1 need to be interchanged. The resultant physical computation picture for navigation is straightforward and illustrated in Figure 11 for two vehicles with a common source and destination. The degrees of freedom are the positions of the vehicles at discretized time values labelled by an index $i = 1, 2 \ldots$. The goal of minimum travel time translates into elastic forces between neighboring beads on each path. Other constraints correspond to avoidance of obstacles — this is represented as repulsive forces between obstacles and beads; and avoidance of collisions corresponding to repulsive forces between beads of different vehicles at each time step. The formalism allows the inclusion of many different issues such as variable terrain and vehicles of finite size and these are discussed in [Gandhi:90b]. This work is still at an exploratory stage, but we believe it is very promising and can be applied to areas such as robot arm manipulation. There, a similar physical approach has been extensively explored [Barraguard:90a], but our elastic network is significantly different as previous approaches have viewed the problem in terms of the "Newtonian" dynamics of the instantaneous position of the "vehicle" whereas we use the complete path as the basic degrees of freedom. The advantage of the elastic net approach is that you can ensure that the paths link source and destination whereas the other methods can get trapped with the "vehicle" (robot arm) evolved in time into a state which cannot (easily) reach the desired destination.

Note that there are two distinct features of physical computation and, in particular, of the elastic net approach to the TSP and other problems. Critical is the choice of degrees of freedom and the idea of the minimization of an energy function E. Secondly, we can use either deterministic or Monte Carlo, or probably better, a mix to find the approximate minimum of E.

We can also relate the elastic network to the deterministic clustering algorithm of Section 4. Indeed, the TSP can be viewed as a special case of clustering where there is but a single point in each cluster. In his thesis, [Rose:90f] and [Rose:90e], Rose has shown how the elastic network approach can be derived from the formalism given in Section 4. Thus, we find a rather unified picture with different formulations of physical computation being related in a clear fashion.

7 Track Finding

7.1 The Problem

Here, we briefly discuss track finding to illustrate the different possible approaches to optimization that we have discussed here. Consider a set of measurements $\underline{x}_k\,(t_l) \pm \delta\underline{x}_k\,(t_l)$ where we observe the data values $\underline{x}_k$ at a set of times t_l. This application is present in many different areas of data analysis and signal processing. These applications differ in the number and reliability of the measurements and the number of underlying tracks giving use to these signals. The example in Figure 12(a) shows five tracks in a very noisy environment with many false signals. Other examples differ in the number of tracks — which could vary from one to say 10^5 in a strategic defense application; the complexity (cross overs) of tracks; the noise level; and the number of dimensions — the measurements can be in one, two or three spatial dimensions. The applications also differ in the shape of the tracks; they can be at their simplest, linear as in

Figure 11: The Physical Computation picture for navigation in the generalized elastic net approach [Gandhi:90b].

Figure 12 or curved as in say high energy physics data analysis, where several hundred particles will bend in a magnetic field as they are produced in a collision in the new SSC (superconducting colliding beams under construction near Dallas).

In each case, one wishes to find an optimal interpretation of the situation given the data measurements $\underline{x}_k\,(t_l)$, their errors, and any knowledge or prejudice as to the nature and number of tracks. So we have what is "just" an (NP-complete) optimization problem which can be approached by the many methods, including those discussed here. In the following sections, we discuss appropriate methods in different parameter domains illustrated schematically in Figure 13.

7.2 The Kalman Filter Approach

When one is finding a relatively small number of tracks in a clean environment, the Kalman filter approach is well developed and optimal [Blackman:86a]. Very crudely, we can explain this as follows. First, one finds all $\mathbf{N}_{Tr}$ tracks up to time t_0 and then extends them to $t_0 + \delta t_0$ by using $\mathbf{N}_M$ measurements at this later time value. The optimization involves a chi squared fit following the solution of an assignment problem of the following kind:

- What is the best assignment of the N_M measurements to lie on the extrapolation of the N_{Tr} tracks. (37)

This is solved by combinational or simple heuristic methods. In many cases, it is easy to find good assignments, but in general one is faced by a well-known optimal assignment problem which can be tackled by clever combinational algorithms such as Munkres' method [Blackman:86a], [Burgeios:71a], [Kuhn:55a] with a time complexity of order $O(N_{Tr}^2\, N_M)$. Although not NP complete (i.e., not exponential in $N_{Tr},\ N_M$), this method is very time consuming in many important cases even though we have been able to parallelize it effectively [Gottschalk:88a], [Gottschalk:90a], [Gottschalk:90b]. The use of neural networks for this case has been discussed in [Sergupta:88a], but this does not address the real difficulty. In cases where N_{Tr} and N_M are large, then the problem in Equation (37) becomes ill posed. There are too many ambiguities in the assignment for one to consider a single "best" solution which matches tracks to measurements. One can and does carry along several possibilities which are resolved by later measurements. However, in general, the strategy of reducing track finding to a single time step matching fails with noisy data or many tracks. Rather, one must go back to the "original full optimization problem" and view it as a simultaneous optimization over all (many) time values t_l, tracks and measurements $\underline{x}_k\,(t_l)$. In the following two sections, we consider this last possibility from two points of view.

Figure 12: (a) The original trajectories plus clutter. (b) The computed trajectories using deterministic annealing [Rose:89b], [Rose:90b], [Rose:90f].

Figure 13: Track Finding divided in three regions based on the noise and track complexity level and number of real tracks. The regions are labelled by the sections of this paper in which they are discussed.

7.3 An Analogy with Vision

One can consider the global optimization approach combinatorially, and, indeed, this is common in high energy physics data analysis. Indeed, I used it myself in analyzing data from an experiment at Fermilab which had around ten tracks to be found from about one hundred measurements [Fox:80a]. In many signal processing applications, it is natural to step in time as this is how the data is gathered. In High Energy Physics, the relativistic particles form tracks in a few nanoseconds, and the data is naturally gathered together for all time values.

However, the combinatorial approach is clearly limited and breaks down when N_{Tr} and N_M are large. In [Fox:89h], we noted that track finding is formally equivalent to edge detection in vision problems. The problem of finding tracks on one space and one time dimension, as illustrated in Figure 12, is formally identical to conventional two-dimensional vision. Track finding in two or three spatial dimensions, requires an obvious generalization of vision to three or four dimensions.

We know that combinatorial methods are not effective in vision, but rather one uses methods like neural networks and Hough transforms [Ballard:82a], [Ballard:86a], [Illingworth:88a]. We have neural variables $\eta(\underline{x}, t)$ which are nonzero when one or more tracks pass through the discretized point $(\underline{x}, t)$. Note that the number of neural variables is independent of the values of N_{Tr} and N_M. Thus, as N_{Tr}, N_M increase, the time complexity of a neural network formalism is invariant, whereas that of heuristic and combinatorial increases rapidly, sometimes exponentially. The major problem with the neural network method for the TSP in Section 5.1 was the redundant formalism and corresponding constraints. In this application, we have some constraints corresponding to continuous tracks, but these are easy to implement as a local term in the energy function. We do **not** have the difficult (in Section 5.1) constraints corresponding to uniqueness. Indeed, one usually does not know how many tracks are present and it is an advantage and not a disadvantage of the neural network formalism that it allows any number of tracks.

There has not been much experience with this approach, although initial results are encouraging [Kuczewski:88a]. It is clear to me that neural networks are the right approach to the case of many tracks in a noisy environment. However, further research and experimentation is needed.

7.4 The Elastic Net Approach

It is also clear to me that for some values of the parameters N_{Tr} and N_M, the elastic net or string (path) approach of Section 6 is appropriate. As shown in Figure 13, this will probably be used in the parameter region that is intermediate between the methods of Section 7.2 and Section 7.3.

At a superficial level, tracking and navigation are the "same" problem.

- In navigation, we are finding paths in some space respecting the terrain and vehicle behavior, and avoiding obstacles.
- In tracking, we are finding paths in some space respecting the track models and passing through measurements.

Thus, we find that the methods of Section 6.3 are immediately applicable to tracking. The repulsive forces in Figure 12 between obstacles and vehicles are replaced by attractive forces between measurements and tracks. Rose has explored this general approach in [Rose:89b], [Rose:90b] and obtained excellent results shown already in Figure 12. These ideas are developed more generally in [Fox:89h] where we note that the elastic net (string) approach is not really an alternative to the neural network method but rather a natural higher level formalism. After preliminary tracks are found from a Hough transform or neural network, this step needs to be followed up by a "clean-up" stage which labels independent tracks and makes them continuous; the step from the field of track densities $\eta(\underline{x}, t)$ which is nonzero at a candidate track, to the discrete set of N_{Tr} tracks $\{\underline{x}_i(t)\}$ is non-trivial. The elastic net is an appropriate approach to this higher level labelling problem.

7.5 Summary

We have illustrated by qualitative arguments in one example that there is no uniquely good approach to optimization. A given problem, namely tracking, is best approached by combinatorial, heuristic, neural network or elastic network methods in different circumstances. The last two methods, elastic and neural networks, are characterized by searching for approximate global solutions over the entire data set.

8 Conclusions

We have shown in this paper how physical analogies can be used in several applications outside the domain of the traditional physics problem. We believe that as we search for the solution of larger and larger problems on more and more powerful parallel computers, these analogies will grow in importance.

Acknowledgements
I would like to thank Paul Messina and his colleagues, especially Almerico Murli from the University of Naples, for their warm hospitality and an excellent conference. This work was supported by the Department of Energy: Applied Mathematical Sciences—Grant: DE-FG03-85ER25009; Joint Tactical Fusion Program Office; and the National Science Foundation under Cooperative Agreement No. CCR-8809165 — the Government has certain rights in this material.

The collaborations with Wojtek Furmanski, Eitan Gurewitz, Ken Rose, and Petar Simic were essential in crystallizing my ideas in this area. I thank TRW for very helpful discussions about robot manipulation.

References

[Ballard:82a] Ballard, D. H., and Brown, C. M. *Computer Vision.* Prentice-Hall, Englewood Cliffs, New Jersey 07632, 1982.

[Ballard:86a] Ballard, D. H. "Cortical connections and parallel processing: Structure and function," *The Behavioral and Brain Sciences*, 9:67–120, 1986.

[Barraguard:90a] Barraguard, J., and Latombe, J. C. "A monte carlo algorithm for path planning with many degrees of freedom," in *Proceedings of 1990 IEEE Robotics and Automation Conference*, May 1990. Held in Cincinnati, May 13–18, 1990.

[Battiti:91a] Battiti, R. "Real-time multiscale vision on multicomputers," *Concurrency: Practice and Experience*, 3(2):55–87, 1991. Accepted for Publication. Caltech Report C3P-932b.

[Blackman:86a] Blackman, S. S. *Multiple-Target Tracking with Radar Applications.* Artech House, Dedham, MA, 1986.

[Brooks:83b] Brooks, B. R., Bruccoleri, R. E., Olafson, B. D., States, D. J., Swaminathan, S., and Karplus, M. "CHARMM: a program for macromolecular energy, minimization, and dynamics calculations," *J. Comput. Chem*, 4(2):187–217, 1983.

[Burgeios:71a] Burgeios, F., and Lassalle, J. C. "An extension of munkres algorithm for the assignment problem to rectangular matrices," *Comm. of the ACM*, 14:802, 1971.

[Duane:86a] Duane, S., and Kogut, J. "The theory of hybrid stochastic algorithms," *Nucl. Phys.*, 275:398, 1986.

[Duane:87a] Duane, S., Kennedy, A. D., Pendleton, B. J., and Roweth, D. "Hybrid Monte Carlo," *Physics Letters B*, 195:216–220, 1987.

[Durbin:87a] Durbin, R., and Wilshaw, D. "An analogue approach to the traveling salesman problem using an elastic net method," *Nature*, 326:689–691, 1987.

[Durbin:89a] Durbin, A., Szeliski, R., and Yuille, A. "An analysis of the elastic net approach to the Travelling Salesman Problem," *Neural Computation*, 1:348–358, 1989.

[Fox:80a] Fox, G. C. "Jet production in high-energy Hadron-Proton collisions," *Nuclear PHysics B*, 171:38, 1980. (with experimental groups at Caltech, UCLA, Chicago Circle, Fermilab and Indiana).

[Fox:85a] Fox, G., Otto, S. W., and Umland, E. A. "Monte Carlo physics on a concurrent processor," *Journal of Statistical Physics*, 43(5/6), June 1986. Proceedings of the Conference on Frontiers of Quantum Monte Carlo, September 3–6, 1985 at Los Alamos. Caltech Report C3P-214.

[Fox:86a] Fox, G., and Otto, S. "Concurrent computation and the theory of complex systems," in M. T. Heath, editor, *Hypercube Multiprocessors*, pages 244–268. SIAM, Philadelphia, 1986. Caltech Report C3P-255.

[Fox:86h] Fox, G., Kolawa, A., and Williams, R. "The implementation of a dynamic load balancer," in M. T. Heath, editor, *Hypercube Multiprocessors*, pages 114–121. SIAM, Philadelphia, 1987. Caltech Report C3P-328.

[Fox:88a] Fox, G. C., Johnson, M. A., Lyzenga, G. A., Otto, S. W., Salmon, J. K., and Walker, D. W. *Solving Problems on Concurrent Processors*, volume 1. Prentice-Hall, Inc., Englewood Cliffs, NJ 07632, 1988.

[Fox:88e] Fox, G. C., and Furmanski, W. "Load balancing loosely synchronous problems with a neural network," in G. C. Fox, editor, *The Third Conference on Hypercube Concurrent Computers and Applications, Volume 1*, pages 241–278. ACM Press, 11 West 42nd Street, New York, NY 10036, January 1988. Caltech Report C3P-363b.

[Fox:88f] Fox, G. C., and Furmanski, W. "A string theory for time dependent complex systems and its application to automatic decomposition," in G. C. Fox, editor, *The Third Conference on Hypercube Concurrent Computers and Applications, Volume 1*, pages 285–305. ACM Press, 11 West 42nd Street, New York, NY 10036, January 1988. Caltech Report C3P-521.

[Fox:88kk] Fox, G. C., Furmanski, W., Ho, A., Koller, J., Simic, P., and Wong, Y. F. "Neural networks and dynamic complex systems." Technical Report C3P-695, California Institute of Technology, December 1988. Proceedings of 1989 SCS Eastern Conference , Tampa, Florida, March 28–31, 1989.

[Fox:88mm] Fox, G. C. "A review of automatic load balancing and decomposition methods for the hypercube," in M. Schultz, editor, *Numerical Algorithms for Modern Parallel Computer Architectures*, pages 63–76. Springer-Verlag, 1988. Caltech Report C3P-385.

[Fox:88tt] Fox, G. C., and Furmanski, W. "The physical structure of concurrent problems and concurrent computers," *Phil. Trans. R. Soc. Lond. A*, 326:411–444, 1988. Caltech Report C3P-493.

[Fox:88uu] Fox, G. C., and Furmanski, W. "The physical structure of concurrent problems and concurrent computers," in R. J. Elliott and C. A. R. Hoare, editors, *Scientific Applications of Multiprocessors*, pages 55–88. Prentice Hall, 1988. Caltech Report C3P-493.

[Fox:89h] Fox, G. C. "A note on neural networking for trackfinding." Technical Report C3P-748, California Institute of Technology, March 1989. Unpublished.

[Fox:89q] Fox, G. C., Furmanski, W., and Koller, J. "The use of neural networks in parallel software systems," *Mathematics and Computers in Simulation*, 31(6):485–495, 1989. Elsevier Science Publishers B. V. (North-Holland). Caltech Report C3P-642b.

[Fox:89aa] Fox, G. C., Gurewitz, E., and Wong, Y. "A neural network approach to multi-vehicle navigation." Technical Report C3P-833, California Institute of Tehcnology, October 1989. Published in Proceedings of SPIE Conference, Philadelphia.

[Fox:90e] Fox, G. C., Gurewitz, E., and Wong, Y. "A neural network approach to multi-vehicle navigation," in D. W. Walker and Q. F. Stout, editors, *The Fifth Distributed Memory Computing Conference, Volume I*, pages 148–152, 10662 Los Vaqueros Circle, P. O. Box 3014, Los Alamitos, California 90720-1264, 1990. IEEE Computer Society Press. Held April 9–12, Charleston, South Carolina. Caltech Report C3P-910.

[Fox:90k] Fox, G. C. "Applications of the generalized elastic net to navigation." Technical Report C3P-930, California Institute of Technology, June 1990. Unpublished.

[Fox:90l] Fox, G. C. "A complex system model for Eastern Europe." Technical Report C3P-931, California Institute of Technology, June 1990. Unpublished.

[Fox:91b] Fox, G. C. "Approaches to physical optimization." Technical Report C3P-959, California Institute of Technology, April 1991. CRPC-TR91124, SCCS-92, Submitted to SIAM J. Sci. Stat. Comp. for publication.

[Gandhi:90a] Gandhi, A., and Fox, G. C. "Solving problems in navigation." Technical Report SCCS-9, Syracuse University, 1990. Unpublished.

[Gandhi:90b] Gandhi, A., and Fox, G. C. "Physical optimization for navigation and robot manipulation." Technical Report SCCS-43, Syracuse University, 1990. Unpublished.

[Gottlieb:87a] Gottlieb, S. A., Liu, W., Toussaint, D., Renken, R. L., and Sugar, R. L. "Hybrid-molecular-dynamics algorithms for the numerical simulation of quantum chromodynamics," *Phys. Rev. D*, 35:2531, 1987.

[Gottschalk:88a] Gottschalk, T. D. "Concurrent multiple target tracking," in G. C. Fox, editor, *The Third Conference on Hypercube Concurrent Computers and Applications, Volume 2*, pages 1247–1268. ACM Press, 11 West 42nd Street, New York, NY 10036, January 1988. Caltech Report C3P-567.

[Gottschalk:90a] Gottschalk, T. D. "Concurrent implementation of Munkres algorithm," in D. W. Walker and Q. F. Stout, editors, *The Fifth Distributed Memory Computing Conference, Volume I*, pages 52–57, 10662 Los Vaqueros Circle, P. O. Box 3014, Los Alamitos, California 90720-1264, 1990. IEEE Computer Society Press. Held April 9–12, Charleston, South Carolina. Caltech Report C3P-899.

[Gottschalk:90b] Gottschalk, T. D. "Concurrent multi-target tracking," in D. W. Walker and Q. F. Stout, editors, *The Fifth Distributed Memory Computing Conference, Volume I*, pages 85–88, 10662 Los Vaqueros Circle, P. O. Box 3014, Los Alamitos, California 90720-1264, 1990. IEEE Computer Society Press. Held April 9–12, Charleston, South Carolina. Caltech Report C3P-908.

[Gurewitz:89b] Gurewitz, E., Fox, G. C., and Wong, Y.-F. "Parallel algorithm for one and two-vehicle navigation." Technical Report C3P-852, California Institute of Technology, 1989. Published in the Proceedings of the 1990 IEEE International Workshop on Intelligent Robots and Systems, IROS'90 held in Tsuchiura, Ibaraki, Japan, July 1990.

[Gurewitz:90a] Gurewitz, E., Fox, G. C., and Wong, Y.-F. "Parallel algorithm for one and two-vehicle navigation," in D. W. Walker and Q. F. Stout, editors, *The Fifth Distributed Memory Computing Conference, Volume I*, pages 140–147, 10662 Los Vaqueros Circle, P. O. Box 3014, Los Alamitos, California 90720-1264, 1990. IEEE Computer Society Press. Held April 9–12, Charleston, South Carolina. Caltech Report C3P-876.

[Heinzinger:90a] Heinzinger, G., Jacobs, P., Canny, J., and Paden, B. "Time-optimal trajectories for a robot manipulator: A provably good approximation algorithm," in *Proceedings of 1990 IEEE Robotics and Automation Conference*, May 1990. Held in Cincinnati, May 13–18, 1990.

[Hopfield:85b] Hopfield, J., and Tank, D. ""Neural" computation of decisions in optimization problems," *Biol. Cybern.*, 52:141–152, 1985.

[Hopfield:86a] Hopfield, J. J., and Tank, D. W. "Computing with neural circuits: a model," *Science*, 233:625, 1986.

[Illingworth:88a] Illingworth, J., and Kittler, J. "A survey of the Hough Transform," *Computer Vision, Graphics and Image Processing*, 44:87–116, 1988.

[Jones:80a] Jones, S. T. "Solving problems involving variable terrain, Part I: a general algorithm," *BYTE*, February 1980.

[Kirkpatrick:83a] Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. "Optimization by simulated annealing," *Science*, 220:671–680, May 1983.

[Kuczewski:88a] Kuczewski, R. M. *Neural Network Approaches to Multi-Target Tracking.* TRW MEAD, 1988.

[Kuhn:55a] Kuhn, H. W. "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, 2:83, 1955.

[Martin:89a] Martin, O., Otto, S. W., and Felten, E. W. "Large-step Markov chains for the traveling salesman problem." Technical Report C3P-836, California Institute of Technology, August 1989. Submitted to Operations Research.

[Papadimitriou:82a] Papadimitriou, C. H., and Steiglitz, K. *Combinatorial Optimization: Algorithms and Complexity.* Prentice-Hall, Englewood Cliffs, NJ, 1982.

[Rose:89b] Rose, K., Gurewitz, E., and Fox, Geoffrey, C. "A nonconvex cost optimization approach to tracking multiple targets by a parallel computational network." Technical Report C3P-853, California Institute of Technology, 1989. Published in Proceedings of the 1990 IEEE International Workshop on Intelligent Robots and Systems, IROS'90 held in Tsuchiura, Ibaraki, Japan, July 1990.

[Rose:90a] Rose, K., Gurewitz, E., and Fox, G. "A deterministic annealing approach to clustering," *Pattern Recognition Letters*, 11(9):589–594, September 1990. Caltech Report C3P-857.

[Rose:90b] Rose, K., Gurewitz, E., and Fox, Geoffrey, C. "A nonconvex cost optimization approach to tracking multiple targets by a parallel computational network," in D. W. Walker and Q. F. Stout, editors, *The Fifth Distributed Memory Computing Conference, Volume I*, pages 78–84, 10662 Los Vaqueros Circle, P. O. Box 3014, Los Alamitos, California 90720-1264, 1990. IEEE Computer Society Press. Held April 9–12, Charleston, South Carolina. Caltech Report C3P-875.

[Rose:90c] Rose, K., Gurewitz, E., and Fox, G. C. "Statistical mechanics and phase transitions in clustering," *Physical Review Letters*, 65(8):945–948, August 1990. Caltech Report C3P-893.

[Rose:90d] Rose, K., Gurewitz, E., and Fox, G. C. "Vector quantization by deterministic annealing." Technical Report C3P-895, California Institute of Technology, May 1990. Submitted to IEEE Transaction on Information Theory.

[Rose:90e] Rose, K., Gurewitz, E., and Fox, G. C. "Constrained clustering as an optimization method." Technical Report C3P-919, California Institute of Technology, June 1990. submitted to the IEEE Transactions on Pattern Analysis and Machine Intelligence.

[Rose:90f] Rose, K. *Deterministic Annealing, Clustering, and Optimization.* PhD thesis, California Institute of Technology, December 1990. Caltech Report C3P-950.

[Sergupta:88a] Sergupta, D., and Iltis, R. A. "A neural-network solution to the data association problem in multiple target tracking," *IEEE Trans. Aerospace and Electronic Systems*, 1988. To be published.

[Simic:90a] Simic, P. "Statistical mechanics as the underlying theory of 'elastic' and 'neural' optimizations," *Network*, 1:89–103, January 1990. Caltech Report C3P-787.

[Stein:89a] Stein, D. L., editor. *Lectures in the Sciences of Complexity*, volume 1. Addison-Wesley, Redwood City, California, lectures edition, 1989. Santa Fe Institute Studies in the Sciences of Complexity.

[Terzopoulos:83a] Terzopoulos, D. "Multilevel computational processes for visual surface reconstruction," *Computer Vision Graphics and Image Processing*, 24:52–96, 1983.

[Terzopoulos:86a] Terzopoulos, D. "Image analysis using multigrid relaxation methods," *IEEE Trans. Pattern Analysis Machine Intelligence*, 8:129, 1986.

[Wilson:88a] Wilson, G. V., and Pawley, G. C. "On the stability of the travelling salesman problem algorithm of Hopfield and Tank," *Biol. Cybern.*, 58:63–70, 1988.

[Williams:91a] Williams, R. D. "Performance of load balancing algorithms for unstructured mesh calculations," *Concurrency: Practice and Experience*, 1991. Accepted for Publication. Caltech Report C3P-913b.