PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES

Scientific Computation on SIMD and MIMD Machines

D. J. Wallace

Phil. Trans. R. Soc. Lond. A 1988 **326**, 481-498
doi: 10.1098/rsta.1988.0099

Phil. Trans. R. Soc. Lond. A. **326**, 481–498(1988) [481]
Printed in Great Britain

Scientific computation on SIMD and MIMD machines

By D. J. Wallace, F.R.S.
Physics Department, University of Edinburgh, James Clerk Maxwell Building, The King's Buildings, Mayfield Road, Edinburgh EH9 3JZ, U.K.

The ICL Distributed Array Processor and the Meiko Computing Surface have been successfully applied to a wide range of scientific problems. I give an overview of selected applications from experimental data analysis, molecular dynamics and Monte Carlo simulation, cellular automata for fluid flow, neural network models, protein sequencing and NMR imaging. I expose the problems and advantages of implementations on the two architectures, and discuss the general conclusions which one can draw from experience so far.

1. Introduction

At a practical level the gain in speed of computers, by a factor of roughly a million in the past 30 years, is due only in part to increases in the intrinsic speed of their components, which accounts for a factor of roughly 1000. The other factor of 1000 is due to the implementation of parallelism. For example, on the large scale, input and output from the computer are dealt with separately from the actual computation, and on a finer scale the multiplication of each of the digits of one number into the other can be done simultaneously. The kind of parallelism with which we are concerned at this meeting and in this paper is rather different: it is how many processors are organized to tackle a big problem cooperatively.

The idea of doing simultaneous calculations with a large number of computational units is not a new one; in fact it was recognized by Babbage last century (Hyman 1982, p. 242), well before the electronic computer was conceived. Another early and explicit pointer to the potential of parallel computing is given by Lewis F. Richardson in his book *Weather prediction by numerical process* (Richardson 1922). From chapter 11/2, I quote:

> If the time-step were 3 hours, then 32 individuals could just compute two points so as to keep pace with the weather, if we allow nothing for the very great gain in speed which is invariably noticed when a complicated operation is divided up into simpler parts, upon which individuals specialize. If the co-ordinate chequer were 200 km square in plan, there would be 3200 columns on the complete map of the globe. In the tropics the weather is often foreknown, so that we may say 2000 active columns. So that $32 \times 2000 = 64{,}000$ computers would be needed to race the weather for the whole globe. That is a staggering figure. Perhaps in some years' time it may be possible to report a simplification of the process. But in any case, the organization indicated is a central forecast-factory for the whole globe, or for portions extending to boundaries where the weather is steady, with individual computers specializing on the separate equations. Let us hope for their sakes that they are moved on from time to time to new operations.
>
> After so much hard reasoning, may one play with a fantasy?

His fantasy is illustrated in figure 1 (Lannerback 1984), which shows a uniformly spaced array of computers (who were *people*, of course, in Richardson's time), taking boundary data as required from their neighbours, and 'coordinated by an official of higher rank', who 'turns a beam of rosy light upon any region that is running ahead of the rest, and a blue light upon those

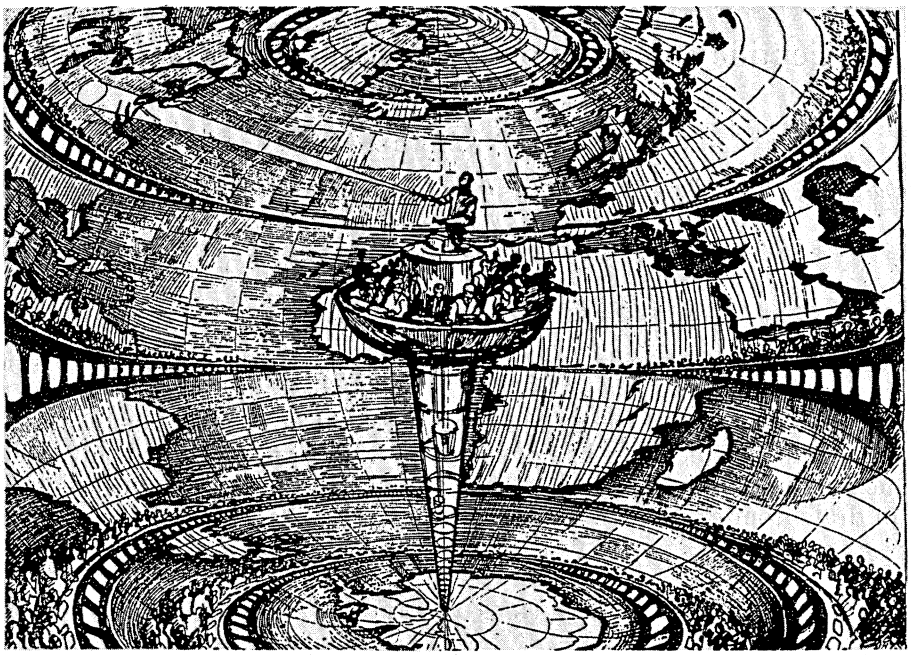


FIGURE 1. Richardson's scheme for numerical weather prediction by parallel computers, as shown by Lannerback (1984).

who are behindhand.' It is remarkable that so many aspects of parallel computing are recognized in this early work.

The range of architecture and organization of multiprocessor machines is of course vast. Most of this paper is based on the experience gained on two particular machines: the ICL (now AMT) Distributed Array Processor (DAP), an SIMD (single instruction multiple data) machine with fine-grain parallelism; and the Meiko Computing Surface, a reconfigurable transputer array. Following a brief reiteration of some of the arguments for exploiting parallel computers, I describe in §3 the dedicated resources available at Edinburgh. Section 4 gives examples of successful applications in physics, biology and image processing. In §5 I discuss the conclusions that one can draw from experience so far.

2. PARALLEL COMPUTING: WHY DO IT?

The above discussion has already highlighted the potential increase in performance, but it is worth expanding on a number of aspects.

2.1. *New science needs an increase by orders of magnitude*

An increase of computing resource by a factor of two makes only marginal impact on any one scientific problem. Consider, for example, the central problem of controlling statistical and systematic errors. Statistical errors reflect the number of independent configurations sampled or events generated. To halve the statistical error one needs to increase the number of configurations sampled, and thus the computing resource, by a factor of four. In many cases, systematic errors vary roughly linearly with the ratio of the grid size to the linear dimension of the system. Thus, if one is to reduce systematic errors by a factor of two, one needs a system

with $2^3 = 8$ times more degrees of freedom. Moreover, the number of updates required to generate significant changes in large-scale structures must also be increased, by a further factor of roughly 2^z, where $z \approx 2$, if the dynamical process is diffusive. This aspect is particularly important in phenomena like turbulent flow, which has significant structures (eddies) on many length scales.

In terms of physics, despite continuing progress, we are still very far from dealing computationally, in a fully controlled way, with the interacting many-electron problem which is at the heart of so many phenomena in condensed matter. Incorporating properly the Pauli exclusion principle for many-fermion systems remains a challenging problem for homogeneous materials. For real materials with defects and impurities, the reliable calculation of properties of great fundamental and practical significance, such as embrittlement and catalysis, makes even greater demands.

Finally, there is of course a host of problems which are only worth doing if they can be performed in real time, for which parallel computing may offer the only solution.

2.2. *The limits of silicon technology*

Whereas science and engineering need an increase in power by orders of magnitude, improvement in intrinsic properties of silicon-based devices appears to be limited by the need to have conductors sufficiently large to support non-ballistic conduction of electrons; device operation speed may increase by a factor of two or so, but there seems no scope for orders of magnitude. Because vector machines now produce an arithmetic result every machine cycle, we cannot expect to see dramatic advances from them: witness how supercomputer companies have moved towards multiprocessor machines.

Of course new technologies, with intrinsically faster electronic switching characteristics, will emerge: gallium arsenide, heterostructures(?), high-temperature superconductors(??). In any event, new technologies may also support the advantages of parallel computing, and one in particular, digital optical computing, is projected to exploit it in a massive way.

2.3. *Parallelism in physical systems*

It is therefore clear that if physical problems were amenable only to serial or vector computations, the outlook for obtaining significant increases in the necessary computational power would be bleak. However, most computationally demanding scientific problems have enormous inherent parallelism. The existence of event parallelism, geometric parallelism (domain decomposition), and algorithmic parallelism has already been exposed and discussed in some detail in previous talks (see, for example, the articles by Hey and Fox in this symposium). Of course it would be quite wrong to convey the impression that every scientific problem can be mounted efficiently on parallel hardware. What is indisputable, however, is that a large subset of problems can be, in very natural ways.

2.4. *Cost-effectiveness*

The assessment of the relative cost-effectiveness of different machines is fraught with difficulties. The crude estimate of megaflops per million dollars of capital cost fails to take into account subjective and historical factors such as ease of use and availability of software. However, even this measure is hard to quantify, because manufacturers' specifications may be misleading, as an unattainable peak performance is usually quoted. Moreover, direct

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES

comparisons between mature and newly announced machines may be misleading in a period of rapid increase in performance characteristics. Finally, the spectrum of performance of the same parallel machine on different problems is much wider than for a conventional architecture. Nevertheless, existing commercial multiprocessor machines appear to be roughly five times more cost-effective on this scale than machines which rely purely on vector capability: say, 50 realistically realizable Mflops/M$ for vector, compared with 250 Mflops/M$ for multiprocessor. (The rapid increase in price-performance, perhaps by a factor of two every year, means that each of these figures is highly time-dependent.) Even higher figures can be quoted for special purpose computers if one counts only the cost of raw silicon and the development effort is not included. Such machines are also likely to be less flexible, and less 'user-friendly' unless considerable system software effort is also expended.

2.5. *Why not?*

At present, of course, there is a penalty to be paid for all of these advantages: the parallelism must be harnessed to applications. For the most powerful parallel machines today this does involve recoding. It is our clear view that the gain in cost-effectiveness fully justifies the recoding effort for a wide range of problems. In the future, as parallel systems become more highly developed and our experience of using them increases, this recoding effort will certainly decrease; vector machines will presumably then have to become more price competitive, but the absolute performance advantage of multiprocessor machines will remain a key factor.

3. Resources at Edinburgh

The existence at Edinburgh of dedicated parallel hardware has been a crucial factor in the expansion of the user community (currently more than 100).

Work on applications of parallel computing began in 1980 with G. S. Pawley's use of the ICL Distributed Array Processor (DAP) at Queen Mary College for molecular dynamics studies. This DAP is an SIMD machine comprising a 64×64 array of simple processing elements, each with 4 Kbits of associated memory (now 16 Kbits on the Queen Mary College machine); an ICL 2900 mainframe acts as the host. For further information on the DAP and its software, see, for example, Hockney & Jesshope (1981). Each processing element communicates with its four nearest neighbours in the array (North, South, East and West), and the user can arrange the array to have either fixed or cyclic boundary conditions. Each processor can perform only bit-serial arithmetic, that is, operating on one bit at a time, so arithmetic operations must be achieved in software. This means that the DAP offers a flexibility in word length unavailable in conventional computers, which usually allow only 16-bit, 32-bit or 64-bit operations. The DAP is programmed in DAP-FORTRAN; the host machine uses standard FORTRAN. Existing software routines therefore need to be converted to DAP-FORTRAN. Our experience is that it is easy to achieve a performance of around 20 Mflops on many problems, with the DAP. For problems with short words, and in particular for parallel bit-manipulation, the DAP is an extremely powerful architecture; see remarks in the examples. Brief descriptions of projects and a list of references is contained in Bowler *et al.* (1987*a*).

Our early experience of the performance of the machine at Queen Mary College was sufficiently convincing that we were led to prepare a proposal to SERC to site a DAP at Edinburgh. The success of this proposal and the work emerging from it resulted in the gift of

a second DAP from ICL and these two machines provided a superb resource, with about 180 publications emerging from this work, covering a much wider range of science than was envisaged in the original proposal. The DAPs were necessarily decommissioned with the replacement of the ICL hosts with new University mainframes at the end of July 1987. Work on this architecture will continue with the installation of an Active Memory Techology DAP 510 system under an Alvey grant. This 32×32 array is hosted by a SUN Workstation (or microVAX). The minimum memory size is 4 Mbytes, with possible expansion up to 128 Mbytes, removing the major limitation of the mainframe DAPs, which had only 2 Mbytes.

In anticipation of the likely loss of the DAPs, we were fortunate to acquire, in April 1986, with support from the Department of Trade and Industry and the Computer Board, the first Meiko Computing Surface delivered to a University. The Meiko machine (Bowler *et al.* 1987*b*) is an electronically reconfigurable transputer array with advanced graphics capability. This demonstrator machine consists of 40 T414 transputers each with 256 Kbytes of RAM (random access memory), and display system, and is hosted by a microVAX. The impending loss of the DAPs, the hardware reliability and software environment of the demonstrator system, and an evaluation exercise carried out over the Summer of 1986, led to the preparation of the Edinburgh Concurrent Supercomputer proposal in September 1986. Phase One support for this project has been awarded by the DTI, Computer Board and SERC, providing a 32-processor (T414 each with 3 Mbyte) development farm, four display systems, 4.5 Gbytes of disc capacity, and compute resource consisting of 200 floating point transputers (T800s) each with 4 Mbytes of local memory. The UNIX†-like operating system offers to the user who is accessing the machine from the network a choice of domains of resource (presently set by the operator). The sustainable performance of the full array is around 200 Mflops. The facility is run for University and national users by the Edinburgh University Computing Service. A condition of DTI support was the establishment of an industrial affiliation scheme, which has already met the initial target of £250 k commitment (in cash and kind). The Phase Two proposal to enhance the system towards the target configuration of 1024 T800s is presently under consideration. Further general information can be obtained from the project newsletter (Wilson 1987).

It is anticipated that these resources will continue to be used primarily for scientific and engineering applications, although artificial-intelligence projects are now beginning to emerge also: the Parallel Architectures Laboratory in the Artificial Intelligence Applications Institute at Edinburgh has substantial parallel hardware specifically for AI applications. The optical computing projects at Heriot Watt University should also be noted, likewise that Bolt Berenek and Newman are installing a 32-processor Butterfly at their European headquarters in Edinburgh.

4. Examples

The following paragraphs review a selection of successful applications of the work so far at Edinburgh; the list is certainly not exhaustive, but is intended to give some impression of the breadth of activity and the particular strengths of each architecture.

† UNIX is a registered trade mark of AT&T in the U.S.A. and other countries.

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES

4.1. *Experimental data analysis*

Most of my examples are based on theoretical modelling and simulation, but computational demands in data analysis are also becoming increasingly onerous, so it is pertinent to consider an example in this area. The potential of 'event parallelism' in the analysis of high-energy physics data has already been noted by Glendinning & Hey (1987). The particular example in condensed matter physics on which we focus here is the calculation of full resolution corrections in neutron scattering data (Mitchell & Dove 1985).

Typically, a neutron inelastic scattering experiment is designed to measure the scattering function $S(\boldsymbol{Q},\Omega)$, which contains information about the microscopic static and dynamic properties of the system under study. However, what is actually measured is a convolution of $S(\boldsymbol{Q},\Omega)$ with some experimental resolution function which in the general case also depends upon the four variables $\boldsymbol{Q}$ and Ω. Such corrections must be made to compare experimental results with theoretical predictions, and can be computationally intensive.

To ease this problem, a package was written for the ICL DAP which exploits algorithmic parallelism for each data point, to perform these corrections for a user-defined model and to fit the resulting theoretical scattering function to the experimental data. The loop over all data points is in fact done serially, although it offers further obvious scope for parallelism. Applications studying scattering by spin waves and phonons in a number of magnetic materials are reported in (Mitchell & Dove 1985). The DAP program enables interactive fits to be made in roughly one minute or less, compared with roughly 20 minutes of CPU time for the equivalent code on a mainframe. This package was in routine use at Edinburgh, until the decommissioning of the DAPs; its advantage for enhancing the effectiveness of research work in this area is obvious.

4.2. *Molecular dynamics*

This is another area which is manifestly amenable to parallel computation, because the time-stepping is naturally done by the simultaneous calculation of the forces on all the N molecules. Usually the number of processors available is less than or equal to N, but even when they are greater than N they can still be used efficiently either by distributing the force calculation for each molecule across a number of processors, or by running independent simulations in parallel to accumulate statistics. Because realistic modelling of solid-state phase transitions requires a sample large enough that the nucleating phase is not restricted or determined by the imposed boundary conditions, significant simulations offer scope for massive parallelism.

The particular example on which we focus is an early study (Pawley & Thomas 1982) of the plastic-to-crystalline phase transition in sulphur hexafluoride, SF_6, with the DAP. The octahedral shape of the molecule is conducive to the formation of a plastic phase at intermediate temperatures, in which the molecules form, on average, a body-centred cubic lattice, and perform occasional reorientational jumps. When a sample (roughly $13 \times 13 \times 13$ BCC unit cells in practice) is simulated at low temperature (25 K), however, the temperature of the sample slowly drifts up as equilibration proceeds, requiring extraction of kinetic energy to maintain the nominated temperature. The total potential energy of the system is accordingly falling (and the volume of the constant-pressure sample is decreasing), pointing to a gradual ordering process.

The obvious question is: what is the nature of the new ordered state? Figure 2 is taken from Pawley & Thomas (1982); it represents a two-dimensional section through a sample after such

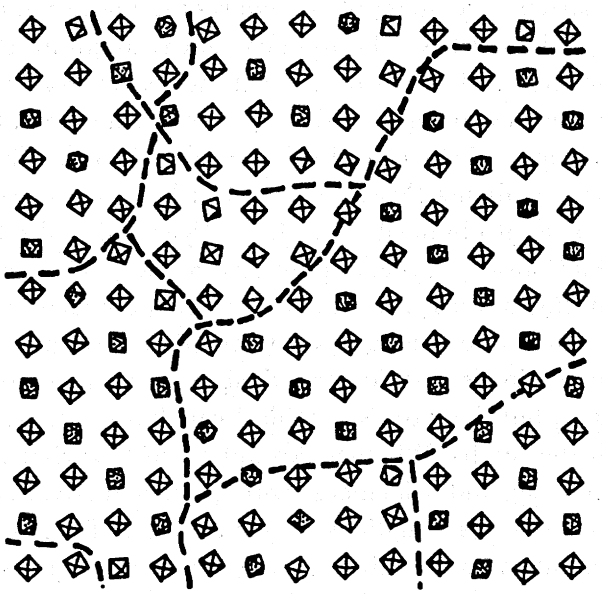


FIGURE 2. Two-dimensional section through a simulation (Pawley & Thomas 1982) of SF_6 at 25 K. The molecules are represented by octahedra. There is clear evidence of microdomains of the low-temperature phase which involves orientational reordering of the SF_6 molecule.

a simulation and shows clearly a mosaic containing microcrystals in which the now triclinic unit cell contains one molecule of one orientation and two molecules of a second orientation. The new structure is well supported by experimental results from neutron powder diffraction refinements (Garg 1977; Dove *et al.* 1988). The figure underlines the necessity of a large enough simulation to support a mosaic of the new structure; clearly simulations with more complicated molecules or ordering require correspondingly even more computational resource.

4.3. *Cellular automata*

Cellular automata are arrays of discrete cells which can contain degrees of freedom which take on discrete values (boolean variables 0 or 1 in the simplest case). These variables evolve in time according to some transition rules which are dependent on the state of variables in neighbouring cells and may be deterministic or stochastic (i.e. affected by noise). In one sense therefore, they can be thought of as primitive molecular dynamics. The motivation for studying them is (at least) twofold. From the point of view of physics, although their microscopic behaviour may not correspond to any specific physical system, their behaviour on distance scales large compared with the cell size can describe macroscopic continuum phenomena. The spirit here is very much akin to universality phenomena at phase transitions underpinned by renormalization group theory (for an early review, see Wilson & Kogut 1974). From the point of view of computation, they are suitable for digital computer simulation, particularly on SIMD computers with powerful parallel bit-manipulation capability like the DAP, Connection Machine or Goodyear MPP.

Microcanonical simulations of the Ising model of a uniaxial ferromagnet can be formulatd in this way for example, but the particular case study on which we focus is cellular automaton modelling for fluid flow (Hardy *et al.* 1976; Frisch *et al.* 1986; Salem & Wolfram 1986). These models represent an extension of the lattice gas concept from statistical mechanics to

hydrodynamics, with 'particles' hopping along bonds of the lattice and scattering from each other according to simple local rules. The primary aim is to ensure that the Navier–Stokes equation emerges on the large scale. It turns out that the discrete symmetry of a square-lattice automaton survives in the macroscopic limit. However, a hexagonal lattice has sufficient symmetry to ensure isotropy, which can also be ensured in three dimensions by allowing hopping beyond nearest neighbours (Frisch *et al.* 1986; Wolfram 1986; Frisch *et al.* 1987).

In figure 3 we show a result obtained by B. J. N. Wylie, using the Computing Surface at Edinburgh (Kenway, McComb & Wylie, unpublished results). The simulation depicts a jet of fluid injected from a pipe into a transverse flow. The system permits the specification of barriers of any shape in the flow, and accommodates the interaction of complex flows as in the figure. This type of bit-serial simulation is more ideally suited to the architecture of the DAP than that of the Computing Surface, but the latter's graphical capability was crucial in motivating the work and in visualizing the results. Whether this approach to turbulent simulation captures hydrodynamic flow effectively, and will emerge with significant advantages over conventional methods using the Navier–Stokes equations, remain matters for research and debate; the existence of parallel computers has certainly been a major factor in stimulating that debate.

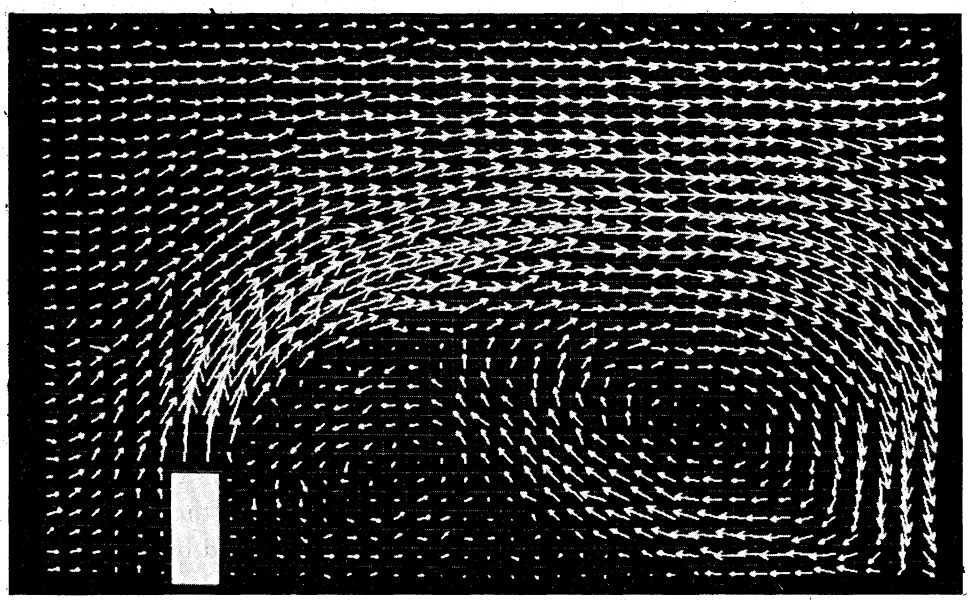


FIGURE 3. Cellular automaton simulation on the Computing Surface (Kenway, McComb & Wylie, unpublished results) illustrating injection from a pipe into a transverse flow.

4.4. *Monte Carlo simulation*

Several distinct kinds of computation come under this heading. For example, it is an integral part of radiation metrology generally, and of the calibration of detector performance in high-energy physics experiments in particular, by studying the response of the detector to events generated at random according to some model of the collision processes. In constrast, the kind of calculation on which we focus is the Monte Carlo simulation of the canonical ensemble in some thermodynamic problem. In particular, we consider an example from the study of critical phenomena at phase transitions. Such problems are very demanding in computing resources,

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES

because the critical singularities emerge only in the infinite volume limit, and their true universal values may be obscured by the finite-size effects which are always present in the finite systems studied on a computer. Moreover, whereas configurations may evolve rapidly on the small (e.g. lattice) scale under the simulation, on the scale of the correlation length, ξ, they evolve with a characteristic time which increases as ξ^z where $z \approx 2.0$. Thus, to obtain reliable results we need large systems and long runs. The calculations are well suited for parallel computation, because any subset of variables can be updated simultaneously and independently, provided there is no direct interaction between them in the hamiltonian.

The particular study we report here concerns the issue of hyperscaling in the three-dimensional Ising model (Wall 1986; Freedman & Baker 1982). The question is whether in this model the relations between the critical exponents governing thermal properties and those governing the correlation length are as predicted by the renormalization group (Wilson & Kogut 1974) (and hyperscaling arguments; see Fisher (1983 and references therein)). Freedman & Baker (1982) studied this problem by considering the quantity

$$g_R = \frac{\chi^{(4)}}{(\chi^{(2)})^2 \xi^d},$$

where $\chi^{(2)}$ is the two-point magnetization cumulant (the susceptibility), $\chi^{(4)}$ is the four-point cumulant and ξ is the correlation length. If one varies the system size L while keeping the ratio ξ/L constant, general arguments indicate that (for large enough L), g_R should behave like

$$g_R \propto L^{-\omega^*},$$

where ω^* is critical exponent; according to hyperscaling and the renormalization group, for the three-dimensional Ising model, ω^* should be zero and g_R should tend to a non-zero constant. Numerical simulation by Freedman & Baker (1982) suggested the value $\omega^* = 0.20(8)$. This calculation was extended on the ICL DAP (Wall 1986), with a lattice of up to 128^3 spins, generating (for the largest lattice) some 70 million configurations using code running at more than 200 million single spin update attempts per second. The combined results are shown in figure 4. The levelling off of the cumulant ratio is rather convincing in the high statistics DAP results, and a fit to the data yielded $\omega^* = 0.008(24)$, in good agreement with the renormalization group predictions.

4.5. *Percolation*

It is well known that percolation processes provide examples of critical phenomena; their reliable study can therefore be as computationally demanding as phase transitions.

In its simplest form, the problem concerns the statistical properties of clusters formed by depositing sites (or bonds) on a lattice with some probability, p. In particular, one is concerned with the universal critical exponents governing the scale of the clusters, their total number, size distribution, etc., in the neighbourhood of the critical concentration p_c (the smallest value of p necessary to generate an infinite cluster). The conventional theoretical analysis predicts that, for example, the singular part of the mean number of clusters per site has the form (Stauffer 1979)

$$K_s(p) = D\,|p - p_c|^{2-\alpha}$$

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES

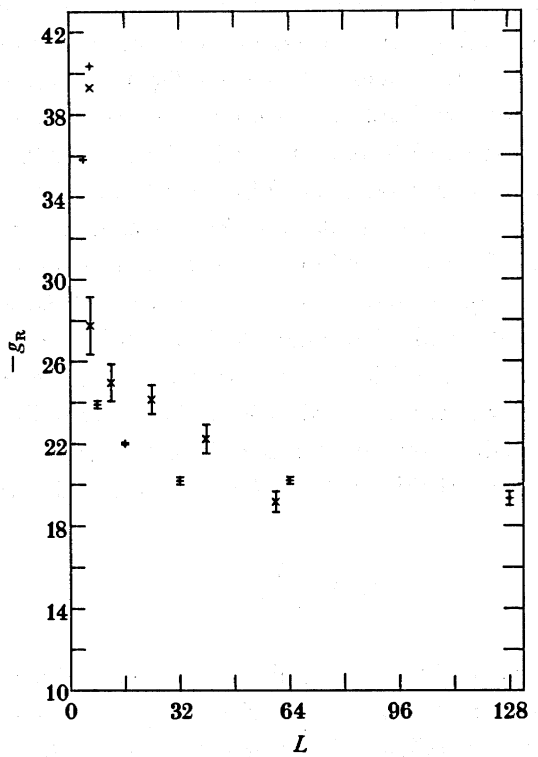


FIGURE 4. Plot of the renormalized coupling constant g_R of the three-dimensional Ising model obtained (×) by Freedman & Baker (1982) ($-\omega^* = -0.20(8)$), and (+) by Wall (1986) ($-\omega^* = -0.008(24)$), plotted as a function of the linear dimension L of the lattice; hyperscaling is valid if g_R tends to a constant asymptotically.

for p near p_c, where $\alpha = -\frac{2}{3}$. However, it has been suggested recently on the basis of numerical work and series expansion studies (Jug 1985) that the singular part K_s should have the form

$$K_s(p) = D(p-p_c)^2 \ln|\ln|p-p_c||.$$

This controversy motivated a study of the problem on the ICL DAP (Dewar & Harris 1987). As part of that work a parallel algorithm was developed for counting clusters in two dimensions by collapsing them to a single site (or spanning loop) by using fast parallel bit manipulations. Towards the end of this computation, when most of the clusters have already been collapsed, the remaining 'active' cluster sites are rather sparsely distributed across the processor array, and efficiency is correspondingly low. However, the DAP code (Dewar & Harris 1987) ran some 20 times faster than a serial algorithm on the CRAY 1 (Jug 1985). This is an interesting case in which the power of parallel bit-manipulation on the DAP compensates for its rigid SIMD parallelism.

In exposing the singularity in K, it is convenient to eliminate the leading regular terms by considering the third derivate with respect to p, K'''. The results for this quantity at the critical concentration, as a function of lattice size L, are shown in figure 5. The DAP results are in complete agreement with the conventional scaling result: $K''' \propto L^{\frac{1}{2}}$.

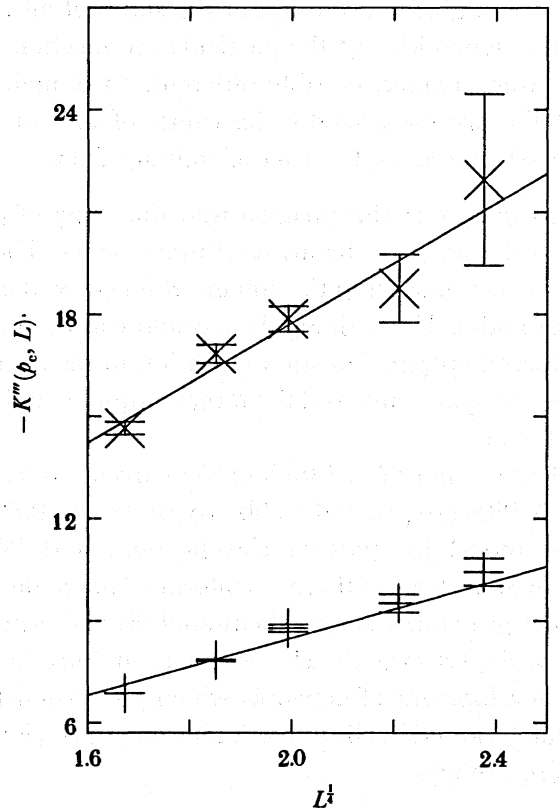


FIGURE 5. In the conventional theory of percolation, the third derivative of the cluster density is predicted to diverge at the percolation concentration as $L^{\frac{1}{2}}$ in two dimensions, where L is the linear dimension of the system. These results are taken from a simulation on the ICL DAP with a fast parallel algorithm for cluster counting (Dewar & Harris 1987).

4.6. *Neural Network Models*

The remarkable processing capabilities of the nervous system, vision, speech-recognition, motor control, reasoning, association, etc., are achieved despite the fact that the typical timescales of the biological 'wet-ware' are of the order of milliseconds, rather than the nanoseconds of modern silicon computers. Neural network models attempt to capture the key ingredients responsible for these faculties; among them is undoubtedly massive parallelism.

A wide range of models has been developed for a variety of purposes; they differ in structure and details, but share some common features:

they contain nodes, or units, which are extreme simplifications of neurons, in the sense that the state of each node is usually described by a single real variable, representing its firing activity;

the nodes are connected, usually in pairs, so that the state of one node affects the potential of all the nodes to which it is connected according to the weight, or strength, of the connection;

the new state of a node is a nonlinear function of the potential created by the firing activity of the other neurons;

input to the network is done by setting the states of a subset of the nodes to specified values; this sets up an image or pattern of activity on these 'input' nodes;

the processing takes place through the evolution of the states of all the nodes on the net, according to the details of the dynamics and the particular connection strengths, until some output activity can be read from another, possibly different, set of nodes;

the training of the net is the process whereby the values of the connection strengths are modified to achieve the desired processing for a set of training data.

For example, for image processing, one can imagine that the array of pixel values from the input image is mapped on to the states of the array of input nodes. The image processing is effected by the dynamics of the net producing the enhanced image or the features identified in the image, etc., at the output nodes. The training data would consist of a set of possibly noisy images with their known, desired output. For an application in medical diagnosis, the input data would be an encoding of the symptoms, and the target output would be the diagnosis, and possibly recommended treatment.

Models based on these ideas are not new. Much of the current effort can be traced to the seminal work of McCulloch & Pitts (1943), and Hebb (1949). In the 1960s, Rosenblatt (1962) developed the 'perceptron' model for pattern classification, and Widrow *et al.* (1960) demonstrated weather prediction for southern California using the 'adaline' analogue computer based on neural net principles. An excellent analysis and critque of the perceptron theory is given by Minsky & Papert (1969). The recent resurgence of interest was in large measure stimulated by the development of networks which go beyond the limitations of the perceptron model; reviews can be found in Hinton & Anderson (1981), Rumelhart *et al.* (1986), Denker (1986) and Grossberg (1987).

4.6.1. *Optimization by analogue neurons*

Analogue neurons were introduced by Hopfield & Tank (1985) as a general technique for optimization problems involving boolean variables. The method was applied by them originally to the travelling-salesman problem (see next section), although it turns out to be not very effective at that particular task (Wilson & Pawley 1987). It has also been used to perform analogue to digital conversion (Tank & Hopfield 1985) and load balancing in parallel computing (Fox & Furmanski 1987). We focus here on image restoration, using the algorithm of Geman & Geman (1984) for binary images which have been corrupted by noise, which had previously been studied by Murray *et al.* (1986) using the simulated annealing algorithm (Kirkpatrick *et al.* 1983). The analogue neuron scheme (Forrest 1987) involves representing the array of pixel intensities as a network of neurons, each of which can 'fire' on a continuous scale from non-firing ('black' pixel) to fully-firing ('white' pixel). The dynamics of the pixel interactions ('neural activities') is controlled by a cost function determined by the input data and by *a priori* assumptions about the statistical properties of the 'clean' images (for example, 'edges are rare'). The 'best' restored image is that which minimizes this cost function, and the dynamics is designed to achieve at least a good approximation to this.

This problem is intrinsically SIMD-parallel, with short-range communications, so that it can be expected to run efficiently on most parallel machines; at Edinburgh it was studied both on the DAP (Forrest 1988) and on the Computing Surface (D. Roweth, unpublished results).

The performance of the restoration by this analogue neural network method was compared to that of a simple majority-rule scheme (where each neuron continually adopts the intensity of the majority of its four nearest neighbours, or remains unchanged if exactly two of its

neighbours are 'white', until the image stabilizes). A third restoration method, performing a gradient descent, was achieved by restricting the neuron firing rates to discrete values ('on' or 'off'). The analogue neural network method consistently finds lower cost solutions (Forrest 1987) than these schemes, as shown in figure 6.

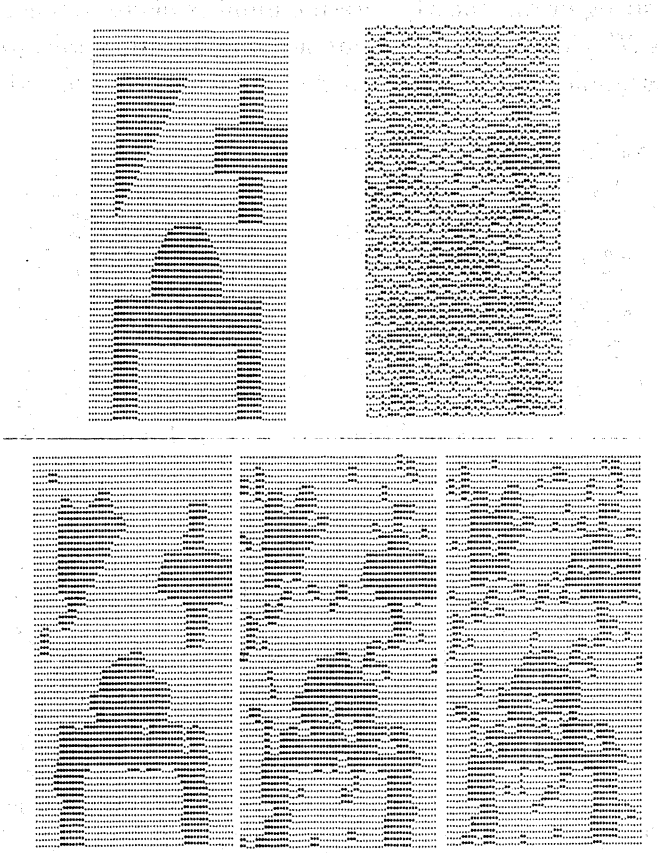


FIGURE 6. Image restoration by analogue neurons (Forrest 1988), in the framework of Geman & Geman (1984). The top two images are an original, and with 30% noise. The three restored images at the bottom were produced by analogue neurons, gradient descent, and a majority rule, from left to right respectively.

4.6.2. *The elastic net*

In a recent paper, Durbin & Willshaw (1987) described what they called the 'elastic net' method for solving the travelling-salesman problem in the plane. The method is generally applicable to problems involving mapping between spaces of different dimensions (Mitchison & Durbin 1986). For the travelling-salesman problem, a closed loop of elastic 'string' is placed in the plane containing the cities which the salesman is to visit, and then slowly deformed into a path which connects all the cities.

The elastic string is modelled as a set of discrete points, each of which is connected to its two neighbours. Attractive forces between each point and its neighbours hold the string together, while each city exerts an attractive force on each point, pulling the point towards it. As the point-to-point forces are relaxed and the city-to-point forces strengthened, the string is gradually deformed into a path. The string dynamics is intrinsically parallel, and has been implemented by a straightforward 'domain decomposition' on the Computing Surface

(D. Roweth & G. V. Wilson, unpublished results). The performance is shown in figure 7 from Durbin & Willshaw (1987).

The parallel elastic net solution to the travelling salesman problem is another example which becomes more efficient as it becomes larger. Because the number of points which must be used to model the elastic string grows linearly with the number of cities, the amount of calculation at each step grows as N^2. However, the amount of communication only grows as N, so the ratio of calculation to communication improves as the problems being solved grow larger.

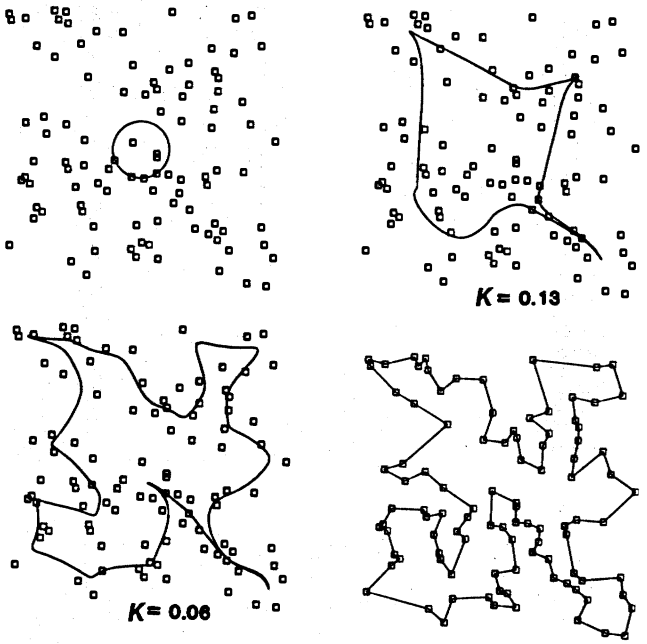


FIGURE 7. Example from Durbin & Willshaw (1987) of the progress of the elastic net method for 100 cities randomly distributed on the unit square. The initial path is a circle, which evolves over the next three pictures to a low-cost, legal tour of the cities.

4.7. *Protein sequence analysis*

Molecular biology has been revolutionized by the development of fast sequencing techniques for nucleic acids. The rate of acquisition of protein sequence data has correspondingly accelerated, and this has led to the urgent need for adequate comparative sequence analysis, to promote the efficient use of other research resources.

Proteins and nucleic acids (the genetic material) are linear polymers whose sequences may be represented by character strings, with a 20-letter alphabet for proteins and a 4-letter alphabet for nucleic acids. The international database collections of sequences are prime resources for molecular biological research. These databases are currently small; the protein database has approximately one million characters of sequence information, and the genetic base has ten million, but already the task of searching them has led to the development of a number of approximate methods for making comparisons. However, the application of the exhaustive inexact string-matching algorithms reviewed by Sellers (1980) has been beyond the capacity of many workstations and mainframe computers. The situation will deteriorate further, as the databases are growing exponentially, doubling in size every two years or less.

A suite of programs for exhaustive inexact string-matching has been developed for the DAP

by Lyall *et al.* (1986); the most valuable of these, especially for novel proteins, has implemented the 'Best Local Similarity' algorithm of Smith & Waterman (1981). The programs exploit the variable wordlength flexibility on the DAP, and perform 4096 comparisons simultaneously (Coulson *et al.* 1987).

They have been extensively applied in several thousand searches, leading to discoveries of biological significance. These include the relationship of the cystic fibrosis antigen to the bovine s-100a alpha protein chain (Dorin *et al.* 1987), and the relation of vitellogenins in *Drosophila melanogaster* to porcine triasyl glycerol lipase (Bownes *et al.* 1988). A similarity between prokaryotic and eukaryotic cell cycle proteins has also been discovered (Robinson *et al.* 1987).

This work is planned to continue on the AMT DAPs and on the Computing Surface, where the latter's FORTRAN Farm facility supports parallel comparisons transparently across the database. This is one particular example of the growing application area of parallel databases.

4.8. *Medical imaging*

New non-invasive medical techniques such as NMR imaging are being used in bodyscanners to produce three-dimensional (tomographic) images of the body. There have been several attempts over the past few years to produce systems which are capable of manipulating and displaying the vast quantities of data which are required to generate each image, but a satisfactory performance has still not been achieved. Three-dimensional image processing suffers from the same speed problems as in two dimensions, but exaggerated by at least one order of magnitude because there are ten or more 2D sections in one 3D image, and the image-processing operations are inherently more complex.

A key problem in implementing this type of processing on a parallel machine is to achieve a high efficiency of processor usage while minimizing communications overheads. This is typically straightforward for low-level image processing, but less so for intermediate and high-level functions. For example, the common requirement of tracking a surface through the data could very easily result in one processor at a time doing all the work, with all the rest lying idle. In particular, the difficulty of mapping a three-dimensional image on to a two-dimensional array of processors means that data at the boundary may need to be passed not to an immediately adjacent neighbour in the array (as, for example, in the simulation of fluid flow, where the same local operations can be performed globally right across the entire array of processors), but to some more remote processor, depending on local conditions within the data. Whereas most of the previous examples have been well suited for SIMD parallelism, the higher level processing in this example certainly benefits from the additional flexibility offered by MIMD (multiple instruction multiple data).

The first phase in handling NMR bodyscan data on the Computing Surface is now complete (Norman 1987). The system includes a generalization of the Zucker–Hummel surface detection operator to a non-square metric, a three-dimensional surface display, and an implementation of a novel and completely asynchronous arbitrary communications network which carries messages around the processors, ensuring that all are operating as closely as possible to maximum efficiency. A typical display is shown in figure 8. The computation runs on 40 T414 transputers at 500 times the speed of a SUN 2, and the display speed is some 50 times greater.

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES

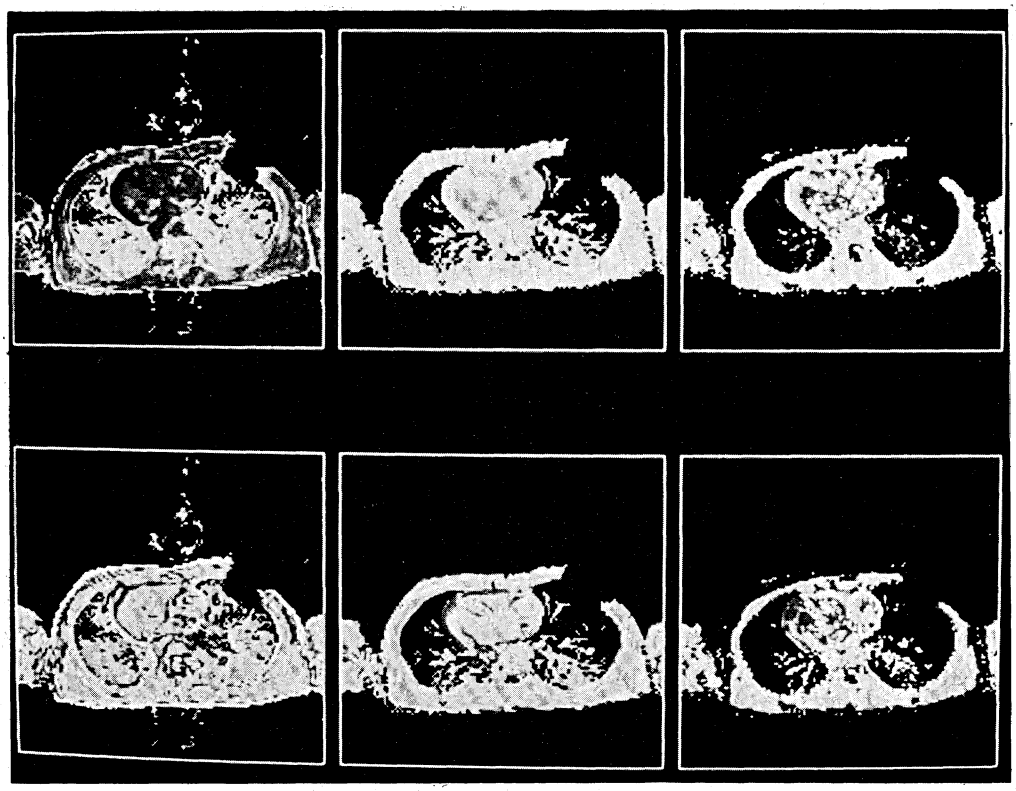


FIGURE 8. Three NMR datasets showing the same cross section of the human thorax. The top row are smoothed images after thresholding. The bottom layer show the results of applying the Zucker–Hummel surface operator to the top row.

5. CONCLUDING REMARKS

I have commented in the text on the mapping of the particular applications on to two types of parallel machine. Two further comments are pertinent here. First, most of the examples discussed in this paper have been naturally suited to SIMD computation. This is only in part a result of the 'historical accident' of the earlier availability of the SIMD DAP machine; it also accurately reflects the wide range on such applications. Second, the performance of the two machines across the range of applications has in fact been dominated rather more by other aspects, in particular the fine-grain against coarse-grain parallelism.

Compared with the theoretical and experimental disciplines, computational science is only in its infancy. Increasing computer resources have already opened up topics for exploration which are inaccessible to direct experimental or theoretical attack. This process will only be accelerated as the software to support applications programming becomes more powerful and usable.

However, if computational science is in its infancy, the impact which parallel computing can make has barely been conceived. More powerful and more cost-effective machines are emerging annually. Developing code for these machines is a challenge, but one which has considerable intellectual satisfaction in its own right, particularly because many physical problems lend themselves to natural solutions. At the same time new software tools will inevitably enhance portability and programmer efficiency; graphical aids are likely to be particularly important in this respect. The case studies outlined in this paper are of course only

highly selective examples obtained from a very short period of activity. Nevertheless, I believe that in conjunction with the conceptual developments reported in other papers in this symposium, they are a sound basis for the claim that, in the future, parallel computing will be an increasingly stimulating tool for research across many areas of science.

I am grateful to many colleagues at Edinburgh and elsewhere, without whose efforts the material for much of this paper would not have existed. I thank N. Stroud also for help in preparing the manuscript, B. Randell for helpful information on Babbage, and P. W. White for bringing the Richardson reference to our attention and for making figure 1 available to us.

References

Bowler, K. C., Bruce, A. D., Kenway, R. D., Pawley, G. S. & Wallace, D. J. 1987*a* Scientific computation on the Edinburgh DAPs, final report. University of Edinburgh Internal Report.

Bowler, K. C., Kenway, R. D., Pawley, G. S. & Roweth, D. 1987*b* *An introduction to Occam 2 programming*. Lund: Chartwell-Bratt.

Bownes, M., Shirras, A., Blair, M., Collins, J. & Coulson, A. 1988 *Proc. natn. Acad. Sci. U.S.A.* **85** (In the press.)

Coulson, A. F. W., Collins, J. F. & Lyall, A. 1987 *Comput. J.* **30**, 420–424.

Denker, J. S. (ed.) 1986 Neural Networks for Computing. In *AIP Conference Proceedings* **151**. New York: American Institute of Physics.

Dewar, R. & Harris, C. K. 1987 *J. Phys.* A **20**, 985–993.

Dorin, J. R., Novak, M., Hill, R. E., Brock, D. J. H., Secher, D. S. & van Heyningen, V. 1987 *Nature, Lond.* **326**, 614–617.

Dove, M. T., Powell, B. M., Pawley, G. S. & Bartell, L. S. 1987 (In preparation.)

Durbin, R. & Willshaw, D. J. 1987 *Nature, Lond.* **326**, 689–691.

Fisher, M. E. 1983 Critical phenomena. In *Lecture notes in physics* (ed. F. J. W. Hahne), p. 186. Berlin: Springer-Verlag.

Forrest, B. M. 1988 In *Parallel Architectures and Computer Vision*. (ed. I. Page). Oxford University Press.

Fox, G. C. & Furmanski, W. 1987 Caltech preprint C3P 363.

Freedman, B. A. & Baker J. Jr 1982 *J. Phys.* A **15**, L715–L721.

Frisch, U., Hasslacher, B. & Pomeau, Y. 1986 *Phys. Rev. Lett.* **56**, 1505–1508.

Frisch, U., d'Humieres, D., Hasslacher, B., Lallemand, P., Pomeau, Y. & Rivet, J.-P. 1987 *Complex Systems* **4** (1) (In the press).

Garg, S. K. 1977 *J. chem. Phys.* **66**, 2517–2524.

Geman, S. & Geman, D. 1984 *IEEE Trans.* **PAMI 5**, 721–741.

Glendinning, I. & Hey, A. J. G. 1987 *Comput. Phys. Communs* **45**, 367–371.

Grossberg, S. (ed.) 1987 *The adaptive brain*, vols 1 and 2. Amsterdam: North Holland.

Hardy, J., de Pazzis, O. & Pomeau, Y. 1976 *Phys. Rev.* A **13**, 1949–1961.

Hebb, D. O. 1949 *The organisation of behaviour*. Wiley: New York.

Hinton, G. E. & Anderson, J. A. (eds) 1981 *Parallel models of associative geometry*. Hillsdale, New Jersey: Lawrence Erlbaum.

Hockney, R. W. & Jesshope, C. R. 1981 *Parallel computers*. Bristol: Adam Hilger.

Hopfield, J. J. & Tank, D. W. 1985 *Biol. Cybernet.* **52**, 141–152.

Hyman, A. 1982 *Charles Babbage; pioneer of the computer*. Oxford University Press.

Jug, G. 1985 *Phys. Rev. Lett.* **55**, 1343–1346.

Kirkpatrick, S., Gelatt, C. D. & Vecchi, M. P. 1983 *Science, Wash.* **220**, 671–680.

Lannerback, A. 1984 *Dagens Nyheter*, p. 35. Sweden.

Lyall, A., Hill, C., Collins, J. F. & Coulson, A. F. W. 1986 In *Parallel computing '85* (ed. M. Feilmeier, G. Joubert & U. Schendel), pp. 235–240. Amsterdam: North-Holland.

McCulloch, W. S. & Pitts, W. A. 1943 *Bull. math. Biophys.* **5**, 115–133.

Minsky, M. & Papert, S. 1969 *Perceptrons: an introduction to computational geometry*. Cambridge, Massachusetts: MIT Press.

Mitchell, P. W. & Dove, M. T. 1985 *J. appl. Cryst.* **18**, 493–498.

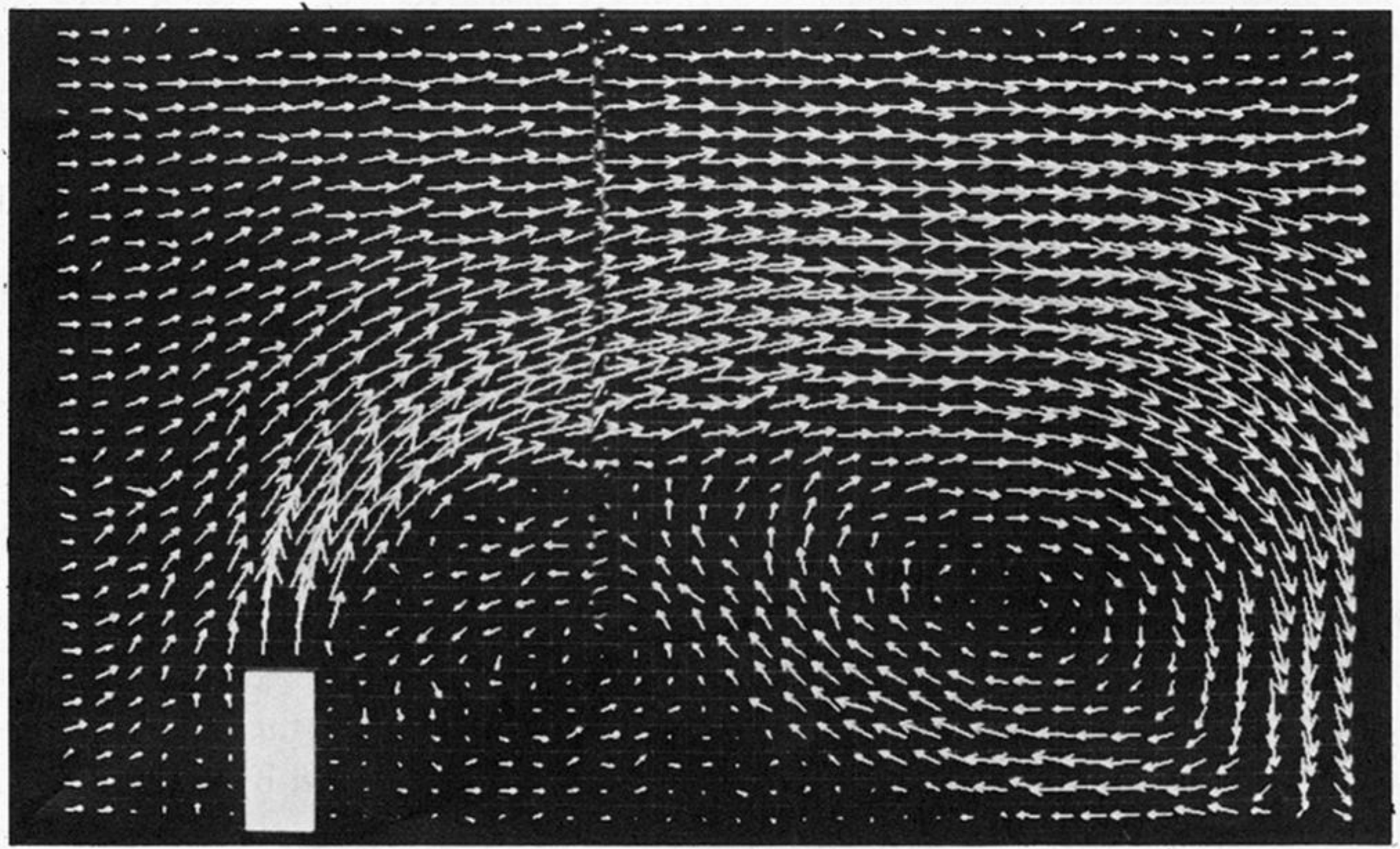
Mitchison, G. J. & Durbin, R. 1986 *SIAM J. Alg. Disc. Meth.* **7**, 571.

Murray, D. W., Kashko, A. & Buxton, H. 1986 *Image Vision Comput.* **3**, 133–142.

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES

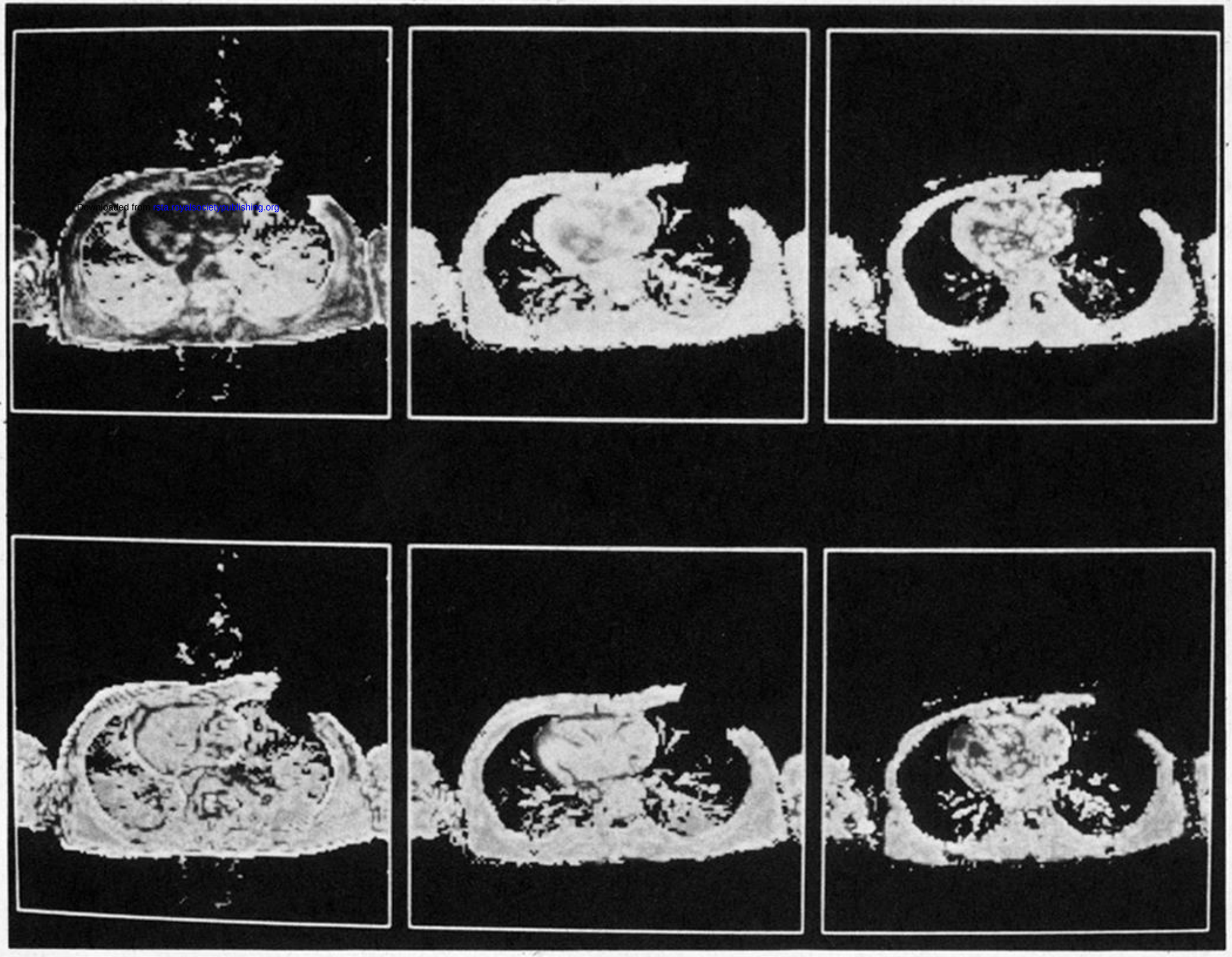
Norman, M. G. 1987 A three-dimensional image processing program for a parallel computer. M.Sc. Thesis, Dept of Artificial Intelligence, University of Edinburgh.
Pawley, G. S. & Thomas, G. W. 1982 *Phys. Rev. Lett.*, **48**, 410–413.
Richardson, L. F. 1922 *Weather prediction by numerical process*. London: Cambridge University Press. (Republished by Dover Publications, New York (1965)).
Robinson, A. C., Collins, J. F. & Donachie, W. D. 1987 *Nature, Lond.* **328**, 766.
Rosenblatt, F. 1962 *Principles of neurodynamics*. New York: Spartan Books.
Rumelhart, D. E., McClelland, J. L. & the PDP Research Group 1986 *Parallel distributed processing: explorations in the micro-structure of cognition*, vols 1 and 2. Cambridge Massachusetts: Bradford Books.
Salem, J. & Wolfram, S. 1986 In *Theory and applications of cellular automata*, (ed. S. Wolfram), pp. 362. World Scientific: Singapore.
Sellers, P. H. 1980 *J. Algorithms* **1**, 359–373.
Smith, T. F. & Waterman, M. S. 1981 *J. molec. Biol.* **147**, 195–197.
Stauffer, D. 1979 *Phys. Rep.* C **54**, 1–74.
Tank, D. W. & Hopfield, J. J. 1985 AT&T Bell Labs preprint.
Wall, C. E. 1986 Numerical investigation of hyperscaling and real space renormalisation group transformations in the three-dimensional Ising model. Ph.D. thesis, University of Edinburgh.
Widrow, B. & Hoff, M. E. 1960 *IRE WESCON Conv Record* part 4, page 96.
Wilson, G. V. (ed.) 1987 *Edinburgh Concurrent Supercomputer Newsletters*, 1 2 & 3.
Wilson, G. V. & Pawley, G. S. 1987 *Biol. Cybernet.* (In the press.)
Wilson, K. G. & Kogut, J. 1974 *Phys. Lett.* C **12**, 75–200.
Wolfram, S. 1986 *J. statist. Phys.* **45** 471–526.

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES



IGURE 3. Cellular automaton simulation on the Computing Surface (Kenway, McComb & Wylie, unpublished results) illustrating injection from a pipe into a transverse flow.

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES



GURE 8. Three NMR datasets showing the same cross section of the human thorax. The top row are smoothed images after thresholding. The bottom layer show the results of applying the Zucker–Hummel surface operator to the top row.

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES