A pedestrian's introduction to spacetime crystallography

T. Toffoli

Ordinary crystallography deals with regular, discrete, static arrangements in space. Of course, dynamic considerations—and thus the additional dimension of time—must be introduced when one studies the origin of crystals (since they are emergent structures) and their physical properties such as conductivity and compressibility. The space and time of the dynamics in which the crystal is embedded are assumed to be those of ordinary continuous mechanics. In this paper, we take as the starting point a ***spacetime crystal****, that is, the spacetime structure underlying a discrete and regular dynamics. A dynamics of this kind can be viewed as a "crystalline computer." After considering transformations that leave this structure invariant, we turn to the possible states of this crystal, that is, the discrete spacetime histories that can take place in it and how they transform under different crystal transformations. This introduction to spacetime crystallography provides the rationale for making certain definitions and addressing specific issues; presents the novel features of this approach to crystallography by analogy and by contrast with conventional crystallography; and raises issues that have no counterpart there.*

1. Introduction

Traditional crystallography enumerates and classifies the regular arrangements of motifs in Euclidean space, where by *regular* one means invariant with respect to a discrete group of spatial isometries that contains as a subgroup the Abelian group freely generated by three independent translations, and by *motif* one means an arbitrary geometrical figure in that space.

We recall that, starting with the 14 Bravais lattices and keeping one point of the lattice fixed, one obtains the 32 *point groups* [1]. If the latter are combined with translations, one obtains the 230 *space groups* (ascertained in 1891). The corresponding regular arrangements are exhibited in all their technical glory in crystallography manuals such as [1–3], and justified in a more abstract mathematical fashion in tracts such as [4].

Besides purely geometrical data such as atom positions, the motif may be augmentedby features that are not strictly geometrical, such as charge, polarization, and magnetic susceptibility, insofar as they are imagined to be acted upon in some definite way by those isometry transformations. (*What is the mirror image of the south pole of a magnet—north or south?—and why?*) Here the boundary that divides an *a priori* deductive exercise from an inductive, experimental discipline becomes uncertain and full of surprises, as shown by Altman's admirable essay [5]. In this sense, crystal classification is still a somewhat open-ended enterprise.[1]

Though started as a specialized, stamp-collecting-like discipline useful for organizing a collection of minerals, crystallography eventually matured into a versatile tool of scientific analysis, useful for inferring the size of atoms, the shape of molecules, and the nature of chemical

Note: This paper is based on a talk presented in May 2003 at a symposium at the IBM Thomas J. Watson Research Center in Yorktown Heights, New York, honoring Charles Bennett on the occasion of his sixtieth birthday.

[1] We give an example of this when we cover cellular automata and lattice gases in Section 5.

0018-8646/04/$5.00 © 2004 IBM

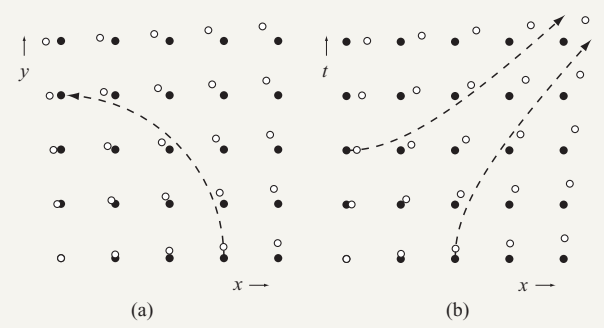


Figure 1

(a) Circular rotation of a spatial lattice (4° from the solid to the hollow lattice). If the rotation were continued for a quarter turn, as indicated by the dashed arrow, the transformed pattern would come to coincide with the original one. (b) Hyperbolic rotation of a space–time lattice with velocity parameter $\beta = 1/2$. As β increases, as indicated by the dashed arrows, the transformed pattern will never overlap the original one.

bonds. The deep emphasis on symmetry and covariant transformations that permeates modern physics [6] owes much to the successes of crystallography as a conceptual discipline.

Finally, the aspect that ultimately matters in crystallography is not a particular geometric lattice *per se*, but certain functions (such as charge distributions) defined on this lattice. Statistical mechanics and quantum mechanics take this approach even further insofar as the constructs with which they deal are linear combinations (with real coefficients for statistical distributions and complex coefficients for quantum states) of lattice functions. While the functions themselves (or these higher-order constructs) may not have any special properties *vis-à-vis* the lattice, nevertheless they may be expressible in terms of functions (e.g., eigenvectors of lattice transformations) that each have particular properties with respect to the lattice itself [7, 8]. In this way, the host lattice becomes a key to describing the guest functions in terms of coordinates that are most natural to the given context. In general, this is really all that one can hope for—and it is, as a matter of fact, quite a lot.

In this paper we ask four questions:

1. How is the concept of crystal naturally to be extended if one adds a time dimension to the dimensions of space? And what does it mean to add a "time" dimension?
2. What kinds of regularity (strict, approximate, or emergent) are relevant when the repeated motifs are no longer static tiles but dynamic function–composition constructs made out of signals and events, and thus the whole regular arrangement represents a kind of computer?
3. What functions—on this lattice computer explicitly "unfurled" in spacetime—are legitimate computational histories? How do these histories transform when the lattice coordinates are transformed? What properties of these histories are left invariant?
4. Does the uniformity group of the computational histories coincide with that of the underlying computational lattice? Or can the latter be augmented by new, emergent symmetries? In this context, certain tradeoffs between periodicity of structure and of function surprisingly follow from mere computability arguments.

We shall see that it makes sense to speak of spacetime crystallography even before introducing a metric (the one on which to base the isometries). In fact, the Minkowski metric itself does not have to be independently postulated, since it naturally emerges from computational lattices having a special discipline (essentially, having no more neighbors than the minimum required for a given number of dimensions). In this context, Lorentz invariance can be interpreted as an effective functional equivalence of different but related computational structures. By its more abstract lines of inquiry, this paper complements Norman Margolus's more phenomenological survey of crystalline computation [9].

2. A disclaimer

As we've seen, by *regular* one means *invariant with respect to a discrete subgroup of isometries of the underlying space*. This means that the issue of spacetime crystallography, formally speaking, revolves around which metric should be used when ordinary space is augmented by time. Now, the pre-relativistic Galilean metric keeps space and time in separate compartments; thus, an isometry of Galilean space–time is a combination of space and time operations that separately leave space and time distances unchanged.

But, aside from combinations of pure space isometries and pure time isometries (the latter consist just of translations and reflections), the only new isometries yielded by uniting space and time are glide reflections (with translation along time and reflection about a hyperplane parallel to time) and screw rotations (with advancement along time and rotation about a time axis). All counted, the family of Galilean crystals is but a trivial extension of that of Euclidean crystals.

On first sight, the Minkowski metric

$$ds^2 = dt^2 - (dx^2 + dy^2 + dz^2) \qquad (1)$$

may give one hope of further enlarging one's crystallographic "stamp collection." In fact, this metric yields a new class of isometries—the so-called *Lorentz boosts*—which perform hyperbolic (as contrasted to circular) rotations on the combined spacetime (**Figure 1**).

Unfortunately, except for a few reflections, the group of Lorentz boosts has no nontrivial discrete subgroups. If we keep rotating the lattice of Figure 1(a), it will come to coincide with the original lattice more than once for every turn; on the other hand, if we keep Lorentz-boosting the lattice of Figure 1(b), it will never come to coincide with the original lattice. Thus, we do not break any new ground with Minkowskian crystals.

From the viewpoint of strict isometries, then, spacetime crystallography is but a marginal annotation to ordinary crystallography. Beyond those already offered by the latter, new rewards from the subject must be sought in other directions.

3. So what is ordinary crystallography?

After a brief interlude with the outward shape of crystalline *objects* ("diamonds are forever"), classified by the point groups mentioned above, crystallography quickly moves on to the main theme, namely, the structure and properties of indefinitely extended crystalline *materials* (for example, "Quartz cleaves along certain planes and is piezoelectric when squeezed in certain directions"). How do the 230 space groups mentioned in the Introduction come about? Why that many and no more?

Consider a generic repetitive pattern such as that of **Figure 2(a)** (for simplicity, let us stick with two dimensions). Out of that pattern one can easily identify the lattice that gives the underlying two-dimensional spatial-repetition "beat" [**Figure 2(b)**] and the repeated motif itself [**Figure 2(c)**].

The central concept in all of this is the family of parallel displacements, or translations, that move the whole pattern into superposition with itself and thus leave it invariant. These translations form a group in the mathematical sense. In our case, it is a free Abelian group of two generators, illustrated in the form of a Cayley graph in **Figure 3**.

The quotient of Euclidean space with respect to this group (that is, what is left of Euclidean space if one identifies any two points that coincide up to a translation) is a finite, wrapped-around version of Euclidean space, namely, a two-dimensional torus. With this identification, the pattern turns into a single copy of the motif, "wrapped" on the torus.[2]

The above properties characterize a crystal. That is, a crystal may be defined as a collection of points, or pattern, in Euclidean n-space that is left invariant by a free Abelian group generated by n independent translations. For classical crystallography, this leaves the combinatorial puzzle of determining what other isometries besides those translations leave the pattern invariant as well.

[2] To form the torus, note that the right edge of Figure 2(c) will be "glued" to the left one in a straightforward way, while the top edge will be shifted leftward before being glued to the bottom edge, so that the two interrupted vertical strokes will match. Compare the lapping of the boxes in Figure 2(a).

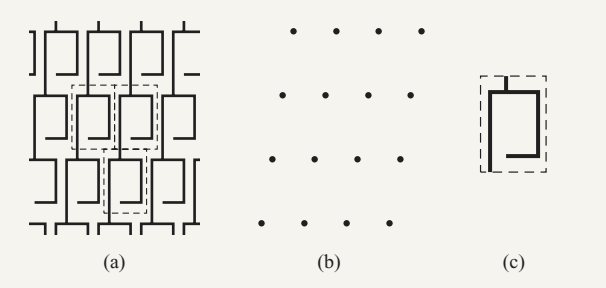


Figure 2

(a) Two-dimensional repetitive pattern. (b) The lattice that captures its two-dimensional spatial repetition period. (c) The repeated motif itself (enlarged). The origin and shape of the box that delimits the motif are arbitrary, provided that such a box tiles the plane with the periodicity of the lattice itself.

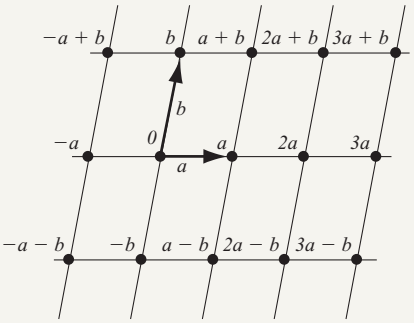


Figure 3

Cayley pattern of the translation group of a pattern like that of Figure 2. The a and b vectors denote the two generators; the dots, the elements of the group. (Lengths and angles are irrelevant to the Cayley graph itself, but have been chosen to suggest the correspondence with the original pattern.)

Since the pattern, as transformed by one of these additional isometries, must map into itself and thus keep obeying the original translation invariances, the task boils down to examining the automorphisms of the free Abelian group and determining which of them correspond to further isometries of the pattern (such as reflections, rotations, and glides). For the latter, one must seek the "greatest common symmetries" between the symmetries of the bare lattice[3] and those of the motif.

In a nutshell, a crystal is a pattern of points in Euclidean space that displays at the very least the symmetries of the parallelepiped (a parallelogram in

[3] These depend in a nontrivial way on the relative lengths and angles of the generators, which are classified in three dimensions by the 14 Bravais lattices and the seven crystal systems.

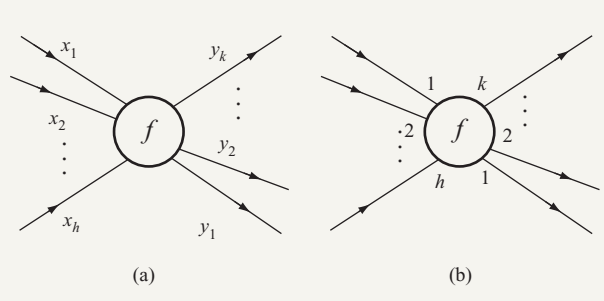


Figure 4

(a) Event representing the mapping of Equation (2). Note that input and output ports are intrinsically numbered according to the order of the input and output components of the mapping in Equation (2), as indicated in (b), irrespective of the names used for input and output variables.

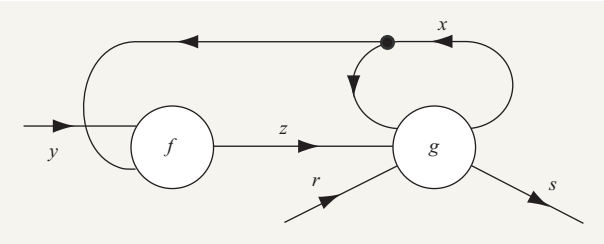


Figure 5

Function–composition graph corresponding to the system represented by Equation (3). Note that signals y and x cross over so as to reach node f as, respectively, first and second input, as specified by the first of the two equations. The dot represents a fan-out node, as explained in the text.

Figure 3) and may enjoy additional ones thanks to degeneracies of the lattice[4] seconded by symmetries in the motif itself.

In this paper we argue that, as a criterion for agreeing whether a transformation of a spatially extended pattern leads to "essentially the same pattern"—and thus as a defining criterion for "crystal"—the concept of *isometry* can be usefully replaced by one that is simpler and more general, namely, *isology*, or "equality of computational structure" (see the subsection on isologies in Section 5). From the latter, one may obtain distance—and thence isometry—as a derived, emergent concept in the context of distributed computation obeying a special discipline.

[4] For instance, two independent length parameters may happen to have identical values, or an angle may turn out to be a small multiple of a turn such as 60° or 90°.

4. Distributed computation

Before turning to crystalline computation, we devote a section to some prerequisite material on distributed computation.

Function–composition graphs

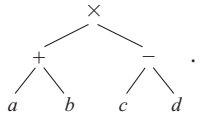
Computation is the exercise of function composition in a context in which building blocks and interconnection rules are specified once and for all, so that the originality of the construction lies not in the introduction of novel components but in obtaining the desired behavior by combining only the given ones—though in as large a number as desired.

As blocks to be interconnected, we consider functions of the form $f: X \to Y$ in which both the domain X and the codomain Y are finite Cartesian products of sets of the form $X = X_1 \times X_2 \times \cdots \times X_h$ and $Y = Y_1 \times Y_2 \times \cdots \times Y_k$, so that the mapping $x \overset{f}{\mapsto} y$ takes the form

$$(x_1, x_2, \cdots, x_h) \overset{f}{\mapsto} (y_1, y_2, \cdots, y_k). \tag{2}$$

That is, the result of f, as well as its argument, will be a finite ordered collection of variables (a "tuple") rather than a single variable. Though the component sets themselves (X_1, X_2, etc.) may be arbitrary, in what follows we are interested only in finite sets.

When using functions whose output is a single variable, function composition yields a tree that can easily be linearized, as when one writes

$(a + b) \times (c - d)$ for 

.

On the other hand, when functions have more than one output, function composition yields a more general function–composition graph, that is, a directed graph (compare Figure 5) in which a variable is associated with each *arc*, or *signal*, and a function is associated with each *node*, or *event*.[5]

Thus, a mapping such as Equation (2) is indicated in graphic form as in **Figure 4**, and an entire function–composition scheme as in **Figure 5**. Note that, besides the usual graph-theoretic understandings, the signals that impinge on an event are assumed to be indexed according to their position in the input or output tuple of the corresponding mapping, so it makes sense to speak of, say, "input signal number 2" irrespective of the name of the variable associated with it, as indicated in Figure 4(b). Because of this, signals must be appropriately routed so as to connect the right signal not only to the right node but also to the right input or output slot of the

[5] In graph-theoretic parlance, the function–composition graph is *labeled* by functions and *colored* by variables.

node (to this end, note, for instance, the crossover of the x and y signals in Figure 5).

A mapping such as (2) can be thought of as an equation relating its input and output variables. The system of simultaneous equations

$$\begin{cases} (y, x) \overset{f}{\mapsto} z \\ (x, z, r) \overset{g}{\mapsto} (s, x) \end{cases} \tag{3}$$

yields the function–composition graph of Figure 5.

Fan-out

Note the appearance in Figure 5 of a fan-out node for x, denoted by a dot. This is an instance of an n-fan-out function of the form $X \overset{I_n}{\to} X^n$, which takes x as an argument and returns n copies of x as a result:

$$x \overset{I_n}{\mapsto} \underbrace{(x, \cdots, x)}_{n}. \tag{4}$$

The fan-out node is not specified in (3) by an explicit equation, but arises from the fact that x appears there as an argument in two places. A more explicit (and physically more realistic) way of representing this situation is to treat the branches of the fan-out node as two new variables, u and v, and add to (3) a third equation relating u and v to x,

$$\begin{cases} (u, y) \overset{f}{\mapsto} z \\ (v, z, r) \overset{g}{\mapsto} (x, s) \\ x \overset{I_2}{\mapsto} (u, v) \end{cases}. \tag{2′}$$

By using convention (2′) for (3), one obtains a one-to-one function–composition scheme [10] in which no more than one input variable is identified with any given output variable (intuitively, any branching of signals can take place only at an actual node). Accordingly, the fan-out node in Figure 5 may be rewritten more explicitly as 

, reflecting the third equation of (3). Whether such a convention is used tacitly or explicitly, it is clear that, without loss of generality, any function–composition scheme can be assumed to be one-to-one. A consistent reminder of this convention is given by the number of arrowheads—one for each variable.[6] In what follows, we always treat fan-out explicitly as a node.

[6] For example, in Figure 5 there are three arrowheads around the fan-out node, indicating three distinct variables, even though their values are asked to coincide by the third equation of (2′).

Partial order and distributed dynamics

In (2), any of the output variables may be said to be *later than* (denoted by ">") any of the input variables. In a function–composition scheme, the meaning of *later* can be transitively extended: If there is an intermediate variable q such that $r > q$ and $q > p$, then $r > p$. If, because of the structure of the function–composition scheme, the relation *later*, besides being transitive, also happens to be antisymmetric (intuitively, if the future, unlike Figure 5, does not feed back into the past), then this relation establishes what is called a *partial order* [11] among the variables.[7] In graph-theoretical terms, signal q is later than p if they are distinct and there is a directed path in the graph beginning with p and ending with q. The relation "later than" is a partial order when the directed graph is acyclic, that is, has no loops. We call such an acyclic function–composition graph a *causal network*.

The partial order discussed here naturally extends to nodes; a node is later than its input signals and earlier than its output signals. Thus, in Figure 5, $x > g > r$.

What we want to stress here is that, just as the total (or linear) order $\cdots, q_{t-1}, q_t, q_{t+1}, \cdots$ established among the variables q_i by the recurrence relation

$$q_{t+1} = \tau q_t \tag{5}$$

allows us to view this relation as the transition function of a dynamical system, so the partial order between signals of a causal network allows us to view this network as a dynamical system. In the first case we have a concentrated system (in which variables do not have an attribute or index denoting "place"). We may call a value of q_t the *state of the system* at time t; from this state, the transition function τ produces the next state, i.e., a value for q_{t+1}. An entire sequence of values for $\cdots, q_{t-1}, q_t, q_{t+1}, \cdots$ (as distinguished from the sequence of variables[8]) satisfying (5) is called a *history of the system*.

In the second case we have a distributed dynamical system governed by local interactions.[9] Though analogous, the picture here is somewhat more complex than for concentrated systems[10] and is discussed in some detail below.

In a distributed system we distinguish between *global state*—the state of the entire system through a given spacelike cut (see below)—and the state of one or more localized components of the system, which we may call *local state*.

[7] It is a matter of taste whether the relation that defines a partial order is required to be reflexive or irreflexive. We opt for the latter—that is, a variable is never later than itself.

[8] Here we are confronted with a common type of language ambiguity, in which the state of a system at time t' may mean either the variable q_t, as distinguished from, say, q_{t+1}, or a particular value (a constant chosen from its state set) assigned to that very variable in certain circumstances. We will be careful to resolve this ambiguity when the context does not provide sufficient clues.

[9] Though the system may extend indefinitely in some abstract "space," each node of the causal network receives inputs from only finitely many nodes.

[10] There is an obvious parallel between concentrated and distributed dynamical systems on one hand and ordinary and partial differential equations on the other.

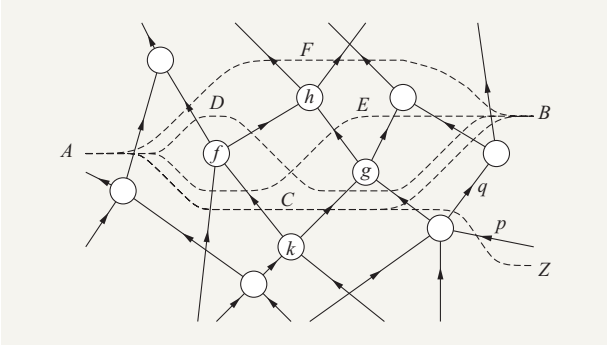


Figure 6

Causal network. The dashed line *ACB* is a spacelike cross section through the system, and the collection of signals intercepted by it forms a global cut for the system; the collection of their values is, correspondingly, a global state; similarly for *ADB*, *AEB*, and *AFB*. On the other hand, the dashed line *ACZ* is not a spacelike cross section because it intersects two signals, p and q, of which one is later than the other.

If the system is viewed as synchronous, as the cellular automaton specified by the recurrence relation

$$q_{t+1;x} = f(q_{t;x-1}, q_{t;x}, q_{t;x+1}) \tag{6}$$

and expressed in graphic form [see Figure 10(a) in Section 5], one can still speak of "the system's global state at time t,"

$$\langle \cdots, q_{t;x-1}, q_{t;x}, q_{t;x+1}, \cdots \rangle, \tag{7}$$

i.e., the collective state of the entire collection of variables whose first index has value t. This collection itself will be called the (spacelike) *cut* of the system at time t. Intuitively, a cut is the act of taking a cross section through the network at time t, and the global state is the result of examining this cross section (cf. Footnote 7). In a similar way one can speak of the *next* global state at time $t + 1$, etc. In other words, though a cellular automaton is a distributed system, it is viewed as operating in Galilean space–time (see Section 2), where space and time can be treated separately and its global states are still totally ordered.

In general, however, a distributed dynamics may not have a global (or synchronous) time naturally associated with it, and the system may better be viewed as *concurrent*,[11] as in the example of **Figure 6**. There it is inessential—for the orderly generation by some computing device or algorithm of signals obeying the indicated functional relationship—whether event f is evaluated before or after g in terms of some absolute clock. All that matters is that both be evaluated earlier than event h to provide the latter with the required input signals, and later than event k to receive from it the required output signals. It is in this sense that the partial order has a natural interpretation in terms of a concurrent dynamics. Note that in Figure 6 we no longer have any use for separate time and space axes such as those of Figure 10, shown later; the orientation in spacetime, that is, the direction of causal effects, is explicitly given for each signal.

Borrowing terminology from special relativity, two distinct signals are in a *timelike* relationship if one is later than the other, *spacelike* if not. (If two signals coincide, by convention they are both in a timelike and spacelike relationship.) A global cut is any maximal collection of mutually spacelike signals. The partial order existing between signals naturally induces a partial order between global cuts; i.e., global cut P is later than Q if they are distinct and any signal of P that is not identical with one of Q is later than some signal of Q. In Figure 6, the dashed line *ACB* (and similarly *ADB*, *AEB*, and *AFB*) is a global cut through the system, since all of the signals intercepted by it are mutually spacelike, and no more can be added; their collective state thus represents a global state for the system. On the other hand, the dashed line *ACZ* is not spacelike, since it intersects two signals, p and q, of which one is later than the other; the signals intercepted by it do not form a global cut. Global cuts *ADB* and *AEB* are both later than *ACB* and earlier than *AFB*, but neither is earlier or later than the other. As mentioned in Footnote 7, when no confusion is possible, we may say *global state* for *global cut*.

Figure 6 illustrates how, in spite of having access to multiple evolution paths, a concurrent dynamics always leads to the same functional dependency between a global state and a later one. State *ACB* may progress to *AFB* by first going through local transition function f (and thus passing through state *ADB*), or first going through g (and thus passing through *AEB*). What qualifies the dynamics as concurrent is that the two trajectories commute, as shown in the following diagram:

$$\begin{array}{ccc} ACB & \rightarrow & ADB \\ \downarrow & & \downarrow \\ AEB & \rightarrow & AFB \end{array} .$$

Intuitively, though state *AFB* can be reached from *ACB* in two different ways, the subsequent evolution of *AFB* is independent ("has no memory") of which trajectory had been followed before.

If signal q is in a timelike relationship with p, then it belongs to its *light-cone*. Using standard terminology from lattice theory, any signal q that is later than a signal p is an upper bound for it. The collection of all upper bounds for p (the principal upper ideal generated by p) may be termed the forward part of the light-cone of p or its

[11] This term is preferred to *asynchronous*, as the latter is often associated with nondeterministic behavior.

fore-cone (see Chapter 9 in [12]), and similarly for the principal lower ideal, or *past-cone*, of p. One can similarly speak of upper and lower bounds of a set P of signals, and thus of the fore-cone and past-cone of P.

5. Crystalline computation

We may now proceed with our approach to crystallography founded on concurrent computational dynamics. Our goal is to characterize an autonomous, indefinitely extended, uniform computational structure that can play the role of a crystalline computational medium.

The "digital physics" hypothesis

Regardless of whether our physical world is at bottom a digital computer,[12] certain features of physics that in textbooks are given as postulates can instead be derived as natural consequences of intuitively simpler premises, e.g., by assuming that the observed continuous dynamics is an emergent one, having as a microscopic substrate a digital computational process.[13]

In the context of this assumption, to match the evident spatial and temporal *uniformity*—at least on a tangible scale—of physical law, one may presume that the underlying circuit itself has a regular structure. Alternatively, one may imagine that the circuitry itself is random, but with statistical properties that are uniform on that scale; uniformity would then be only an emergent feature. To reconcile the *discreteness* in space, time, and state of the underlying digital circuitry with the apparent continuity, on a tangible scale, of physical law, one may demand that circuit features—wires and gates—be much finer than that scale, so that continuity in this respect would again be an emergent feature.[14] Of course, in the case of random circuitry and emergent uniformity, circuit elements would have to be even smaller, to leave room for the necessary statistical averaging.

Our arguments in this paper should not be construed as an attempt to garner evidence for some form of "digital physics" hypothesis. It is one thing to maintain that the mathematical form of a certain aspect of physics has "emergent" painted all over it, and it is another thing to suggest *what* precise microscopic combinatorial dynamics it could emerge from. For an analogy, think of the central limit theorem. Almost any distribution, if iterated (i.e., convolved with itself) a large number of times, yields in the limit the normal distribution. Therefore, whenever we experimentally encounter a normal distribution, we may legitimately conjecture that (a) this distribution is not primary, but is emergent with respect to some underlying microscopic distribution. However, the very reasons that make this conjecture extremely likely make it extremely unlikely that (b) any specific microscopic phenomenon will be the one actually underlying the observed macroscopic behavior. In other words, in arguments of this kind, certainty about (a) is bought at the expense of a proportionate uncertainty about (b). For example, when we show below that Lorentz invariance naturally emerges from a simple computerlike substrate, all we are doing is making a plea for the extreme genericness of Lorentz invariance rather than suggesting an explanatory mechanism for it.

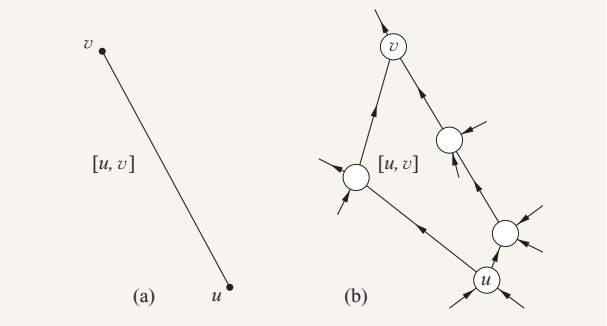


Figure 7

(a) The interval $[u, v]$ in Euclidean space is the "stretch of space" between u and v. (b) The interval $[u, v]$ in a causal network is the "stretch of hardware" later than u and earlier than v.

Isologies, or computational isomorphisms

A basic concept in ordinary crystallography is that of the *distance* $r(P, Q)$ between two points P and Q. An *isometry* is a mapping σ from pattern A to pattern B that preserves distances, i.e., such that, for any two points P, Q of A,

$$r[\sigma(P), \sigma(Q)] = r(P, Q) . \tag{8}$$

We would like to introduce analogous, but more general, concepts useful for showing that two causal networks (rather than just two patterns of points) are essentially the same.

The portion of network consisting of all of the events w between u and v (that is, $v > w > u$) and all of the signals incident to them is called the (*closed*) *interval* between u and v, denoted $[u, v]$ [11]. **Figure 7** contrasts (a) the segment (or collection of points) $[u, v]$ between

[12] A scramble to establish priority for this speculation [13–15] would be a bit silly, not only because there is as yet no direct evidence for it, but also because it is one of the most obvious conjectures that may come to one's mind [16, 17] (and, as a referee pointed out, it is also a rather vague one).
[13] For example, concurrent computation naturally suggests, as we've seen in Figure 6, that space and time may be coupled as an indivisible whole, as in Einsteinian relativity, rather than just as the Cartesian product of two independent entities [Figure 10(a)], as in Galilean and Newtonian mechanics.
[14] As an alternative, if the circuit is a regular lattice, one may imagine that, even if the lattice spacing is coarse, the state of the system may be a linear superposition of lattice states, so that by continuous interpolation between two discrete lattice states one may obtain intermediate states spanning a continuous group of translations [18], just as a continuum of spin states may be obtained, in quantum mechanics, from the linear superposition of just two states.

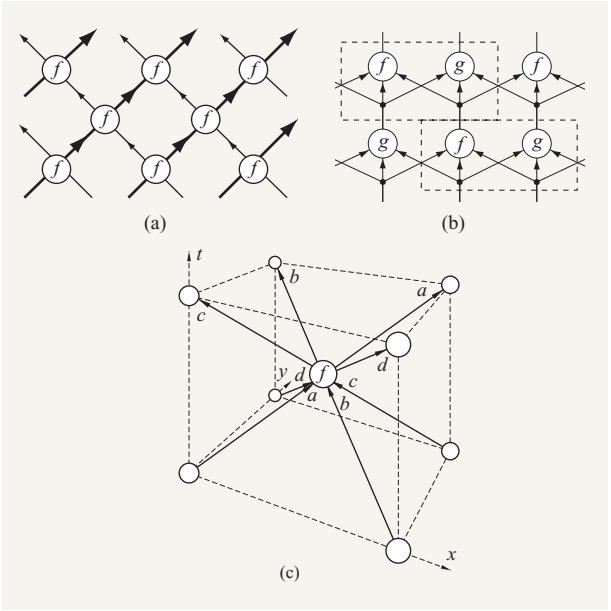


Figure 8

Examples of computational crystals: (a) Lattice gas having a very simple structure. (b) One-dimensional cellular automaton with a two-node primitive unit cell. (c) Highly symmetric two-dimensional lattice gas; the two-node unit cell shown here is not primitive.

two points u and v in Euclidean space with (b) the interval (or collection of events and associated signals) $[u, v]$ between two events in a causal network. If we imagine the length of a segment to represent, as it were, the "amount of space" between its endpoints, then $d(u, v) = d(u', v')$ if there is an equivalent amount of space between u and v and between u' and v'. In a similar way, we say that interval $[u, v]$ is *equivalent to* $[u', v']$, or $[u, v] \equiv [u', v']$ if there is an isomorphism in terms of function–composition structure (events and signals incident to them) from one to the other, i.e., there is an equivalent stretch of computational machinery between u and v and between u' and v'.

Finally, corresponding to the concept of isometry, we define an *isologic transformation* of a causal network, or *isology*, to be a mapping of the network to itself that takes closed intervals into equivalent closed intervals; this means that, up to a relabeling of signals and events, the two networks correspond to identical function–composition schemes and represent, in that sense, identical computers.

Isologic translations

In an isometry, the images P', Q' of points P, Q that are near to one another are, by construction, themselves near to one another. However, to go to P', point P may move a different distance than Q to Q'. In a reflection, for instance, a point moves by an amount proportional to its distance from the plane of reflection. A translation (*parallel transport*) is a special kind of isometry in which all points move by the same amount. A set of n translations are *independent* if none can be expressed as a linear combination of the remaining $n - 1$.

The same term, *translation*, may be used for an isology in which every event moves by the same local recipe; if, say, the way to go from event p to p' is by following the hth output signal of the starting node to some intermediate node, and then the kth output signal of this node to the destination node, the same directions must be employed to go from q to q'. In the case of isologies, n translations are independent if none can be expressed as a linear combination with integer coefficients of the remaining $n - 1$.

Invertible causal networks

A causal network is *invertible* if all of the mappings from global states to global states induced by its function–composition structure are invertible. A causal network turns out to be invertible if all of its nodes are invertible functions. This is proved by an argument similar to the one that connects an extremum principle to an Euler–Legendre equation [19], or global minima to local minima in dynamic programming. That is, one can go from one space cut to a later space cut by making the space cut *go over* (from "before" to "after") one node at a time; thence the necessity and the sufficiency that each single node be invertible. Therefore, in causal networks, structural invertibility, determined by inspection of local hardware, and functional invertibility, which is a relation between global states, go hand in hand. However, see Section 6.

Crystalline computers

We are now coming to the conclusion of our "reasoned definitions" journey. A causal network is *closed* if the underlying partial order has no maximal or minimal element. In a closed network, there are no "dangling" signals; every output of an event is an input to another event, and vice versa. A causal network is a *causal lattice* if every subset of signals has an upper bound and a lower bound.[15] Intuitively, given any subset of signals, there is a signal that is later than all of them and one that is earlier than all of them. Thus, in a causal lattice, any two (or more) signals, even if widely separated, will eventually share some history in the remote future and did share some history in the remote past.

A *crystalline causal network*—or, depending on the viewpoint, a *crystalline computer* or a *computational crystal*—is a closed causal lattice M such that

[15] In lattice theory, this property is one of several kinds of *completeness* [11].

1. M admits of a free Abelian group of isometries, G, generated by $d + 1$ independent translations.
2. The quotient of M with respect to G, M/G, is a finite function–composition network.

We call d the number of *spatial dimensions* of M, and M/G its *unit cell*. As in the case of ordinary crystals, the group G, and consequently the unit cell M/G, are not unique. A primitive unit cell is one that is "as small as possible," i.e., no proper subset of it is itself a unit cell. Also, primitive unit cells are not unique.

Figure 8 shows some computational crystals; **Figure 9** gives some counterexamples.

Cellular automata and lattice gases

Tradition reserves a special role to two kinds of crystalline computers, namely cellular automata and lattice gases, both having an enormous literature—see representative references in [20]; also, see [13] for an astounding amount of high-quality material, unfortunately not cross-referenced in a standard way to the extant literature.

Cellular automata have been given countless definitions, all more or less equivalent, and we do not try to give the perfect one here. All that matters is that they are viewed as synchronous function–composition schemes, as in (7), and use fan-out junctions to distribute the output of each designated transition function event at time t to the events that depend on it at time $t + 1$, as illustrated in **Figure 10(a)**.

Lattice gases have been extensively used as models reduced to the bare essentials of various aspects of physics. Lattice-gas fluid dynamics [21] has become a veritable industry. Though there is practical agreement over basic concepts, precise definitions are rare, *ad hoc*, or contradictory. We suggest a simple definition that in our opinion captures the essential nature of a lattice gas: *A lattice gas is a crystalline computer for every node of which the input set and the output set are of equal cardinality*. Thus, even though a lattice gas need not be invertible (that is, conserve information), it must formally conserve "amount of state," that is, number of states in going from left to right of the clause of the form $g: X \rightarrow Y$ that accompanies the definition of the transition function g of any node [**Figure 10(b)**]. Equivalently, even though the number of input signals and output signals impinging on an event need not be the same, the product of the numbers of available states of all input signals must equal that of the numbers of available states of all output signals. A corollary of this is that a lattice gas must do without fan-out nodes.

Space and time tradeoffs for the unit cell

Figure 8(c) illustrates a very versatile type of crystalline computing structure used, for instance, in the *Hardy–de Pazzis–Pomeau* (*HPP*) model of hydrodynamics [22] and in Ising spin models [23]. It is also used for the delightful *Critters rule* introduced by Norman Margolus [9]. We use this rule to illustrate how the type of crystal (that is, which space group, or, in our case, spacetime group, it belongs to) can be affected by how much of the dynamics the isometries are allowed to "know."

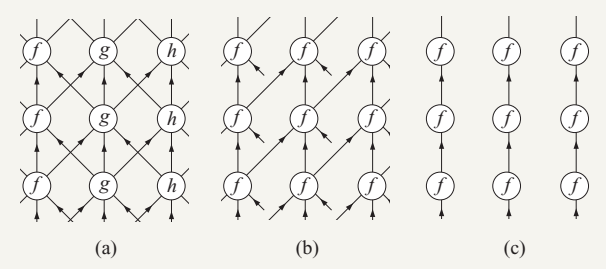


Figure 9

Counterexamples to Figure 8. Causal network (a) is not a crystalline computer because its primitive unit cell is infinite! (In fact, G is the one-generator Abelian group that shifts the network one row up, and M/Q is a horizontal strip containing an infinity of different nodes, $\cdots f\ g\ h \cdots$.) In (b), the network is not closed, because it has external inputs. In (c), the network is not a lattice, because there are signals that have no common past or future history.

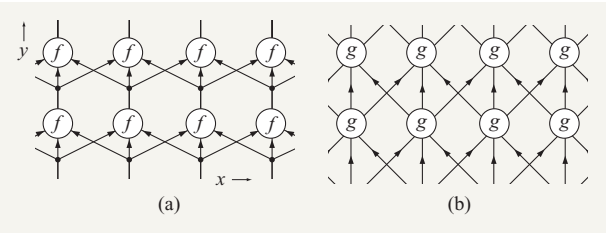


Figure 10

(a) Cellular automaton; note the systematic use of fan-out junctions. (b) Lattice gas; the "amount of state" that enters and leaves an event is the same.

Let each arc be a binary signal, with states 1 (particle) and 0 (hole). The Critters rule is best described by the following progression. Note that at each of the three stages we have an invertible rule:

- Stage 1. No interactions are turned on. Particles in the four channels of Figure 8(c) go straight through a node without seeing one another.
- Stage 2. The HPP interaction is turned on. Two particles colliding head on (i.e., coming from two opposite input channels) scatter off at 90° from the original directions and come out through the other two

Figure 11

The compressibility of quartz as a function of spatial direction. Reprinted with permission from [24]. © 1977 Elizabeth Wood.

channels. This is the mechanism that, in the long run, equalizes pressure in all directions.

- Stage 3 (the Critters plug-in). We turn on one more kind of interaction. First, if a third particle in addition to the two of Stage 2 converges on a node, it will bounce back instead of going straight through the node. Second, on leaving the node, all 0s turn into 1s and vice versa (this, at least, was Margolus's original formulation).

Note that the rule as given above is the same at every event (the primitive unit cell consists of a single event). On the other hand, the state of the system flashes back and forth; at every step, areas with a majority of 0s turn into a majority of 1s and vice versa. Particles are not conserved on an individual step basis.

An alternative formulation of the rule replaces Stage 3 with the following:

- Stage 3′. (The exchange between 1s and 0s on exiting a node is not performed.) On even steps (imagine spacetime to be checkered into even and odd events), the middle of three particles bounces back as in the original Stage 3. However, on odd steps it is not the middle one of three particles that bounces back, but the middle one of three *holes*. In other words, particles and holes take turns at interpreting the *bounce back* directive.

The two versions of the rule commute and are, in this sense, essentially isomorphic. However, the primed variant has the advantage that it conserves particles at every step, just as Stages 1 and 2 do, and so has a smoother and more natural visual evolution. The price paid for this is that the original version has a straight "cubic"[16] crystal structure with a primitive unit cell consisting of one node, while the variant has a "body-centered cubic" structure with a primitive unit cell consisting of two nodes. Intuitively, the variant has weakened a time-translation symmetry into a glide-reflection symmetry in which the glide is actually along the time direction and the reflection is not in space, but within an internal "isospin-like" space, that is, the finite state set consisting of just *particle* and *hole*.

Emergent properties

From the crystallographic group of an ordinary crystal one can derive some general structural constraints that its macroscopic properties must obey. Essentially, these properties must be at least as symmetric as the unit cell, and possibly more so. Thus, for example, in a crystal where no two axes are equivalent, the parametric surface representing thermal expansion in different spatial directions can be as "bad" as a triaxial ellipsoid, but no worse. In quartz, which belongs to the trigonal group, the parametric surface representing compressibility must be invariant for 120° rotations about the main axis, but to predict whether this surface will remain unchanged upon inversion of the axis (it doesn't, as shown in **Figure 11** [24]), one must know more about the atomic arrangement in the unit cell.

In an analogous way, one may wonder about dynamical properties within a spacetime crystal. I cannot forget the sense of elation I felt when I discovered not only that a certain cellular automaton I was playing with—the *Toffoli–Margolus gas*, or *TM* [25]—supported well-formed density waves,[17] but also that these waves propagated at the same speed in all directions (rotational invariance!) even though the cellular automaton was built on a square lattice. Unfortunately, for both HPP and TM gases, it turned out that while the velocity was direction-independent, the viscosity (and thus the wave attenuation) was not. It was left to Frisch and collaborators (see [26] and references therein) to determine what kinds of spacetime lattices in two and three dimensions would guarantee rotational invariance for all of the tensors relevant to low-Reynolds-number fluid dynamics.

Crystalline computation, invertibility, and computability

The intimate connections among crystalline computation, invertibility, and computability have been discussed in [20, 27] and, more recently, [28–30]. Here we give a brief summary.

Even though cellular automata are causal networks, for historical reasons an *invertible cellular automaton* is defined as one for which bijections are only required

[16] "Cubic" in the sense of two dimensions in space and one in time.
[17] I had stumbled on lattice-gas hydrodynamics independently of [22].

between "public" global cuts, disregarding the ancillary states temporarily created by the fan-out nodes for "internal distribution" purposes.[18]

This concept of invertibility differs from the one we defined in the section on invertible causal networks above; we may say that an invertible cellular automaton is functionally invertible in a global sense, but it is not obvious whether it is also structurally invertible in a local sense as a causal network. An interesting question is whether an invertible cellular automaton can be rewritten isomorphically as a standard causal network with no fan-out, namely, as an invertible lattice gas. In 1990, Toffoli and Margolus [20] conjectured that this is the case, and the conjecture was proved ten years later by Durand-Lose [29]. But even though the existence of such a lattice gas is thus affirmed, there is, in general, no effective way to construct it!

What is of interest is the nature of the difficulty. It turns out that, in contrast to ordinary physics, the backward light-cone of an invertible cellular automaton may have a different shape and width than the forward cone, and this width is, in general, not effectively computable on the basis of the forward dynamics. As a consequence (via a relatively recent theorem by Kari [28]), the lattice-gas crystal that isomorphically reproduces the behavior of an invertible cellular automaton may have a much coarser structure (in terms of primitive unit cell size) than the cellular automaton itself—and no effective bound is, in general, available for the coarseness ratio. We may thus have the paradox of an invertible crystalline computer whose functional behavior can be described in terms of a very small primitive unit cell, but whose physical implementation, surprisingly, may require an (unboundedly) larger primitive unit cell! In sum, structure and function may make very different demands on the fineness of the spacetime crystalline structure.

6. Lorentz transformations and relativity

The *Lorentz transformation* is a mapping of spacetime taken for the moment to be a (1 + 1)-dimensional Euclidean space indexed by Cartesian coordinates x and t, into itself. Using appropriate units for x and t, the family of Lorentz transformations is given by

$$\begin{cases} x' = \dfrac{x - \beta t}{\sqrt{1-\beta^2}} \\ t' = \dfrac{t - \beta x}{\sqrt{1-\beta^2}} \end{cases} . \tag{9}$$

[18] More precisely, the global cuts that are retained are those which intersect signals only immediately after a designated transition function node and immediately before the corresponding fan-out node. The ancillary signals that come immediately after a fan-out junction and immediately before a transition function event are never taken into account when invertibility is an issue [30].

The so-called *velocity parameter* β may be interpreted as the velocity of the *primed* frame with respect to the *rest* frame.

A Lorentz transformation becomes an isometry if the Euclidean distance r between points $P_0 = \langle t_0, x_0 \rangle$ and $P_1 = \langle t_1, x_1 \rangle$, which satisfies

$$[r(P_0, P_1)]^2 = (t_1 - t_0)^2 + (x_1 - x_0)^2,$$

is replaced by the *pseudo-Euclidean* distance s, which satisfies

$$[s(P_0, P_1)]^2 = (t_1 - t_0)^2 - (x_1 - x_0)^2$$

(note that a plus sign has been turned into a minus sign). As defined so far, a Lorentz transformation merely involves kinematic quantities, i.e., points in space and time. In addition to transforming space and time coordinates, it automatically induces transformations on all kinematic quantities, such as velocity, acceleration, frequency, spatial density, and spatial frequency (wave number); however, the corresponding transformations of dynamic quantities—such as mass, energy, pressure, charge, and electric field—associated with those points in spacetime are not automatically prescribed. An extension of the Lorentz transformation to those quantities is not a matter of pure logic but must be based at least in part on experiment [5].

The nontrivial aspect of the matter is that indeed there exists an extension of the Lorentz transformation to dynamic quantities, and this extension yields, in addition to the isometries mentioned above, invariance of *all* physical laws under the transformation. As a matter of fact, this set of correspondence rules, called *special relativity*, is valid only in the limit of very low matter/energy density, i.e., almost empty space. To deal satisfactorily with the general case, one must have recourse to a more complex, nonlinear theory, namely, general relativity.

All of this may sound quite arbitrary, something that perhaps has to be learned and accepted (after all, that is the way the world works), but in which there is nothing really to be understood. On the contrary, we would like to show that relativity makes a lot of intuitive sense if imagined to emerge from a crystalline computer substrate.

A simple substrate

Perhaps the simplest kind of crystalline computer is a lattice gas having the structure of Figure 8(a). The primitive unit cell consists of a single event with two inputs and two outputs. As suggested by the different line thicknesses, the state set of the right-going signals may differ from that of the left-going signals. In what follows, we graphically simplify the picture of this crystal by using dots instead of circles for events and the same line thickness for all signals. Also, even though the causal

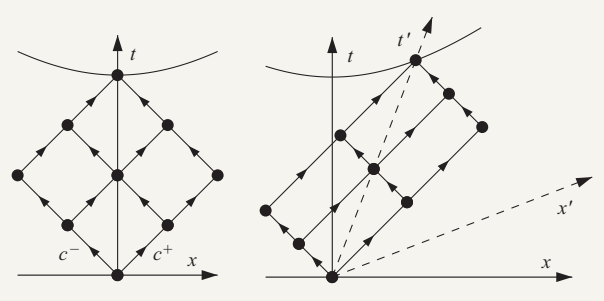


Figure 12

Action of a Lorentz boost on a crystalline computer. This transformation stretches the crystal by a factor of $\alpha = 3/2$ in the c^+ direction and compresses it by the same factor in the c^- direction.

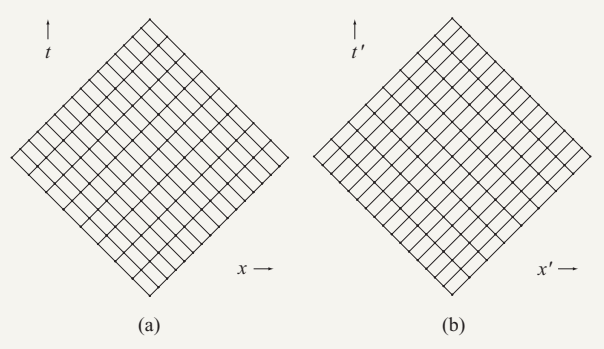


Figure 13

Two views of the computational substrate of Figure 8(a). Alice (a) is traveling at a velocity of 1/3, and Bob (b) at $-1/3$. They see materials having different textures. (Even though the geometry of the two lattices happens to be identical up to a reflection about a vertical axis, the function at each node is not.)

network extends indefinitely, we draw only a small portion of it.

Figure 12 depicts a Lorentz transformation of the crystal. We assume that the transformation affects the position of nodes and arcs thought of as inhabiting spacetime, but not their intrinsic attributes, i.e., the state sets and functions attached to them as in (2).

Relations (9) become less forbidding when expressed in more natural coordinates (namely, in place of t and x, the generators c^+ and c^- of the crystal lattice). In those coordinates, a Lorentz transformation simply stretches the lattice by a certain factor α along the c^+ generator[19] and shrinks it by the same factor along c^-. A stretch of $\alpha = 2$ corresponds to $\beta = 2/3$. In Figure 12, we stretch by $\alpha = 3/2$, giving $\beta = 5/13$. The picture makes evident the hyperbolic nature of the Lorentz transformation; specifically, while spacetime shapes are altered, all spacetime volumes, including that of the unit cell of the crystal, remain unchanged.

Because these are merely lattice coordinate transformations and do not affect the graph–theoretical properties of the causal lattice, it is clear that the function–composition scheme itself remains unchanged. At a macroscopic level, however, one does not have access to the individual signals and their individual microscopic interactions at the nodes, but only to macroscopic averages of signal states, and thus one must try to formulate macroscopic dynamical laws that express how these averages evolve as well as how they Lorentz-transform. And, because the discrete lattice coordinates are no longer directly accessible, these dynamical laws and transformations must be expressed in terms of continuous spacetime coordinates.

For example, even though one cannot resolve a unit cell in a macroscopic view, nonetheless the spacetime density of events is left invariant by a Lorentz transformation; moreover, even though one may not be able to directly count the number of events contained in a certain portion of spacetime history, one may make use of the knowledge that this number is invariant.

To more clearly trace the connection between the underlying digital computation and the emerging continuous dynamics, we proceed gradually from the microscopic view to the macroscopic one. Given a crystalline computer that appears like Figure 8(a) in its rest frame, let us consider two copies of it, one Lorentz-boosted by $\alpha = \sqrt{2}$ and the other by $\alpha = 1/\sqrt{2}$. Intuitively, this corresponds to the same crystalline substrate being employed as a computer by two users, *Alice* and *Bob*, one moving at a speed of 1/3 with respect to it and the other at a speed of $-1/3$, each expressing time and space coordinates with respect to his or her own reference frames. Even though the crystalline substrate is one, to Alice and Bob it looks, respectively, like the left and right of **Figure 13** (arrows and dots must be imagined like those of Figure 12), and has different computational properties (for instance, certain left-traveling tokens may move at different speeds in the two frames); from a macroscopic viewpoint, Alice and Bob see two different materials.

Relativity

Let us now consider a specific computation taking place on a crystalline substrate, i.e., a particular solution of the recurrence relations associated with the function–composition scheme; this may also be thought of as a *trajectory*, or a *history*, of the system. The computation

[19] The direction of the c^+ generator in spacetime corresponds to the speed of light rightward, while that c^- corresponds to the speed of light leftward.

is given different descriptions by the two users if the variables involved in it (no matter whether microscopic or macroscopic) are indexed by their time and place of occurrence in each user's spacetime frame (rather than by some absolute labeling scheme).

Besides this difference in the way the two users depict the same computation, it is in the nature of a computer that, with different initial conditions, it will give different computations to the same user. A most important question in the present investigation—the "only" question, one may say with little risk of exaggeration—is the following:

- Let Alice see initial state p_0 evolve into final state p_1 under the action of a certain dynamics τ. To Bob, who differs from Alice by a Lorentz transformation λ, this looks[20] like a certain state q_0 evolving into q_1.
- Can we find a different initial state r_0 for Alice that will look to Bob just like p_0 looks to her, and that, in particular, will evolve into a final state r_1 that will look to him just like p_1 looks to her?

Intuitively, imagine the substrate to be a computer and the initial conditions to represent a "program" that makes that computer show a certain animation movie to Alice. Bob, being a Lorentz-scrambled version of viewer Alice, sees the movie all scrambled. Can we suitably reprogram the computer, that is, run a different program on it, so that it will let Bob, in spite of being Lorentz-scrambled, effectively see an unscrambled movie?

As one may expect, the answer to the above question is, in general, negative.[21] The two main stumbling blocks are as follows:

1. The system state set may be too poor to provide a counterimage, under a Lorentz transformation, to an arbitrary state; in the terms of the above question, to provide an appropriate r_0 for every q_0. Special relativity is, in this sense, a statement about the richness of dynamical states in physics.
2. The Lorentz-transformed viewer in general experiences a transformed, and thus different, dynamics, τ' (as explained in Figure 13). Even if we have found a state r_0 that looks like p_0 to Bob, nonetheless, while state r_0 evolves into r_1 under the dynamics τ of Alice—and the Lorentz-transformed state of the latter is p_1—state p_0 evolves for Bob into $\tau'(p_0)$, not $p_1 = \tau(p_0)$!

Both obstacles may be literally smoothed out by taking a macroscopic view and letting a lot of detail wash out by averaging and blurring. On that scale, the variables that have retained significance are usually continuous ones, offering plenty of room for state interpolation (cf. Point 1 above). By the same token, on a macroscopic scale computational power tends to become fungible between different computer architectures, so that only the raw amount of computing power matters. Because, as we have already remarked, the number of events or primitive interactions—and thus, raw computing power—is conserved by a Lorentz transformation, other distinctions between τ and τ' (cf. Point 2 above) become less important.

The fungibility approach is discussed at greater length in [31]. In the next section, we prove that, for crystalline computers satisfying a certain rather weak condition, Lorentz invariance is attained in the limit as the density of computational tokens goes to zero (in which case, of course, the computing medium is grossly underutilized).

Invariance by rerouting

Let us again take a lattice gas like that of Figure 8(a) and require that f satisfy the following condition:

Among all possible states for the right-going signal as well as for the left-going signal, there must be a distinguished state such that, when this state appears at one of the inputs, the state at the other input is propagated through unchanged.

If such a state, which we may call the *vacuum state*, exists, signals will travel freely through the "vacuum." What this means is that a *token* (any non-vacuum state) has no way to sense that it is going through a node unless the other channel carries a token as well. Even though all tokens travel at light speed all of the time, they have no way of knowing how much ground they cover, nor can they tell elapsed time, nor be deflected unless they undergo a collision with another token. In a vacuum, tokens are in a state of suspended animation, as it were.

In the figures that follow, a thin stroke denotes an arc carrying the vacuum state; a thick stroke denotes an arc carrying any token. Note that, with the vacuum-state condition, a fresh token cannot be created out of the vacuum. As for token destruction, if f were noninvertible, a token might be annihilated in a collision; merely to simplify the following discussion, we assume that f is invertible, and thus no tokens are destroyed. As a consequence of token conservation, thick lines proceed straight and uninterrupted regardless of the details of collisions, as in **Figure 14(a)**; the actual state along a thick line may of course change from one arc to the next. For our purposes, the only relevant consequence of the vacuum-state condition is that certain of the straight lines that make up the grid are singled out—once and for all, as a function of the initial state[22]—as token-carrying

[20] When we say that Alice "sees" a state p, we mean that p is the description she gives to a certain state. If this "looks" like q to Bob, that means that q is Bob's description of the same state.

[21] Indeed, what relativity does is celebrate the fact that this question has a positive answer for *our* physics.

[22] For instance, the collective state of the signals entering the diamond from below in the figure.

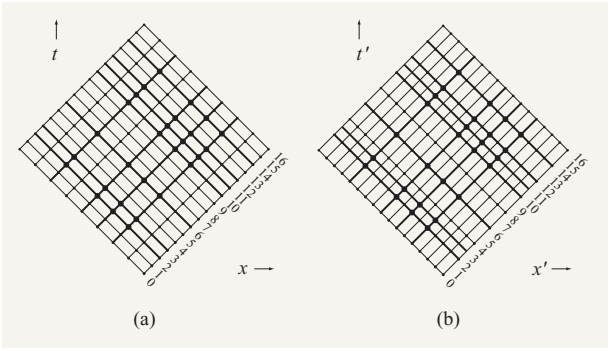


Figure 14

(a) A computation in a portion of a lattice gas having a distinguished vacuum state. The picture reports the existence of non-vacuum states or tokens (thick lines), but not their nature, which is irrelevant here. Tokens arrange themselves on straight, uninterrupted lines. The larger dots indicate interactions between tokens. (b) An attempt to fit the "same" computation in the Lorentz-transformed lattice of Figure 13(b). In this lattice, lines 3 and 11 cannot find host signals in the precise required positions.

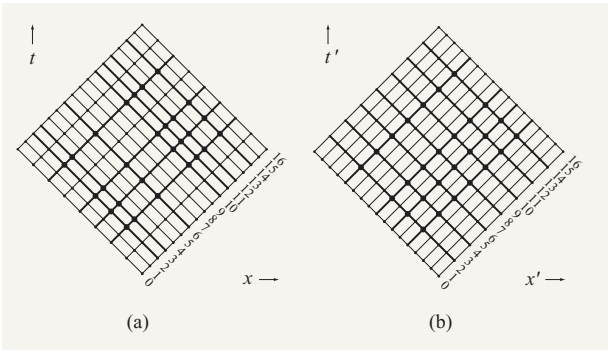


Figure 15

(a) The original computation, copied here for convenience. (b) The same logic evolution, but with small adjustments in the spacetime placement of certain signals made necessary by pitch differences between the original lattice and the Lorentz-boosted lattice. Signal 3 has been moved to line 4, and signals 10 and 11 respectively to 8 and 10.

lines, while the others are effectively suppressed. In a similar way, the nodes at the intersections of the thick grid lines are singled out as non-vacuum events (and are distinguished by a larger dot in the figure); all other nodes are effectively suppressed. We thus obtain a *reduced* function–composition graph which contains only thick lines and in which the distance between parallel lines has been normalized to one signal hop. This reduced graph conveys all of the functional dependency between states except for the original spacing between non-vacuum lines. The latter information is, in the present circumstances, decoupled from token-sequencing information and, from a macroscopic viewpoint, carries very little intelligence value—as we shall see in a moment.

Suppose now that Bob wants to observe a computation that must look to him just like what Alice sees from her viewpoint. What initial state do we have to set up for this purpose? We should lay the thick grid of Figure 14(a) as a template on top of that of Figure 13(b) and try to set Bob's initial signals to match the template. We attempt to do so in **Figure 14(b)**.

All of the right-going lines of the template find a matching line in Bob's grid, since the latter contains all of Alice's right-going lines and as many more, because of the Lorentz contraction (for lines of that direction, Bob's grid has twice as fine a pitch as Alice's). Unfortunately, because of the accompanying Lorentz "expansion" for the spacing of left-going lines, Bob's grid has half as many lines in that direction, and some of the template left-going lines (those marked "3" and "11") have no match at all in Bob's original grid. This is a case of Obstacle 1 in Section 6 under relativity.

Thus, we cannot always have an exact correspondence between a computation in Alice's grid and one in Bob's. However, as long as the lines that carry tokens are *very sparse*, the ones that carry a token and cannot be placed where the template would want them can, most of the time, be moved one position over to an available vacuum "slot." For instance, compared with **Figure 15(a)**, line 3 has been moved to position 4 in **Figure 15(b)**. Note that the functional dependency between all tokens has remained the same—only the spacing of some of the thick lines (but not their relative order) has changed.

Occasionally, both lines of Bob's grid that are closest to the position demanded by the template, such as lines 10 and 12 in Figure 14(b), are already token-carrying ones, and the idea of pushing 12 to 14 to make room is not viable because the latter as well is also already busy with tokens. But we can push signal 10 to line 8, making room for placing 11 in 10, and so forth. . . . The probability that longer-range adjustments will be needed goes down rapidly with range size.

Note that all we are doing, by pushing these "wrinkles" around until they find a "crack" to settle in, is slightly changing the spacetime position of certain data—but not the data themselves, their sequencing, or their functional dependency. And this positional error is not amplified as the computation proceeds—neither exponentially nor even linearly; it just remains constant and thus tends to vanish as one moves to a macroscopic view.

To sum up, we have exhibited a crystalline computing medium that is computation-universal[23] and exhibits Lorentz invariance in the limit as the density of tokens goes to zero. (This latter "fine-print" provision also applies, as we have seen, to ordinary special relativity.)

Emergence of the Minkowski metric

It should not come as a surprise that something like the Minkowski metric (1) emerges from a spacetime crystalline dynamics like that just discussed. Our message, again (see the digital physics discussion in Section 5), is not that there must be a lattice computer underneath relativity, but that the mathematical form of relativity makes the latter a prime candidate for an emergent feature.

In a lattice such as that shown in Figure 8(a), and using as units the dimensions of the unit cell, let us count all of the possible spacetime paths leading from the origin (0, 0) to a point (t, x), as in **Figure 16**. No matter what path is chosen, t represents the number of computation steps along the path and x the overall displacement of the result from the origin. Since

$$h = \frac{t + x}{2} \text{ and } k = \frac{t - x}{2},$$

the number of these paths is given by the binomial coefficient

$$N(t, x) = \binom{t}{\frac{t + x}{2} \quad \frac{t - x}{2}}.$$

Now, for $t \gg 0$ and $x \ll t$, that is, for a large number of computational steps and for a modest displacement from the origin, we have

$$t^2 - x^2 \approx t^2 \frac{\ln N}{t}. \tag{10}$$

The left-hand side of (10) is the squared *Minkowski distance* as in (1) (here we are considering a single spatial dimension). The right-hand side is t^2, that is, the squared *raw causal distance* (number of computational steps between origin and destination), which is independent of the displacement x, corrected by a factor that gives the efficiency of these computational steps in the spacetime direction from (0, 0) to (t, x). This efficiency is measured in terms of the average amount of choice at every step [31], that is, the log of the number, N, of possible trajectories divided by the number of steps t. (In the extreme case $x = t$, that is, for a "computation" that proceeds on a light-wave front, there can be no "side talk" between the different portions of the front; thus, a single trajectory is available at every point, the amount of choice is zero, and the computation grinds to a halt.) In other words, in the approximation considered here, (10) can be seen as a *prosthapheresis* formula—a device in which a product is replaced by a difference. The right-hand side of (10) appears to be intuitively more meaningful, while the left-hand side is computationally more convenient.

[23] The reduced graph is certainly capable of being computation-universal, since it is a lattice gas which has an arbitrary number of states (even after subtracting the vacuum state), and for which the transition function f can be chosen arbitrarily.

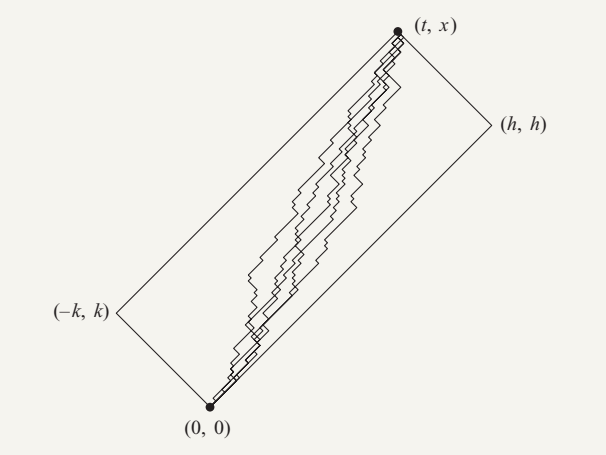


Figure 16

A few of the forward-directed spacetime paths from (0, 0) to (t, x) drawn on the lattice of Figure 8(a).

7. Conclusions

By recasting crystallography on a somewhat deeper and more generic foundation—a foundation based on constructs originating from distributed digital computation—one obtains a discipline that requires less and gives more. The fundamental topological concept is taken to be the partial order naturally induced by function composition, rather than the Euclidean metric as customary; the latter, when needed, can be obtained as an emergent construct, together with the pseudo-Euclidean metric of relativity.

Because the time aspect is made an integral part of the theory, not a separate plug-in, one can deal with concurrent, rather than just sequential, dynamics, either fundamental or emergent; describe spacetime unit cells in more general terms than merely spatial unit cells to which some time depth has been added; and deal with connections between regularity, on one hand, and effective computability on the other.

In sum, one obtains a more versatile, as well as conceptually simpler, tool, able to go beyond the needs of just mineralogy or materials science.

Acknowledgments

This research was funded in part by the U.S. Department of Energy under Grant No. DOE-4097-5. The author wishes to thank an anonymous referee.

References

1. W. Borchardt-Ott, *Crystallography*, Springer-Verlag, New York, 1993; ISBN: 3540594787.
2. M. J. Buerger, *Introduction to Crystal Geometry*, McGraw-Hill, Inc., New York, 1971.
3. J. Smith, *Geometrical and Structural Crystallography*, John Wiley & Sons, Inc., Hoboken, NJ, 1982.
4. P. Yale, *Geometry and Symmetry*, Dover Publications, New York, 1968.
5. S. L. Altman, *Icons and Symmetries*, Oxford University Press, New York, 1992.
6. U. Fano and A. R. P. Rau, *Symmetries in Quantum Physics*, Academic Press, San Diego, 1996.
7. I. Schensted, *A Course on the Applications of Group Theory to Quantum Mechanics*, NEO Press, Peaks Island, ME, 1976.
8. J. Rosen, *Symmetry in Science*, Springer-Verlag, New York, 1995.
9. N. Margolus, "Crystalline Computation," *Feynman and Computation*, A. J. G. Hey, Ed., Perseus Books, Reading, MA, 1999, pp. 267–305.
10. E. Fredkin and T. Toffoli, "Conservative Logic," *Int. J. Theor. Phys.* **21,** No. 3/4, 219–253 (1982).
11. J. C. Abbott, *Sets, Lattices and Boolean Algebras*, Allyn and Bacon, Boston, 1969.
12. W. H. Greub, *Linear Algebra*, Springer-Verlag, New York, 1981.
13. S. Wolfram, *A New Kind of Science*, Wolfram Media, Champaign, IL, 2002; ISBN: 1-57955-008-8.
14. E. Fredkin, "An Introduction to Digital Philosophy," *Int. J. Theor. Phys.* **42,** No. 2, 189–247 (February 2003).
15. See *digitalphilosophy@yahoogroups.com/*, Spring 2003 (*passim*) and thread starting with Juergen Schmidhuber's message of June 11, 2003.
16. K. Zuse, *Rechnender Raum*, Vieweg (Braunschweig) 1969; translated as "Calculating Space," *Tech. Transl*. AZT-70-164-GEMIT, MIT Project, Cambridge, MA, 1970.
17. J. von Neumann, *Theory of Self-Reproducing Automata*, A. Burks, Ed., University of Illinois Press, Urbana, IL, 1966.
18. G. 't Hooft, "Can Quantum Mechanics Be Reconciled with Cellular Automata?," *Int. J. Theor. Phys.* **42,** No. 2, 349–354 (February 2003), and personal communication.
19. G. Sussman and J. Wisdom, *Structure and Interpretation of Classical Mechanics*, MIT Press, Cambridge, MA, 2001; ISBN: 0-262-19455-4.
20. T. Toffoli and N. Margolus, "Invertible Cellular Automata: A Review," *Physica D* **45,** 229–253 (1990).
21. G. Doolen, Ed., *Lattice-Gas Methods for Partial Differential Equations*, Addison-Wesley, Boston, 1990.
22. J. Hardy, O. de Pazzis, and Y. Pomeau, "Molecular Dynamics of a Classical Lattice Gas: Transport Properties and Time Correlation Functions," *Phys. Rev. A* **13,** 1949–1960 (1976).
23. C. H. Bennett, N. Margolus, and T. Toffoli, "Bond-Energy Variables for Ising Spin-Glass Dynamics," *Phys. Rev. B* **37,** No. 4, 2254 (February 1988).
24. E. A. Wood, *Crystals and Light*, Dover Publications, New York, 1977; ISBN: 0486234312.
25. N. Margolus, T. Toffoli, and G. Vichniac, "Cellular-Automata Supercomputers for Fluid-Dynamics Processing," *Phys. Rev. Lett.* **56,** 1694–1696 (1986).
26. U. Frisch, D. d'Humières, B. Hasslacher, P. Lallemand, Y. Pomeau, and J.-P. Rivet, "Lattice Gas Hydrodynamics in Two and Three Dimensions," *Complex Syst.* **1,** No. 4, 649–707 (1987).
27. B. Grünbaum and G. C. Shepard, *Tilings and Patterns*, W. H. Freeman, New York, 1987.
28. J. Kari, "Reversibility of 2D Cellular Automata Is Undecidable," *Physica D* **45,** No. 1/3, 379–385 (1990).
29. J. Durand-Lose, "Representing Reversible Cellular Automata with Reversible Block Cellular Automata," *Discrete Models, Combinatorics, Computation and Geometry*, R. Cori, J. Mazoyer, M. Morvan, and R. Mosery, Eds., Springer-Verlag, New York, 2001.
30. T. Toffoli, S. Capobianco, and P. Mentrasti, "How to Turn a Second-Order Cellular Automaton into a Lattice Gas: A New Inversion Scheme," *Theor. Computer Sci*., 2004; in press.
31. T. Toffoli, "Action, or the Fungibility of Computation" *Feynman and Computation*, A. J. G. Hey, Ed., Perseus Books, Reading, MA, 1998, pp. 348–392.

Received July 25, 2003; accepted for publication September 9, 2003

Tommaso Toffoli *Boston University, Electrical and Computer Engineering Department, 8 Saint Mary's Street, Boston, Massachusetts 02215 (tt@bu.edu).* Dr. Toffoli received a doctorate in physics from the University of Rome in 1967 and a Ph.D. in computer and communication science from the University of Michigan in 1976. He joined the MIT Laboratory for Computer Science in 1977, eventually becoming the leader of the Information Mechanics Group. In 1995 he joined the faculty of Boston University's ECE Department. His main area of interest, information mechanics, deals with fundamental connections between physical and computational processes. He has developed and pioneered the use of cellular automata machines as a way of efficiently studying a variety of synthetic dynamical systems that reflect basic constraints of physical law, such as locality, uniformity, and invertibility. Related areas of interest are quantum computation, correspondence principles between microscopic laws and macroscopic behavior, quantitative measures of the "computation capacity" of a system—as contrasted to "information capacity"—and connections between Lagrangian action and amount of computation. Dr. Toffoli is at present intensely involved in a new initiative, called *A Knowledge Home*, which aims at developing cultural tools that will help ordinary people turn the computer into a natural extension of their personal faculties.