Reflections on PSCQM for Pierre's Book

December, 2000

Section 1. The significance of PSQM today

The first draft of PSCQM [1] is now more than four years old, and a good deal has happened since its was written. The present commentary will summarize some of these new developments. First, however, I would like to reflect a bit from the perspective of hindsight on the place of this paper within what I see as a larger ongoing transformation of the scientific worldview.

Prior to the twentieth century, no one but a few speculative philosophers questioned the primacy of space, time and matter as irreducibly fundamental concepts. But thanks to the uncertainty principle and the quantum "paradoxes" of amplitude cancellation, , there are a growing numbers of scientists today who believe we should be looking for definitions of space, time and matter in terms of even more fundamental concepts. There is an analogy here to the way heat, temperature and entropy came to be defined in terms of energy and probability. Entropy is particularly relevant to our present topic, since its definition extends beyond physics into probability theory in general. What PSCQM has shown is there is a similar extension into probability theory of the basic quantum formalism of state, transformation and observable. In short, the "core" of quantum mechanics belongs to mathematics as well as to physics.

Let's look briefly at how entropy overflowed the bounds of physics. First of all, note that its definition in statistical mechanics as summation pLog (p) refers only to probability, not to space, time or matter, and thus makes sense when applied to any probability distribution whatsoever, physical or not. Entropy becomes a physical concept only when we add to its abstract definition a physical definition of p. In classical physics, p is simply decreed to be proportional to volume in phase space. In quantum mechanics, however, p is more integral to the "machinery" of mechanics itself, since it is built into the very idea of observing a mechanical system.

The best-known general fact about physical entropy is the second law of thermodynamics. This law doesn't belong to mechanics, which is symmetrical in past and future. We do, however, see it universally manifested in the everyday physical world as an aspect of the *large-number* behavior of things that move or otherwise behave independently. The statistical explanation of the second law has been formalized within the mathematics of so-called *Markov chains*, which are abstract "processes" having nothing per se to do with space, time or matter. It is just this broader scope of the abstract second law that makes it such a plausible explanation of the physical second law. By simply extrapolating Markovian behavior to atoms that obey the laws of mechanics, it turns out that we can not only explain the physical second law but many other observed connections among entropy, energy and heat.

Historically, entropy began as a physical concept, and its statistical nature was not fully recognized for many years. The central concepts of quantum mechanics today, namely state, transformation and observable, have a position in the mainstream of physics today similar to that of entropy in the early nineteenth century.

These quantum concepts are of course physical concepts. They are, however, subject to two basic laws that don't in themselves have anything to do with matter in motion, namely $D' = T^{-1}DT$ and average(Q) = trace(QD), where D and D' are successive states, T is a transformation, and Q is any observable. It is these two laws about the operators D, T and Q that constitute what in PSCQM we have called the *quantum core*. A useful way to characterize the quantum core is as that part of quantum physics needed for the logical design of quantum computers. The variables of a quantum computer, like those of an ordinary computer, are simply bits, and the design of a quantum computer as such has only to do with how these bits relate to each other and nothing to do with space or matter.

Might it be that the quantum core, like entropy and the second law, really belongs to a wider domain of nature than physics? This is what PSCQM set out to show, and what it did in fact show is the mathematical fact that *the laws of the quantum core, in a generalized form, are universal laws governing Markov processes*. There is a remarkable parallel here with the second law. There is also a curious complementarity. The second law is asymmetrical in past and future, and indeed is often said to define the arrow of time. The quantum core laws, on the other hand, are time symmetrical, and this in fact turns out to be their defining property in the Markovian setting. All this will be spelled out further below; suffice to say here that the general category of Markov processes allows for harmonious hybrids of second-law and quantum processes. These are set within a covering theory that permits a much larger range of Markovian structures, some of which even have two kinds of entropy, one increasing and the other decreasing with time.

In order for the abstract second law to become a physical law it had to be interpreted as a large-number law governing physical things. The same is of course true of the quantum core. For the second law, those physical things are atoms. But atoms won't do for the quantum core, since it is the precisely the quantum behavior of things like atoms we are trying to explain by large-number laws. Recall that I began by remarking that people are starting to question the primacy of space, time and matter, and to ask whether these should be replaced by more fundamental concepts. If we hope to turn the abstract quantum core into real physics, this is no longer an idle question.

I also began by saying that the larger picture is not just about the future of physics but about the future of science itself. The science you and I learned in school took space, time and matter for granted, as it did the temporal arrow defined by the mathematics of Markov chains. The primacy of space, time and matter, classically conceived, underlies not only the specific content of the sciences but the whole of scientific method. If it is abandoned, what will replace it? What kind of science will our descendents learn in the classrooms of 2100?

It's time to get more specific about Markov processes and Markov chains. Just what are these things? Unfortunately, the answer varies quite a bit from one author to the next. I have to confess that even our Markovian terminology in PSCQM is not entirely consistent, so I'll try to clean it up somewhat here. We'll start out with the definitions in Feller's classic textbook [2], and then introduce some further distinctions that he omits.

Feller defines what he calls a *general Markov process* as a succession of variables, continuous or discrete, on which there is a joint probability distribution satisfying what I'll call the *Markov separation law*. The usual informal statement of this law is that the past only influences the future via the present. This statement is misleading in one important respect, though, in that it appears to mathematically distinguish past from future, whereas what it actually asserts is *symmetrical* in past and future. A better, though mathematically equivalent, statement of the separation law is that the correlation of the past with the future depends only on the correlation of the past with the present and of the present with the future. Feller's qualification "general" is redundant, so I'll refer to his general Markov processes simply as Markov processes unless their generality needs to be emphasized. The definition of Markov process in PSCQM is in one respect more general than Feller's in that it allows for negative as well as positive probabilities, a point to which we'll return in the next section. This doesn't bear on what follows here, however, so we for now can continue to think of probabilities in the usual positive sense.

A *Markov chain,* on the other hand, is defined as the repeated application of a *transition matrix* of conditional probabilities to an initial probability distribution. More exactly, it is the joint probability distribution that results from this repeated application. It is easily shown that, in the absence of other conditions, this joint probability distribution is always a (general) Markov process. The converse is not true, however; not every Markov process is a Markov chain, a fact that is crucial to our present topic.

How do these two definitions differ? For one thing, chains are more specialized, as we just remarked. However, there is a more fundamental difference. A Markov *process* is a joint probability distribution given simply *as such*, whereas a Markov *chain* is a joint probability distribution given as a *composition* of simpler parts. To make an analogy to integers, the difference is like that between an integer as such and an integer given as a product of its prime factors. Or to switch the analogy to vectors, it's like that between a vector as such and a vector as a superposition of basis vectors. This distinction here is subtle, but extremely important.

There is another important distinction to be made, which is that between a Markov composition of *arbitrary* parts and a Markov composition whose parts are all *alike*. If the parts of the latter, except for the first, are transition matrices, it is a Markov chain. To pursue the analogy to numbers, the difference between a general Markov composition and a Markov chain is like that between a number, any number, factored into its primes, and a number that is given as a power of a single prime. Any Markov process can be factored into *some* collection of transition matrices, but only a Markov chain can be factored into a collection of transition matrices that are all *alike*. To put it less formally, the law of change in a general Markov process can be *variable* while that in a Markov chain is *fixed*.

A useful way to characterize a Markov process is to think of it as a computer with random inputs. Clearly such a computer is a Markov process, since, if we are given its state at time t, then the probability of its states at time t+1 is independent of its history prior to t. If the computer is a *closed system*, i.e. if its random inputs are from a constant random source and it has no other inputs, then its transition probabilities are constant, i.e., it's a Markov chain. If on the other hand it is an *open system*, i.e. its transition probabilities depend on variable random sources or inputs from other systems, then it is a general Markov process. The same considerations apply to processes running inside a computer, which includes all real-time simulations.

Given the importance of computer modeling in today's science, it's hardly an exaggeration to say that, for most scientists, to explain something means to describe it in a way that could in principle be turned into a real-time computer simulation. This belief, which I'll call *computerism*, usually does not rise to the level of an explicit statement; it's just one of those things that "goes without saying". It's a funny thing about things that go without saying, though, which is that when you actually say them carefully, and then take a close look at what you have said, they sometimes turn out to be wrong!

Is computerism wrong? That's not something I'll take sides on here. However, I have observed that many people hold onto computerism simply because they can't imagine any other possibility. Here is where a proper understanding of Markov processes makes a big difference. It turns out that computers are only a tiny island in the vast sea of formal possibilities encompassed by the general concept of a Markov process. The quantum is another tiny island. As mentioned, there are also hybrid forms that belong to neither island. The important point is that by no stretch of imagination can the encompassing expanse of Markovian forms be located on Computer Island alone. Quantum structures can't be located there, even quantum computers can't be located there, and most of the remaining expanse isn't even in sight.

Which brings us to the future of science. Physical science grew up in close collaboration with engineering, and for the most part shares with engineering a view of the world as something to be taken apart into functional units. To this the engineer adds the art of reassembling functional units into useful functional wholes; this is called technology. The abstract skeleton of a functional part is a *transition matrix*, also sometimes called a *transfer function*, representing the functional dependence of a set of *outputs* on a set of *inputs*. In the deterministic or "causal" case, the actual values of the outputs are a function of the values of the inputs, while in the more general case it is only the probabilities of these values that are a function of the inputs. The *generality* of engineering consists in its being to able to use a small variety of functional parts and design principles to assemble a large variety of useful complex structures.

Here is where I see the broader significance of PSCQM. I believe its chief accomplishment was to mathematically extend the basic conception of lawful change that underlies current scientific practice. This extended lawfulness retains Markovian separability, but no longer requires that we separate things into functional parts. To put it another way, it no longer requires that the internal variables be inputs connected to outputs. The links between parts, and even between past and future, can now have a two-way information flow. This is easy to say, and it turns out to be rather easy to formulate mathematically, but it also turns out to be very hard to digest. Indeed, most of the work since PSCQM has involved trying to digest it. We have studied numerous examples, which provided numerous surprises, and a lot of work has gone into grounding the mathematics at a more fundamental level – we'll come to this in the next section.

Major changes in science are foreshadowed by movements in the culture at large. A variety of cultural movements in modern times, ranging from the counterculture of Woodstock to the arcane isms of Continental philosophy, share a strong discontent with the technocratic narrowness of science as it stands. The broad message here is that nature, including human nature, has many ways of *being* besides *using things*. A world that is nothing but *functionality* is a world fit only to be *used*. The world of the engineer is an abstraction geared to a particular mode of activity, not the world we live in.

But the world of the engineer is also an enormous intellectual achievement, and there is the problem. It is romantic folly to think that throwing away this achievement would return us to some imagined idyllic state of nature. I would like to think that PSQM offers a hint of a less foolish path. It clearly describes radical alternatives to functional composition that are none-the-less accessible to the engineer's mathematical tools. It also shows how these can simply explain some of the more puzzling laws of physics. This is certainly not The Answer, but it does offer hope that there may be ways to steer the intellectual power of science into a better partnership with our real human nature.

Section 2. Further developments: Link Theory

Not long after the original PSCQM was finished I discovered a simpler way to formulize its mathematics that has proved to be much easier to understand and use. The name "link theory", which came up a few times informally in PSCQM, has been transferred to this new formalization. Link theory was first presented at PhysComp '96, and soon after became part of the research agenda of the theory group headed by Dick Shoup at Interval Research, Inc. After the demise of Interval, I continued to work on some of its more abstract aspects at Hewlett-Packard in collaboration with Jim Bowery of the E-Speak project. More recently, I have rejoined Dick Shoup in continuing our Interval work at the newly formed Boundary Institute. Apart from the original idea of link theory, the developments I will now describe took place at one or the other of these three institutions.

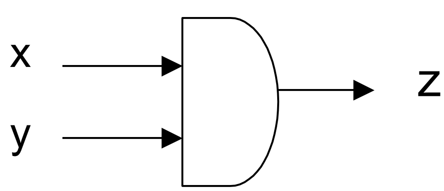
PSCQM uses the standard language of probability theory: random variables, sample space, marginals etc. In terminology it follows Feller, though not always in notation. This kind of language was designed for quite different purposes than link theory, however, and is a bit awkward at best. Link theory, so far, has been entirely finitistic. Issues of convergence, continuity etc. don't arise, so the measure-theoretic generality of standard probability language is wasted and just muddies the waters. For present purposes, it's better to go back to the Eighteenth Century conception of probability as defined by case counts, as in calculating the odds in a game of chance.

By a *case* will be meant the assignment of a value to each of a set of variables. For instance, $\{x=2, y = 1, z=3\}$ would be a case of x, y and z. We say that several variables are *uncorrelated* or *independent* if all cases are allowed; to put it another way, if knowing the values of some of the variables gives no information about the values of the others.

Laplace defined *probability* as the number of *favorable* cases divided by the total number of cases. For instance, the odds of getting a four in rolling a pair of dice x and y is the number of cases where x and y add up to four, i.e. 3, divided by the total number of cases when x and y are independent, which is 36; thus it is 1 in 12. This of course assumes that the dice are unbiased. We can handle biased dice by imagining that there is a more numerous set of equally probable cases of which the observed cases are unequal fractions. This makes case counting adequate at least for the mathematics of combinatorial probability theory, and its intuitive transparency compared to measure theory is a great advantage.

It turns out to be very useful to represent the correlation among a set of variables by writing their allowed cases as the rows of a table whose columns are labeled by these variables. In logic, where there are only two values 0 and 1, these are known as *truth tables*. The truth table of an AND gate, for instance, has three columns labeled x, y and z, where x and y are the inputs and z the output, and four rows 000, 010, 101, and 111. The probability of a statement S about x, y z etc. is the number of rows for which S is true divided by the total number of rows. If S is the statement z = 1 about the AND gate, for instance, we go down the table and pick out the single case where z = 1, showing that the probability of S is ¼.

In this example, S only involved the single variable z, which means we need only to look at the z column to count the cases "favorable" to S. More generally, when we are working with probabilities that only involve a subset of our variables, we can simply erase the columns headed by the others. The resulting table will, in general, contain duplicate rows. Instead of writing out all of these duplicates, we'll adopt the shorthand of writing each row only once, followed by its *count*. Such *count* tables are the basic notational building blocks of link theory in its new form. Here are the rows of the count table of our AND gate if we are only considering the variables x and z: 00|2, 10|1, 11|1. Remember, this is just an abbreviation; the rows of the actual table with y hidden are 00, 00, 10, 11.



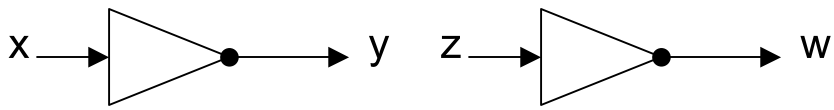
x	y	z	n
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	1

x	z	n
0	0	2
1	0	1
1	1	1

Fig 2.1. Count table of an AND gate {x, y, z} and of its subtable {x, z}

The count column of a count table assigns to each row a case count proportional to its probability. Normalizing the count column by the total count gives the actual probability distribution on the variables of the chosen subtable. In practice it is usually better to work directly with case counts and to postpone normalization until our combinatoric calculations are finished. Thus the count table itself will take on the basic role of the sample space in standard probability theory.

We've considered independent variables; now let's turn to independent *subtables.* Consider a table T with four columns x, y, z , w. Assume there are no duplicate rows in T. Let T_1 be the subtable of T consisting of columns x and y, and T_2 be the subtable consisting of columns z and w. We say that T_1 and T_2 are *independent* in T if the rows of T are the pairs of *distinct* rows of T_1 and T_2. Notice that each distinct row of T_1 occurs in T as many times as there are distinct rows of T_2, and vice-versa. Here, for instance, is a truth table T in which T_1 and T_2 are independent NOT gates:



x	y	z	w	n
0	1	0	1	1
0	1	1	0	1
1	0	0	1	1
1	0	1	0	1

Fig. 2.2. Truth table of two independent NOT gates.

Recall that in Section 1 we distinguished between a joint probability distribution given simply as such, and a joint probability distribution given as a composition of simpler parts. In our present context, the counterpart of the former is a count table given simply as such, as remarked above. What, then, is the counterpart of the latter? How do we take apart a table?

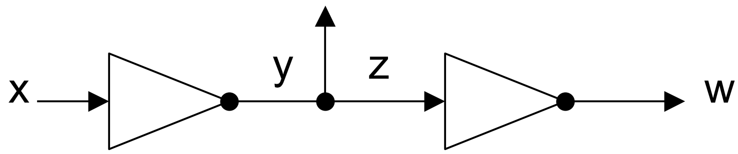
For the simple table in Fig 1, the answer is obvious. The parts are two NOT gates, and to combine them into T, we take every combination of their rows. This operation is called *outer product*, and it applies to any two tables given separately. If the component tables are count tables, we multiply their separate row counts to get the row counts of T. It is easily shown that every table can be uniquely factored in this way into a product of "prime" tables.

In a Markov process, however, there are no independent parts, except in the most degenerate cases. How, then, do we take apart a Markov table?

Here is where link theory departs fundamentally from standard practice. Recall that the parts of a Markov chain, as defined by Feller, are *transition matrices* of conditional probabilities. We could follow standard practice in our table notation by introducing a new kind of table, call it a "transition table", whose rows are *conditional* cases. This would considerably complicate the logic of table analysis, however. There is a much better way, called *linking*.

The basic idea of linking is simplicity itself. Given a table T with variables x and y, to *link* x and y means to create a new table T' from T by requiring that x and y be equal. That is, T' is the table that results from discarding all of the rows of T in which the values of x and y disagree.

Let's create a table T' by linking the variables y and z of T in Fig. 1:



x	y	w	n
0	1	0	1
1	0	1	1

Fig 2.3. Two linked NOT gates

In effect, we are wiring the output of the first NOT gate to the input of the second. This turns the table of the outer two variables x and w into a straight "wire". Notice, incidentally, that we would get the same table T' by linking x and w. There is a lesson to be learned in passing here, which is that de-linking is not always unique. We'll encounter more subtle and important cases of this non-uniqueness in Section 3.

It's easy to extrapolate from this example to the general case of wiring up elementary logic gates into a logic diagram. A link is simply a wire from one gate terminal to another. Certain terminals are chosen as inputs, others as outputs. The rest are ignored; these are the so-called "hidden" variables. A computer is an initial state plus a repeated logic diagram, one copy for each clock tick, with some of the outputs from the copy at t linked to some of the inputs to the copy at t+1. Here is a link diagram of a very simple computer, whose logic diagram consists of a single AND gate:

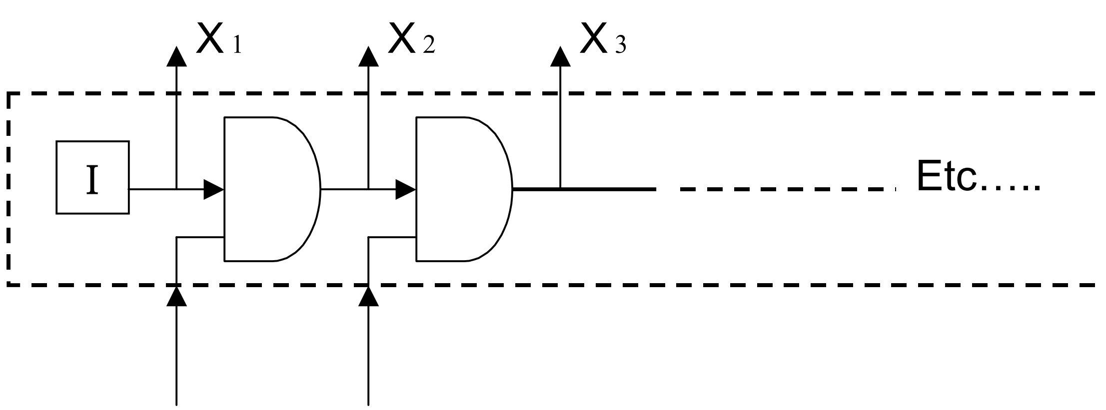


Fig 2.4. Simple open computer

Notice that we have "exposed" all of the variables in this diagram by running them outside of the dotted box. Since there are inputs from the external world, this computer must be thought of as an open system. As such it is a deterministic system. Its case table shows its inputs as independent variables and its outputs as functions of its inputs.

Had these inputs been left dangling inside the dotted box, they would be treated as hidden variables and thus erased from the case table:

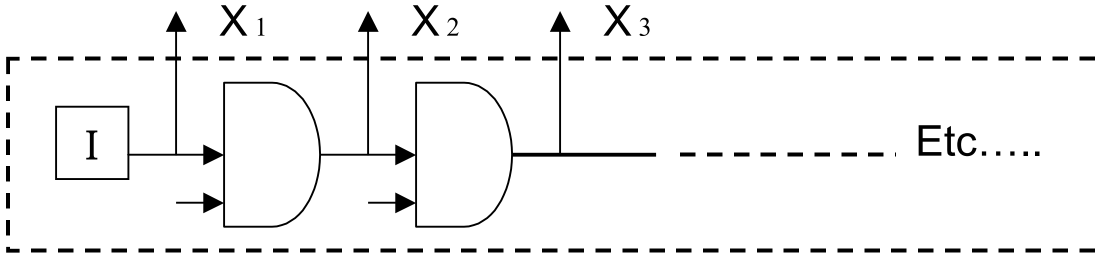


Fig. 2.5. Computer with "random inputs" as a Markov process

The outputs are no longer a function of the inputs, and are thus undetermined. The case table has well-defined case counts, however, so its outputs have a well-defined joint probability distribution. It's easy to show that this distribution is a Markov process (we must of course supply an initial state to the first upper input.)

We now come to a very important point. Remember, we carefully distinguished a Markov process from a Markov chain, which is a Markov process that has been "taken apart" into copies of a transition matrix. Fig. 2.5 shows a Markov process taken apart, not into copies of a transition matrix, but into copies of a partial case table. Be aware: this is a fundamentally different way to take things apart!

Let's go back to the analogy to numbers. What are the parts of a number? Its prime factors? That's one good answer. But an equally good answer is its decimal digits. Neither of these is the *right* answer, but each can be right in the appropriate context. There is one important difference between the two, however, which is that the first only applies to *whole* numbers while the second also apply to fractions and even to irrational "quantities", to use the Greek term. Indeed, for the Greeks, irrational quantities were geometric rather than arithmetic entities; it was only after the invention of infinite decimal representations that it made sense to speak of irrational quantities as numbers.

As the reader may have guessed, the Markov process of Fig. 2.5 is a Markov chain. More accurately, we can reinterpret its AND gate with y hidden as a *transition matrix*. We can in fact do this for any case table representing a correlation between input and output variables. Indeed the very concepts of "input" and "output" rest on interpreting cases *conditionally*; we say of a NOT gate, for instance, that *if* the input is 0, *then* the output is 1, and *if* the input is 1 *then* the output is 0. Only certain kinds of case tables can be reinterpreted as transition matrices, however. Here is where linking shows its essentially greater generality, since there is no restriction on what kind of case tables can be "chained" by linking,

The result of either kind of chaining is a Markov process, and a *lawful* Markov process in the sense that the rule connecting adjacent states is constant. However, linked chains are a much larger class of lawful structures than Markov chains. Perhaps a better way to put it is that link analysis *extends* the concept of Markov chain, just as infinite decimal notation extends the concept of number. We don't yet have a word for this new kind of chain, so let me here coin one: *parachain*. This term has a precedent in the term *paracausality*, which I used in a 1977 paper for a somewhat more general concept.

Special kinds of parachains were studied as long ago as the 1960's, the earliest examples that I know of being presented in papers by the astrophysicist Helmut Schmidt and myself [3]. It is only since link theory, however, that the concept has received a clear definition at an elementary level.

Do parachains encompass quantum processes? Not quite. The class of Markov chains, and indeed of Markov processes, must be extended a little further. To see what this involves, let's consider a real-world problem.

Imagine that killer bees have spread over a large part of the US and are continuing to spread. Where did they come from? We can trace them back to Texas, but we'd like to trace them back to the port or border town in Texas where they first made their unwelcome appearance. We have no empirical evidence about this early phase, so what we are confronted with is a problem in statistical retrodiction.

To tackle this problem, we first observe how the bees are continuing to spread and use this information to construct a *diffusion equation*, which is a certain kind of Markov chain. If this equation describes an unchanging law, we can use it to calculate the bee distribution at any time by simply running it forward or backward. Now running it forward means successively applying the Markov transition matrix to an initial state. But running it backward means successively applying the *inverse* of this transition matrix to a *final* state, assuming that it does have an inverse (most transition matrices do). The "logic" of this process is just like that of the forward process in that we can think of each entry in the inverse matrix as the proportion of bees starting in one place that end in another. There is one difference here, though, in that some of these "proportions" in the inverse matrix will be negative!

Just what is a negative swarm of bees? That's actually a deep question, to which we'll briefly return in Section 3. But then, just what is a negative pile of dollars? Suffice to note here that negative bees, like negative dollars, solve practical problems, and also enter our mathematics very easily and smoothly. The way we introduce them into link theory is by giving the their cases a negative sign; when we abbreviate duplicate rows to create a count table, those rows with negative signs subtract from the count. As we'll see, this allows both probabilities and probability amplitudes in quantum mechanics to be defined as ratios of case counts. This is spelled out in detail in PSCQM, where we also show, following Mackey, that complex amplitudes can be defined in a quantum mechanics with only real amplitudes by imposing a certain symmetry on its operators.

Section 3. Further developments: Relational logic and Markov cycles

At Interval we applied link theory to a variety of things ranging from puzzles and problems in AI to quantum computers. We also programmed a link calculator, which has since become indispensable in studying examples. Actually, we made several link calculators, the first of which was just a simple Microsoft Access macro. It turns out that the Link Theory operations of linking, hiding columns and creating count tables are in fact standard operations in relational databases, something I hadn't realized at first. It was this primitive Access calculator that first confirmed the validity of link diagrams as a way of representing quantum computers.

The simplicity of the Access calculator points to the deep roots of Link Theory in the logic of relations, a topic I have spent a good deal of time looking into over the past several years. Two long papers emerged from this work, each taking its departure from what Russell and Whitehead in *Principia Mathematica* called *Relation-Arithmetic*. The first, called *Structure Theory* [4], fixed a flaw in the *Principia* exposition that had prevented *Relation-Arithmetic* from dealing properly with relational composition. This made it possible to formulate Link Theory at a very abstract level. The second, called *Relation-Arithmetic Revived* [5], was written at Hewlett-Packard as part the theoretical work that Jim Bowery and I were doing on the design of transactional languages for the Internet. It went much further in developing another idea briefly introduced in *Structure Theory*, which is that the theory of relations, and indeed the whole of mathematics, can be formulated in a language whose only primitive predicates are *identity relations*. This new work appears to have good long-range prospects for putting link theory on a deeper logical foundation, but is outside the scope of the present paper.

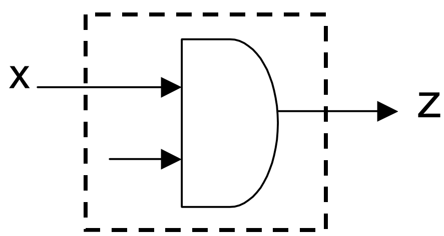
What I want to present here in conclusion is a brief account of some surprising newly discovered facts about Markov processes that have emerged from our theoretical work at the Boundary Institute.

I pointed out in the last section that Markov chains are only a small fraction of the full range of orderly Markov processes, i.e., of Markov processes that have a constant dynamical law. I mentioned that link theory generalizes Markov chains by taking their "dynamical parts" to be linked two-variable probability distributions rather than transition matrices. It also generalizes Markov chains in a different way, which is by allowing independent boundary conditions on both the past and *future*.

In standard Markov theory, where the matrices represent *conditional* probability distributions, it would make no sense to place a condition on the future, since that would put it on the wrong end of the *if-then* arrow. However, in a link representation there *is* no if-then arrow. There is indeed conditionality, but it operates in a very different way: the conditions on a link composition are its essentially timeless *links*.

Let me spell this out a bit. In probability theory, to *apply* a condition means to disallow certain cases. Translated into the language of link theory, this means discarding certain rows. When we construct a Markov process in link theory, we start out by forming the outer product of a collection of two-variable tables, which are the dynamical components, together with a one-variable table, which is the initial state. The result is a table with many independent parts and an exponentially larger number of rows. To create the *interdependence* that turns our table into a Markov process, we shorten this large list of rows by linking the successive parts, which means applying *conditions* of the form Y = X that discard every row in which the "output" Y and "input" X are unequal.

As mentioned, a Markov *chain* in link theory is a link composition in which the (repeated) transition matrix is replaced by a (repeated) table. The relationship between this matrix and this table is very simple. Let M_{ij} be the matrix. The corresponding count table T(i,j) is a table with columns i and j which has a row for every pair of values of i and j whose count is M_{ij} (this ignores normalization, but in link theory only the ratios of counts are significant.) The matrix M, as a transition matrix, has the special property that its columns all have the same sum. A link table need not be similarly constrained, i.e., its corresponding matrix need not be a transition matrix, which is why the Markov processes represented by a repeated link table are of a more general form than Markov chains.



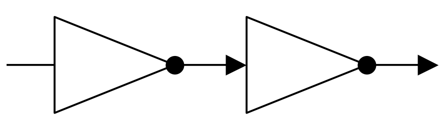
=

x	z	n
0	0	2
1	0	1
1	1	1

$$= \begin{pmatrix} 2 & 1 \\ 0 & 1 \end{pmatrix}$$

Fig. 3.1 Matrix of an AND gate with one input hidden

Multiplying matrices M and N corresponds to linking their tables and then hiding their linked variables, leaving the remaining outer variables as the indices of MN. This correspondence between table and matrix operations is crucial for the application of link theory to standard science and probability theory. Here is a comparison of the two operations for a pair of NOT gates:



$$= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} =$$

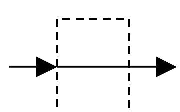


Fig 3.2 Multiplying the matrices of two NOT gates

The interchangeability of matrices and tables makes it possible to give meaning to the concept of a *final* boundary condition on a Markov chain. Here is how this works:

Start with a Markov chain given by an initial state vector I and a transition matrix T. Make I into a one-column table and T into a two-column table and replace products by links; this yields a link representation of the same Markov process. To apply a final condition given by a vector F, simply make F into a one-column table and link it to the last variable of the T series. Compositions created in this way also constitute an essentially larger class than Markov chains; they are also *parachains*, to use the term coined in Section 2. Here is a simple example involving three AND gates. It's an interesting exercise to work out its count table and compare that to the count table of the example in fig. 2.5.

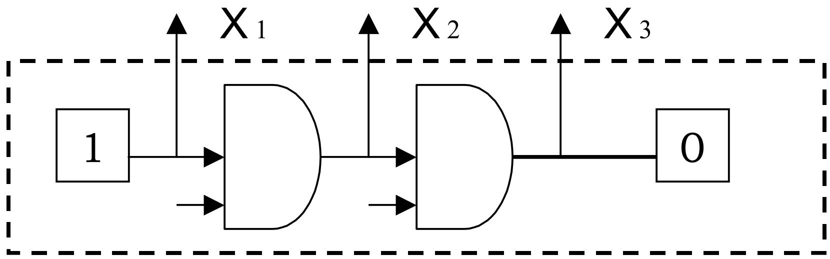


Fig.3.3 Simple finalistic parachain

Do such finalistic parachains exist in nature? Since "final causes" are generally rejected by today's science, the reader's first reaction is probably a simple no. However, leaving metaphysics aside, the empirical issues here are much more subtle than they might at first seem, as the following mathematical argument will show.

The time-evolution of a Newtonian system with n degrees of freedom is determined by an initial state that must, in general, be given by 2n parameters. In the most familiar cases, these n degrees of freedom are positions and orientations, the other parameters being their rates of change, or *velocities*. For *perfectly damped* systems, however, such as a capacitor discharging into a resistor, the velocity parameters are redundant. Indeed, a perfectly damped system can be *defined* as one for which knowing the rates of change of its degrees of freedom supplies no additional predictive information. Aristotelian physics, stripped of its metaphysics, is simply the physics of perfectly damped systems. As such, it persisted well into modern time as a respectable competitor to kinetic theory; Newton, for instance, proposed an Aristotelian explanation of Hooke's law [6].

In around 1730 the French priest Maupertuis brought final causes into physics with his principle of least action, which says that if we are given the values of the n degrees of freedom of a Newtonian system at the beginning and end of a certain period of time, we can calculate its time-evolution over that period by minimizing a certain integral. Notice the resemblance to our parachain example above, where an initial and final vector together with the law of the process gives the time-evolution of the probability vector. In the special case of an Aristotelian (perfectly damped) physical system, the final n parameters are of course redundant. Similarly, in the special case of an ordinary Markov chain represented as a parachain, its final boundary condition is redundant, and it can be shown that this final condition must always have the special form of a table whose counts are all equal (i.e. it must be a *white* vector; see PSCQM).

The resemblance between these two kinds of finality becomes even closer when we look at certain kinds of continuous parachains, most notably random walks for which we are given both the initial and final positions. It turns out that it is by minimizing a certain integral with the dimensions of *information* that we get the expected trajectory of such a doubly conditioned random walk. This actually leads in the limit to the laws of Newtonian mechanics for the walker if we identify dispersion rate with mass [3], and it turns out that there are features of this situation suggestive of both quantum mechanics and relativity. That's another story, however. For the present, the following are the essential points:

Aristotelian physics is the special case. The general case is Newtonian physics.

The theory of Markov chains is the special case. The general case is the theory of Markov parachains.

In around 1750, final causes made a hasty and somewhat embarrassed retreat from physics when Lagrange showed that they could always be replaced by initial velocities. In essence, the reason for this is very simple. To calculate the trajectory of a general Newtonian system with n degrees of freedom we need 2n independent parameters, but these parameters can belong to the state of the system at any time so long as they are independent. In particular, we can have n position parameters at the beginning and n at the end. But we don't have to specify when that end comes! Even if the process ends after an infinitesimal time dt, our 2n parameters still remain independent and thus give us enough information to calculate the trajectory for all time. Thus, as Lagrange so famously remarked, we can start with initial positions and velocities and predict the exact course of events as far into the future as we wish. The principle of least action is completely equivalent to this formulation that puts both boundary conditions at the beginning.

What I have recently shown is that Lagrange's theorem on replacing final causes by initial velocities has a precise analogue in parachains. The mathematical details can be found on the Boundary Institute web site [7]; here I'll give a very informal sketch.

It turns out that in any Markov process there is a purely probabilistic analogue of an evolving "dynamical" state, which is simply the matrix of joint probabilities on a pair of adjacent variables X_t and X_{t+1}. We'll call this the *digram state* G_t at time t. In link theory language, G_t is the two-column table on X_t and X_{t+1} that results from hiding all other columns in the process table.

We'll come to the actual dynamical law for G_t shortly, but first I would like to reflect a bit on how this new result bears on the discussion of "computerism" in Section 2. Recall that computerism was defined as the belief that to explain something means to describe it in a way that could in principle be turned into a real-time computer simulation. In practice this means using the concept of a Markov chain as the paradigm of lawful process. Now a Markov chain is a perfectly damped system as defined above, since knowing how a state has changed tells us nothing more about the states to come. The "parameters" of a computer state are simply the computer's bits; knowing their "velocities", i.e. whether or not they have changed since the last tick, add no predictive information. Thus computerism, right or wrong, must be seen as a throwback to Aristotelian thinking.

Our history books usually tell us that Galileo and Newton refuted Aristotelian physics. That's an oversimplification. One can almost always cook up ad hoc Aristotelian gadgetry to explain the observed facts, as illustrated by Newton's explanation of Hooke's law. That's why Aristotelian physics lasted so long. The eventual triumph of Newtonian over Aristotelian physics was not so much the triumph of truth over falsehood as of conceptual simplicity and unity over overextended common sense. Today's computers have greatly increased our ability to deal with what we perceive to be complex situations without our having to find new ways of simplifying them. This is not necessarily a good thing. It's interesting to speculate what the history of physics might have been if present-day computers had existed in the sixteenth and seventeenth centuries.

The essence of the Newtonian revolution in physics was to bring velocity into the concept of state. Parachains bring velocity into the states of those systems studied by the information sciences. Therefore, when I say we should study parachains, I am not so much calling for a new revolution as enlisting in an old revolution that still has important unfinished business.

Let's now turn to the dynamics of digram states. To understand the digram dynamical rule we must introduce a new algebraic concept: the *circle product*. In link theory terms, the circle product A°B is the *linked cycle* consisting of a pair of two-column tables A and B, as shown in Fig 3.4A and 3.4B.

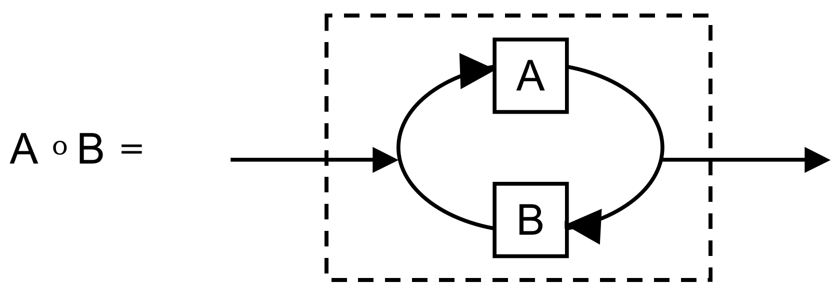
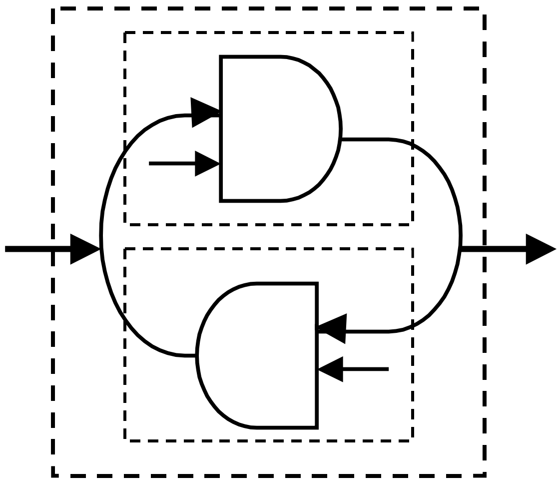


Fig 3.4A. Circle product



$$= \begin{pmatrix} 2 & 1 \\ 0 & 1 \end{pmatrix} \circ \begin{pmatrix} 2 & 0 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 4 & 0 \\ 0 & 1 \end{pmatrix}$$

Fig 3.4B. Circle product of two AND gates

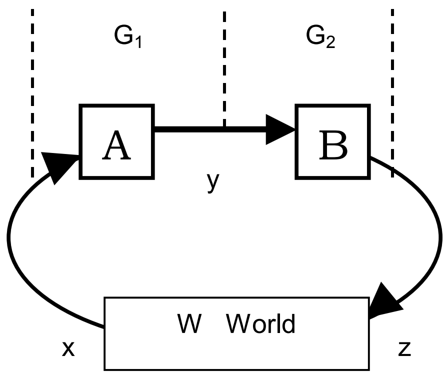
In working with the circle product it's important to keep track of the arrow directions – ingoing is column 1, outgoing is column 2 – since reversing arrows will usually create a different table. One consequence of this directionality is that A°B is usually not equal to B° A. We'll write A* for A with arrows reversed. Circle algebra is neither commutative nor associative; instead, we have the two rules: B°A = A* ° B* and A°(B°C) = (A°B)°C*. Despite these peculiarities, it's a very useful tool for analyzing digram states and their relationship to link concepts such as the density matrix, as we'll soon see. One hint of this comes from the fact that, for unitary U, U°U is the matrix of transition probabilities connecting a preparation to a measurement in quantum mechanics.

The example in 3.4 shows the circle product of the AND gate x-z subtable shown in Fig 2.1 times itself. Let's call this subtable A. In matrix form, the "circle square" A°A of A is gotten by multiplying the elements of A by the corresponding elements of A*, which is the transpose of A. This is the general rule for A°B if A and B are square matrices of the same size.

A *Markov cycle* is defined as a joint probability distribution that can be given by a cycle of linked two-column tables. A Markov process can then be redefined as a Markov cycle in which at least one box is the *empty box* E, where E is the matrix whose elements are all 1's. Digram dynamics is most naturally formulated for Markov cycles; it can then be easily applied to Markov processes and Markov chains as special cases.

The empty box E has an important role in circle algebra, where it is the *right identity*, i.e. A°E = A. However, because of non-associativity it is not the left identity; rather we have E°A = A*. Nevertheless, it can be used to define the *circle inverse* $A^{\sim}$ of A as the table satisfying $A°A^{\sim} = A^{\sim}°A = E$. We'll also sometimes write the circle inverse as E/A, which makes certain formulas more understandable. The circle inverse of a matrix is gotten by first inverting all of its elements and the taking the transpose.

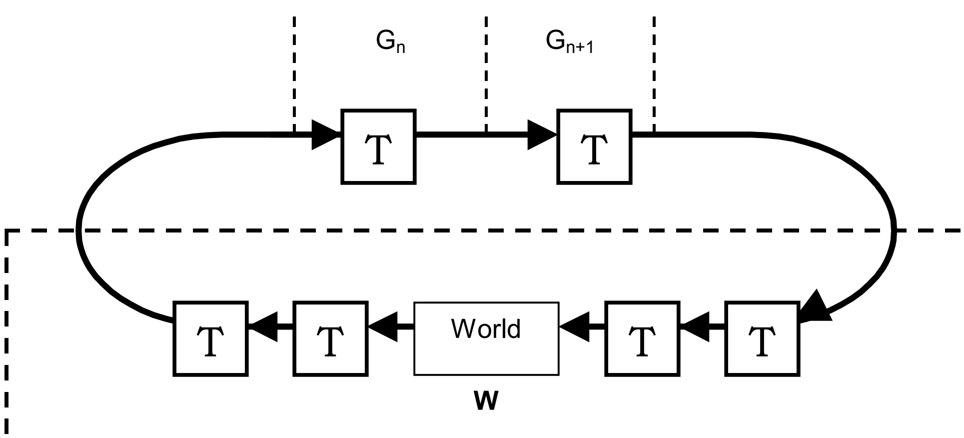
The basic formula of digram dynamics, which allows one to calculate successive digram states, can be derived as a theorem about three-box Markov cycles. In Fig 3.5 we can think of the cycle ABW as a process that separates from the world W at x, remains isolated at y and returns W at z. The digram state G_1 is the subtable on x and y that results from hiding z, while G_2 is the subtable on y and z that results from hiding x. G_2 is derived from G_1 by the formula $G_2 = B°(B^{-1}(A^{\sim} \circ G_1)A)$.



$$G_1 = A°BW$$
$$G_2 = B°WA$$
$$G_2 = B°(B^{-1}(A^{\sim} \circ G_1)A)$$

Fig 3.5. The digram transition formula

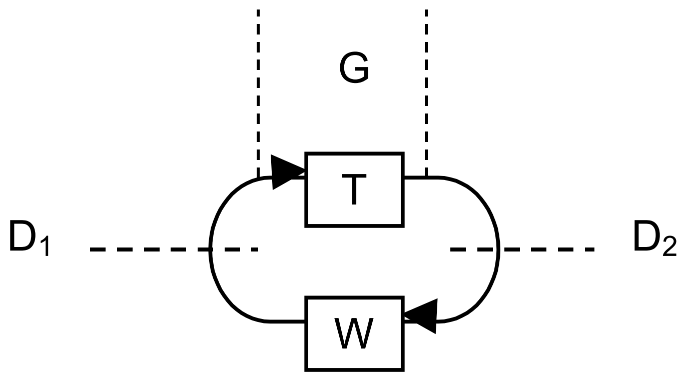
The most important thing to note about this formula is that it doesn't explicitly contain the world box W. It only refers to the two isolated boxes A and B and their digram states. This means it can be applied to any two consecutive boxes in an isolated series by simply dumping the remaining boxes into W:



$$G_{n+1} = T°(T^{-1}(T^{\sim} \circ G_n)T)$$

Fig 3.6. Digram dynamics for a parachain

This formula, apart from being rather long, does not easily fit into standard mathematics because of its mixture of the two kinds of product. Worse yet, it does not gracefully go to the limit with G_n as an infinitesimal. This makes it awkward to apply to continuous processes, though using a finite G that “slides” along the time axis can in fact do the trick. Fortunately, there is better way, which is to convert to *density matrices*.



$$D_1 = T(T^{\sim} \circ G)$$
$$G = T \circ (T^{-1}D_1)$$
$$D_2 = T^{-1}D_1T$$

Fig 3.7. Density matrix dynamics

A density matrix, as defined in PSCQM, is the matrix that results from removing a link in a link diagram and hiding all but the two unlinked variables. It is this definition that leads to the combinatorial interpretation of quantum mechanics. Rather surprisingly, it turns out that, given the transformation matrix T, the density matrix can be defined in terms of the digram matrix and vice versa; the conversion formulae are $D = T(T^{\sim} \circ G)$ and $G = T \circ (T^{-1}D)$. The density matrix dynamical law $D_2 = T^{-1}D_1T$ is much more manageable and has a simple infinitesimal form, where T is written as $1 + Adt$. The great advantage of the digram law is that it refers to potentially observable quantities – G is an *actual* joint probability distribution that might well be observed as a statistical distribution – whereas D is a *counterfactual* distribution that can only be observed by breaking a connection. However, by using the above conversion formulas we can have the best of both worlds, using finite G’s in the context of measurement and working with infinitesimal T’s in the context of smooth change.

This has been a very condensed summary of digram dynamics; as mentioned, a more expanded version can be found in [7]. In conclusion, let me make one final point about Newton vs. Aristotle.

We first defined a Markov process in Section 1 as a process for which the past plus the present contains no more predictive information than the present alone. We have just been looking at examples for which this doesn’t at face value seem to be true, since in a parachain, which is a Markov process, the digram state on X_{n-1} and X_n definitely has more predictive power than the probability distribution on X_n alone. The resolution of this seeming paradox lies in the fact that the transformation matrix is to some extent arbitrary. Any Markov chain, even a parachain, can be analyzed as an initial state to which we apply a succession of transition matrices T_1, T_2, T_3 etc. Given all of these transition matrices, then the states prior to any time t indeed do not supply predictive information for the states after t. However, and here is the key point, in a parachain all of these transition matrices are different, whereas the digram transformation matrix T, which is not a transition matrix, remains fixed. The digram dynamical law is given by T alone, but we would need a second dynamical law governing the changing T_i if we insisted on using transition matrices to calculate the changing single-variable states. Neither way of describing the chain dynamics is “truer” than the other, but the way that keeps T constant is essentially simpler.

Galileo and Newton described a falling object as being subject to a constant force that produces a constant acceleration. To do so requires that velocity belong to its state. The Aristotelian account of a falling body, which also makes force into the agency of change, identifies the state with position alone. Since the velocity of the falling object changes, for this to work the force pushing it down must change too. There is no logical contradiction in such an account, and indeed Kepler at one point tried to calculate the “forces” that push the planets around in their orbits. But the Newtonian account, with its constant force that operates to change velocity, has the virtue of much greater simplicity and unity in this and a wide diversity of other situations. If we take the “force” of change in a Markov process to be the transformation matrix T, having that T remain constant is much simpler and cleaner than having to deal with a changing T_i, as we must if we ignore the “velocity” component of G. My biggest hope for the new revolution, or rather for the continuation of the Newtonian revolution, is that its way of analyzing statistical change can make the kind of difference in our understanding of the life world that the Newtonian way made in our understanding of matter in motion.

References.

1. *Process, System Causality and Quantum Mechanics* by Tom Etter and H. Pierre Noyes, Physics Essays Dec. 1999
2. *An Introduction to Probability Theory and its Applications* by William Feller, John Wiley & Sons, 1950
3. *On the Occurrence of some Familiar Processes Reversed in Time* by Tom Etter, 1960, www.boundaryinstitute.org
4. *Structure Theory* by Tom Etter, 1999, www.boundaryinstitute.org
5. *Relation Arithmetic Revived* by Tom Etter, 1999, www.boundaryinstitute.org
6. *Statistical Physics and the Atomic Theory of Matter* by Stephen G. Brush, Princeton University Press, 1958, p.20
7. *Digram States in Markov Parachains* by Tom Etter, www.boundaryinstitute.org