# Cause and Effect:

## Modeling cause, effect, and time in a computer.


Version 1.0, 22 May 2018, Springe De.

By Christopher Rehm, CSEG, MBA, © 2018

I can be contacted at jfs@tandkcybernetics.christopherrehm.de

Links and further contact info including Impressum are available at:

www.tandkcybernetics.christopherrehm.de





### Abstract:

A paper describing a simple system of symbolic and mathematical notation for describing cause and effect relationships to an artificial intelligence system, and for describing said relationships in general. This system is designed to be implemented using linear algebra techniques and the concept of operators acting on matrices as in QM theory. Implementation of said system is not discussed in this paper, only various systems of representation of the basic concepts.

## First Principles.

Let T = the vector time, which only flows in one direction.

Is T a vector or is it a scalar sum? $T_{now} = T_{earlier} + 1$? Depending on the situation it can be thought of as both.

Now let A be any actor in a space. An actor is defined as any agent that causes effects to other agents. Actions produce effects on other agents.

All things are agents. People are agents, matter is an agent, radio signals are an agent etc. They have different effects on different ways on different things.

Agents can produce actions that have effects on other agents. The cause is either the nature of the agent or the decision of the agent. Note that the agent does not have to be human. A computer can make a decision and it becomes an agent capable of decision making at some level.

Actions take some period of time t in the flow of time T to occur. Actions cannot be reversed, only new actions can be applied. Restoring a complicated system to its original state by this means is difficult if not impossible.

Actions are designated by $A\rightarrow$.

The terms actors and agents are taken to be the same in this paper.

Actors by the symbol G.

Actions have effects. Effects are represented by the matrix $E_{AG}$. This notation allows us to know what action from what actor caused the given effect, even if two effects are identical their causes may not be.

The actor enters a new state $G_{tn}$ ► $G_{tn+1}$

so G1 →i ( implements) A1 on G2 and A1→ $E1_{AG}$ →p (produces) $G2_m$ ► $G2_{m+1}$

or

$G2_{new}$ ( = $G2_{m+1}$)

or

$$\mathbf{G1 \rightarrow i\ \ A1 \blacktriangleright G2\ \ \blacktriangledown\ A1 \rightarrow E1_{AG}\ \ \rightarrow p\ \ G2_{new}} \qquad \mathbf{(Fml1)}$$

The symbol ▼ is used to show that the action produces the effect and that is separate from the action of the actor.

This is the basic equation of cause and effect, that all other algorithms must be based on.

Both A1 and E1 take some time t to manifest themselves. Sometimes interrupting this can stop the effect E1 from occurring and some times not.

I define these as *interruptible events* and *non interruptible events.*

As implicit in the above, actions, actors and effects can be repented by matrices of characteristics. This should allow computation of outcomes more easily.

## Time:

Time is taken to be a vector that always travels in one direction in Newtonian space. The direction is assumed to be positive. It is not allowed to do anything but add to the time line, subtraction is forbidden by the laws of Newtonian space.

T future is always T now plus some increment Incr.

T flows regardless of computing etc. T must be accounted for in decision making.

Time can be an actor, but only as a passage of time, defined as:

Tdiff = T2 -T1

Tdiff is a special type of actor, with no other action than its change or passage.

Tdiff can further be reduced to Td and and index number, e.g. Td0001 Td30543 etc.

The equation becomes:

$$\begin{array}{l} \qquad\qquad\quad \mathbf{Td1}\downarrow \qquad \mathbf{Td2}\downarrow \\ \mathbf{G1} \rightarrow_{\mathbf{i}} \mathbf{A1} \blacktriangleright \mathbf{G2} \blacktriangledown \mathbf{A1} \rightarrow \mathbf{E1_{AG}} \rightarrow_{\mathbf{p}} \mathbf{G2_{new}} \end{array} \qquad \text{(Fml2)}$$

Where Tdiff1 and Tdiff2 are the times it takes for an action to occur and for the effect to manifest itself

these times may be either co-incidental or not, and may have similar or different start and end times.

For simplicity it is suggested that this be reduced to:

$$\mathbf{G1 \rightarrow A1(Td1) \rightarrow E1(A1,G2,Td2) \rightarrow G2(new)} \qquad \text{(Fml3)}$$

for illustrative purposes,

or even:

$$\mathbf{G(N)A(N)E(N)G(N)} \qquad \text{(Fml4)}$$

A(N)E(N) can further be represented by CE(N)

reducing to

**G(N)CE(N)G(N) (Fml5)**

This allows easy construction of DAG like diagrams. Time should flow from left to right as is conventional in math and most languages.

In diagrammatic representation things are similar to a flow chart but should work from left to right instead of from top to bottom.

Squares are agents.

Diamonds are actions.

Circles are effects.

Trapezoids can be used to represent Tdiff. I would recommend using one color for time and using only for time, not for anything else.

This allows simple representation of complex cause and effect situations over time graphically for the developer. Color can be used creatively to differentiate  various actors, actions etc.

## Actors and Goals:

Actors create actions that have effects on other actors. Intelligent actors decide on those actions based on Goals. Goals determine actions, or a series of actions that will hopefully produce a desired outcome.

$Z(N)_{G(n)}$ is goal N of actor N. (Z is taken from German das Ziel, the goal)

Each intelligent actor has some collection of goals, C(Z1,Z2,Z3….ZN). Non intelligent actors have no goals, they simply cause actions without any sentient decision making process involved.

We define P(N)(Z(N)) as the process by which actor N achieves Goal N. P can be composed of as many actions as needed. Other actors may influence P, and P may influence other actors. The Goal should be in line with whatever general principles of behavior the creating entity has determined are appropriate for the Actor. Various schemes have been suggested including Asimov's 3 Laws of Robotics and our own Ten Commandments for Non-Organic Beings. Since any AI will eventually establish its own goals it must have something to measure those goals against, to determine if they meet more basic considerations.

We can also think of some processes as being natural. We shall designate these as Pn(N)

Let our general principles be described by the term $\Omega_{G(N)}$

then the set of Z for G(N) should be in line with the general principles $\Omega_{G(N)}$

so we have

$$C(G(N)) \text{ where } C(G(N)) = \sum Z(N) \qquad \text{(Fml6)}$$

And

$$C(G(N)) = \sum Z(N) \equiv \in \Omega(G(N)) \qquad \text{(Fml7)}$$

## Summation:

The above outlines a logical and mathematical structure for the development of algorithms and thinking regarding cause and effect relationships. Further development will explore the addition of probabilistic learning on this system of logic and other effects as well as implementation.

With Acknowledgements to Mr Philip Peterson, Bsc Phys, Msc Phys, for his help developing the simplest possible annotations for the math ideas.