Julie A. Dickerson
and
Bart Kosko
Department of Electrical
Engineering—Systems
Signal and Image Processing Institute
University of Southern California
Los Angeles, California 90089-2564

Virtual Worlds as Fuzzy Cognitive Maps

Abstract

Fuzzy cognitive maps (FCM) can structure virtual worlds that change with time. An FCM links causal events, actors, values, goals, and trends in a fuzzy feedback dynamical system. An FCM lists the fuzzy rules or causal flow paths that relate events. It can guide actors in a virtual world as the actors move through a web of cause and effect and react to events and to other actors. Experts draw FCM causal pictures of the virtual world. They do not write down differential equations to change the virtual world. Complex FCMs can give virtual worlds with "new" or chaotic equilibrium behavior. Simple FCMs give virtual worlds with periodic behavior. They map input states to limit-cycle equilibria. An FCM limit cycle repeats a sequence of events or a chain of actions and responses. Limit cycles can control the steady-state rhythms and patterns in a virtual world. In nested FCMs each causal concept can control its own FCM or fuzzy function approximator. This gives levels of fuzzy systems that can choose goals and causal webs as well as move objects and guide actors in the webs. FCM matrices sum to give a combined FCM virtual world for any number of knowledge sources. Adaptive FCMs change their fuzzy causal web as causal patterns change and as actors act and experts state their causal knowledge. Neural learning laws change the causal rules and the limit cycles. Actors learn new patterns and reinforce old ones. In complex FCMs the user can choose the dynamical structure of the virtual world from a spectrum that ranges from mildly to wildly nonlinear. We use a simple but adaptive FCM to model an undersea virtual world of dolphins, fish, and sharks.

1 Fuzzy Virtual Worlds

What is a virtual world? It is what changes in a "virtual reality" (Krueger, 1991) or "cyberspace" (Gibson, 1984). A virtual world links humans and computers in a causal medium that can trick the mind or senses.

At the broadest level a virtual world is a dynamical system. It changes with time as the user or an actor moves through it. In the simplest case only the user moves in the virtual world. In general both the user and the virtual world change and they change each other.

Change in a virtual world is causal. Actors cause events to happen as they move in a virtual world. They add new patterns of cause and effect and respond to old ones. In turn the virtual world acts on the actors or on their physical or social environments. The virtual world changes their behavior and can change its own web of cause of effect. This feedback causality between actors and their virtual world makes up a complex dynamical system that can model events, actors, actions, and data as they unfold in time.

Virtual worlds are fuzzy as well as fedback. Events occur and concepts hold

Presence, Vol. 3, No. 2, Spring 1994, 173–189

only to some degree. Events cause one another to some degree. In this sense virtual worlds are fuzzy causal worlds. They are fuzzy dynamical systems.

How do we model the fuzzy feedback causality? One way is to write down the differential equations that show how the virtual "flux" or "fluid" changes in time. This gives an exact model. The Navier–Stokes equations (Brown, 1991) used in weather models give a fluid model of how actors move in a type of virtual world. They can show how clouds or tornadoes form and dissolve in a changing atmosphere or how an airplane flies through pockets of turbulence. The inverse kinematic equations of robotics (Craig, 1986) show how an actor moves through or grasps in a virtual joint space. The coupled differential equations of blood glucose and insulin (Ackerman, Gatewood, Rosevear, & Molner, 1969) cast the patient as a diabetic actor awash in a virtual world of sugar and hormones. Such math models are hard to find, hard to solve, and hard to run in realtime. They paint too fine a picture of the virtual world.

Fuzzy cognitive maps (FCMs) can model the virtual world in large fuzzy chunks. They model the causal web as a fuzzy directed graph (Kosko, 1986, 1988a). The nodes and edges show how causal concepts affect one another to some degree in the fuzzy dynamic system. The "size" of the nodes gives the chunk size. The causal concept node SURVIVAL THREAT can measure the degree that the fuzzy event of survival threat occurs. In a virtual world the concept nodes can stand for events, actions, values, moods, goals, or trends. The causal edges state fuzzy rules or causal flows between concepts. In a predator–prey world survival threat increases prey runaway. The degree of runaway grows or falls as the degree of threat grows or falls. The fuzzy rule states how much one node grows or falls as some other node grows or falls.

Experts draw the FCMs as causal pictures. They do not state equations. They state concept nodes and link them to other nodes. The FCM system turns each picture into a matrix of fuzzy rule weights. The system weights and adds the FCM matrices to combine any number of causal pictures. More FCMs tend to sum to a better picture of the causal web with rich tangles of feedback and fuzzy edges even if each expert gives binary (present or absent) edges. This makes it easy to add or delete actors or to change the background of a virtual world or to combine virtual worlds that are disjoint or overlap. We can also let a FCM node control its own FCM to give a nested FCM in a hierarchy of virtual worlds. The node FCM can model the complex nonlinearities between the node's input and output. It can drive the motions, sounds, actions, or goals of a virtual actor as in Figure 1.

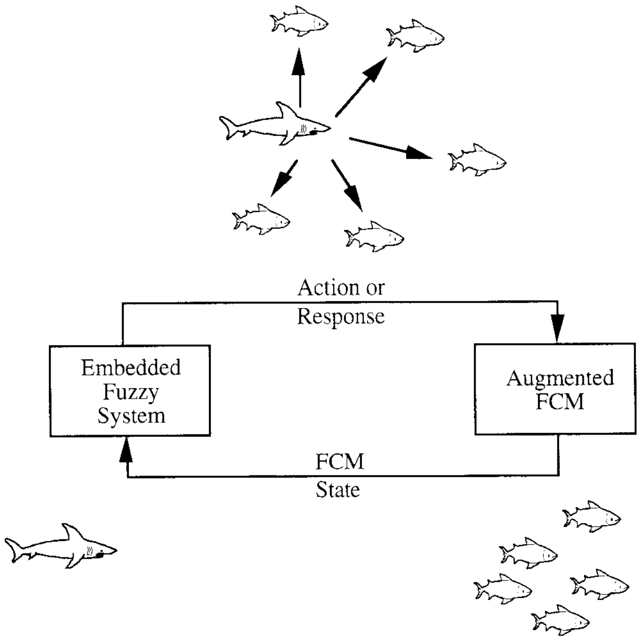


Figure 1. *Fuzzy cognitive maps can structure virtual worlds. Embedded fuzzy systems drive lower level fuzzy systems for animation, sounds, and other virtual world outputs. Here a shark finds a school of fish. The shark attacks and the fish flee.*

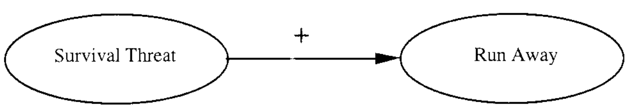
The FCM itself acts as a nonlinear dynamical system. Like a neural net it maps inputs to output equilibrium states. Each input digs a path through the virtual state space. In simple FCMs the path ends in a fixed point or limit cycle. In more complex FCMs the path may end in an aperiodic or "chaotic" attractor. We illustrate the FCM technique with a simple FCM that learns and converges to a binary limit cycle.

In contrast an AI expert system (Winston, 1984) models a system as a binary rule tree with graph search. Each input fires one rule or a few rules and the search spreads down the tree branch to a leaf or leaves. Reactive systems (Kaelbling, 1987) choose from a set of precompiled actions in the tree. This means that the virtual world designer must anticipate what action to select under all possible conditions. Situated agents with goals can also help choose actions in diverse situations (Maes, 1990). Each of these systems keeps the feedforward tree structure. The lack of feedback loops allows the tree search. But each serial inference uses only a small part of the stored knowledge. Each FCM input fires all the rules to some degree. The causal "juice" swirls through the tangles of fuzzy feedback and equilibrates in a global system response. In this way FCMs model the "circular causality" (Minsky, 1985) of real and virtual worlds.

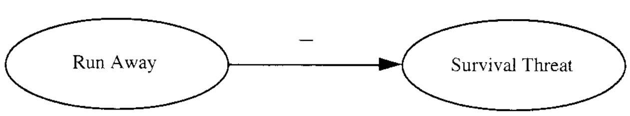
2 Fuzzy Cognitive Maps

Fuzzy cognitive maps (FCMs) are fuzzy signed digraphs with feedback (Kosko, 1986, 1988a). Nodes stand for fuzzy sets or events that occur to some degree. The nodes are causal concepts. They can model events, actions, values, goals, or lumped-parameter processes.

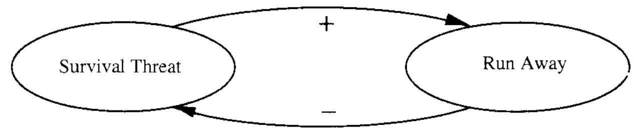
Directed edges stand for fuzzy rules or the partial causal flow between the concepts. The sign (+ or −) of an edge stands for causal increase or decrease. The positive edge rule



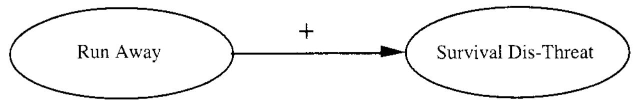
states that a survival threat increases runway. It is a positive causal connection. The runaway response grows or falls as the threat grows or falls. The negative edge rule



states that running away from a predator decreases the survival threat. It is a negative causal connection. The survival threat grows the less the prey runs away and falls the more the prey runs away. The two rules define a minimal feedback loop in the FCM causal web:



We can replace all negative rules with a positive rule if we double the number of concept nodes to add disconcepts (Kosko, 1986) like SURVIVAL DIS-THREAT:



Disconcepts keep the FCM edge math in the standard fuzzy framework of concept values in the unit interval [0, 1]. We shall work with edge values in [−1, 1].

An FCM with n nodes has n^2 edges. The nodes $C_i(t)$ are fuzzy sets and so take values in [0, 1]. So an FCM state is the *fit* (fuzzy unit) vector $\mathbf{C}(t) = [C_1(t), \ldots, C_n(t)]$ and thus a point in the fuzzy hypercube $I^n = [0, 1]^n$. An FCM inference is a path or point sequence in I^n. It is a fuzzy process or indexed family of fuzzy sets $\mathbf{C}(t)$. The FCM can only "forward chain" (Winston, 1984) to answer what-if questions. Nonlinearities do not permit reverse causality. FCMs cannot "backward chain" to answer why questions.

The FCM nonlinear dynamical system acts as a neural network. For each input state $\mathbf{C}(0)$ it digs a trajectory in I^n that ends in an equilibrium attractor $\mathbf{A}$. The FCM quickly converges or "settles down" to a fixed point, limit cycle, limit torus, or chaotic attractor in the fuzzy cube.

The output equilibrium is the answer to a causal what-if question: What if $\mathbf{C}(0)$ happens? In this sense each FCM stores a set of global rules of the form "If $\mathbf{C}(0)$, then equilibrium attractor $\mathbf{A}$."

The size of the attractor regions in the fuzzy cube governs the number of these global rules or "hidden patterns" (Kosko, 1988a). All points in the attractor region map to the attractor. An FCM with a global fixed point

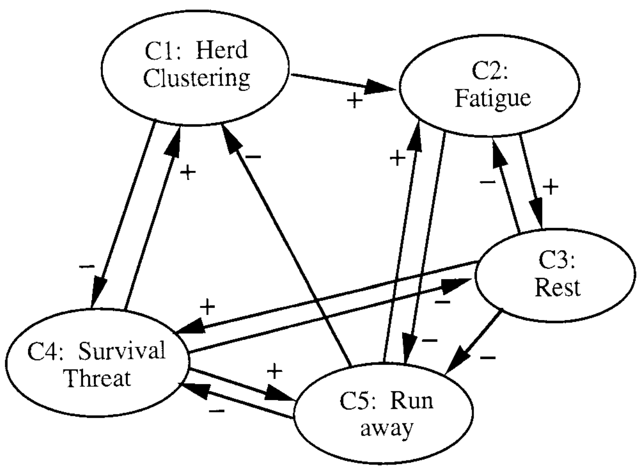


Figure 2. *Simple FCM with five concept nodes. Edges show directed causal flow between nodes*

has only one global rule. All input balls "roll" down its "well." FCMs can have large and small attractor regions in the fuzzy cube. The attractor types can vary in complex FCMs with highly nonlinear concepts and edges. Then one input state may lead to chaos and a more distant input state may end in a fixed point or limit cycle.

2.1 Simple FCMs

Simple FCMs have bivalent nodes and trivalent edges. Concept values C_i take values in $\{0, 1\}$. Causal edges take values in $\{-1, 0, 1\}$. So for a concept each simple FCM state vector is one of the 2^n vertices of the fuzzy cube I^n. The FCM trajectory hops from vertex to vertex. I^n ends in a fixed point or limit cycle at the first repeated vector.

We can draw simple FCMs from articles, editorials, or surveys. A delphi process can also produce the FCMs (Martino, 1972). Most persons can state the sign of causal flow between nodes. The hard part is to state its degree or magnitude. We can average expert responses (Kosko, 1988a; Taber & Siegel, 1987) as in Equation (2.3) below or use neural systems to learn fuzzy edge weights from data. The expert responses can initialize the causal learning or modify it as a type of forcing function.

Figure 2 shows a simple FCM with five concept nodes. The connection or edge matrix **E** lists the causal links between nodes:

$$\mathbf{E} = \begin{array}{c} \\ C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \end{array} \begin{array}{c} \begin{array}{ccccc} C_1 & C_2 & C_3 & C_4 & C_5 \end{array} \\ \left[\begin{array}{ccccc} 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 \\ 0 & -1 & 0 & 1 & -1 \\ 1 & 0 & -1 & 0 & 1 \\ -1 & 1 & 0 & -1 & 0 \end{array}\right] \end{array}$$

The ith row lists the connection strength of the edges e_{ik} directed out from causal concept C_i. The ith column lists the edges e_{ki} directed into C_i. C_i causally increases C_k if $e_{ik} > 0$, decreases C_k if $e_{ik} < 0$, and has no effect if $e_{ik} = 0$. The causal concept C_4 causally increases concepts C_1 and C_5. It decreases C_3. Concepts C_1 and C_5 decrease C_4. Concept C_3 increases C_4.

2.2 FCM Recall

FCMs recall as the FCM dynamic system equilibrates. Simple FCM inference thresholds a matrix–vector multiplication (Kosko, 1988a,b). State vectors $\mathbf{C}_n$ cycle through the FCM adjacency matrix **E**: $\mathbf{C}_1 \rightarrow \mathbf{E} \rightarrow \mathbf{C}_2 \rightarrow \mathbf{E} \rightarrow \mathbf{C}_3 \rightarrow \ldots$. The system nonlinearly transforms the weighted input to each node C_i:

$$C_i(t_{n+1}) = S[\sum_{k=1}^{N} e_{ki}(t_n)C_k(t_n)] \qquad (2.1)$$

Here $S(x)$ is a bounded signal function. For simple FCMs the sigmoid function

$$S(y) = \frac{1}{1 + e^{-c(y-T)}} \qquad (2.2)$$

with large $c > 0$ approximates a binary threshold function.

Simple threshold FCMs quickly converge to stable limit cycles or fixed points (Kosko, 1988a,b). These limit cycles show "hidden patterns" in the causal web of the FCM. The FCM in Figure 2 gives a three-step limit cycle when input state $\mathbf{C}_1 = [0\ 0\ 0\ 1\ 0]$ fires the FCM network. Equation (2.1) and binary thresholding gives

$$\mathbf{C}_1 = [\ 0 \quad 0 \quad 0 \quad 1 \quad 0]$$

$$\begin{aligned}
\mathbf{C}_1\,\mathbf{E} &= [\ 1 \quad 0 \quad -1 \quad 0 \quad 1] \rightarrow \mathbf{C}_2 = [1 \quad 0 \quad 0 \quad 0 \quad 1] \\
\mathbf{C}_2\,\mathbf{E} &= [-1 \quad 2 \quad 0 \quad -2 \quad 0] \rightarrow \mathbf{C}_3 = [0 \quad 1 \quad 0 \quad 0 \quad 0] \\
\mathbf{C}_3\,\mathbf{E} &= [\ 0 \quad 0 \quad 1 \quad 0 \quad -1] \rightarrow \mathbf{C}_4 = [0 \quad 0 \quad 1 \quad 0 \quad 0] \\
\mathbf{C}_4\,\mathbf{E} &= [\ 0 \quad -1 \quad 0 \quad 1 \quad -1] \rightarrow \mathbf{C}_1 = [0 \quad 0 \quad 0 \quad 1 \quad 0]
\end{aligned}$$

the four-step limit cycle $\mathbf{C}_1 \rightarrow \mathbf{C}_2 \rightarrow \mathbf{C}_3 \rightarrow \mathbf{C}_4 \rightarrow \mathbf{C}_1$ shown above. In a virtual world the limit cycle might make in order wake up, go to work, come home, then wake up again. Some complex actions such as walking break down into simple cycles of movement (Brooks, 1989).

Each node in a simple FCM turns actions or goals on and off. Each node can control its own FCM, fuzzy control system, goal-directed animation system, force feedback, or other input–output map. The FCM can control the temporal associations or timing cycles that structure virtual worlds. These patterns establish the rhythm of the world. "Grandmother" nodes can control the time spent on each step in a FCM "avalanche" (Grossberg, 1982). This can change the update rate and thus the timing for the network (Grossberg, 1982).

2.3 Augmented FCMs

FCM matrices additively combine to form new FCMs (Kosko, 1986). This allows a combination of FCMs for different actors or environments in the virtual world. The new (augmented) FCM includes the union of the causal concepts for all the actors and the environment in the virtual world. If an FCM does not include a concept, then those rows and columns are all zero. The sum of the augmented (zero-padded) FCM matrices for each actor forms the virtual world:

$$\mathbf{F} = \sum_{i=1}^{n} w_i\, \mathbf{F}_i \tag{2.3}$$

The w_i are positive weights for the ith FCM $\mathbf{F}_i$. The weights state the relative value of each FCM in the virtual world and can weight any subgraph of the FCM. Figure 3a shows three simple FCMs that might come from a simple delphi process (Martino, 1972). Equation (2.3) combines these FCMs to give the new simple FCM in Figure 3b that has fuzzy or multivalued edges as shown in equation (2.4) below.

$$\mathbf{F} = \frac{1}{3}(\mathbf{F}_1 + \mathbf{F}_2 + \mathbf{F}_3) = \frac{1}{3}\begin{bmatrix} 0 & 2 & -1 & 0 & 0 & -1 \\ 0 & 0 & 2 & 3 & 1 & 0 \\ 0 & 0 & 0 & 2 & 1 & 0 \\ 2 & 0 & 0 & 0 & -1 & 0 \\ 0 & -2 & 0 & 0 & 0 & 0 \\ 1 & -1 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{2.4}$$

The FCM sum (2.3) helps knowledge acquisition. Any number of experts can describe their FCM virtual world views and (2.3) will weight and combine them. In contrast an AI expert system (Winston, 1984) is a binary tree with graph search. Two or more trees need not combine to a tree. Combined FCMs tend to have feedback or closed loops and that precludes graph search with forward or backward "chaining." The strong law of large numbers (Kosko, 1988a) ensures that the knowledge estimate $\mathbf{F}$ in (2.3) improves with the expert sample size n if we view the experts as independent (unique) random knowledge sources with finite variance

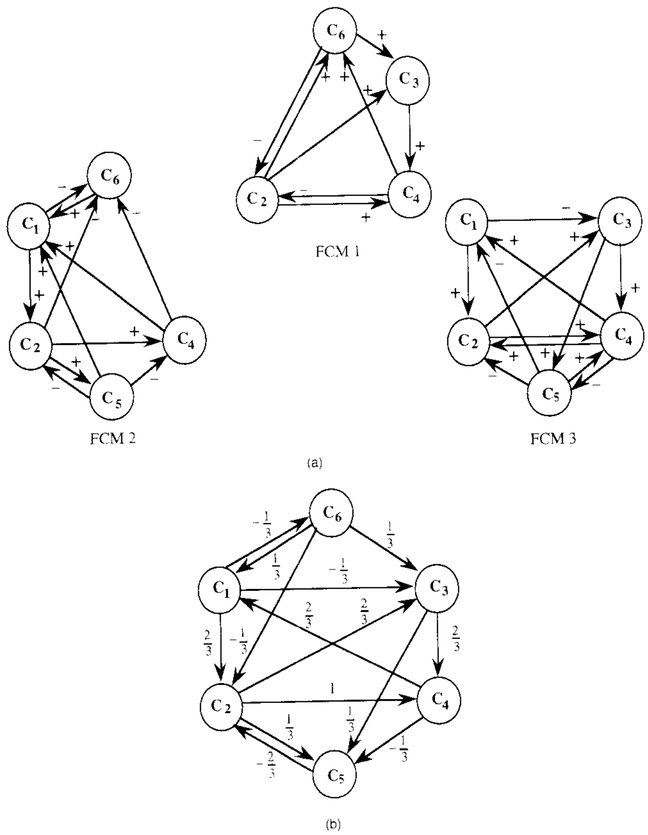


Figure 3. *FCMs combine additively. (a) Three bivalent FCMs. (b) Augmented FCM. The augmented FCM takes the union of the causal concepts of the smaller FCMs and sums the augmented connection matrices: F = 1/3 (F + F + F).*

(bounded uncertainty) and identical distribution (same problem-domain focus). The sample FCM converges to the unknown population FCM as the number of experts grows.

2.4 Nested FCMs

FCMs can endow virtual worlds with goals and intentions as they define dynamic physical and social environments. This can give the "common representation" needed for a virtual world (Badler, Webber, Kalita, & Esakov, 1991). The FCM can combine simple actions to model "intelligent" behavior (Brooks, 1989; Connell, 1990). Each node in turn can control its own simple FCM in a *nested* FCM. Complex actions such as walking emerge from networks of simple reflexes. Nested simple FCMs can mimic this process as a net of finite state machines with binary limit cycles.

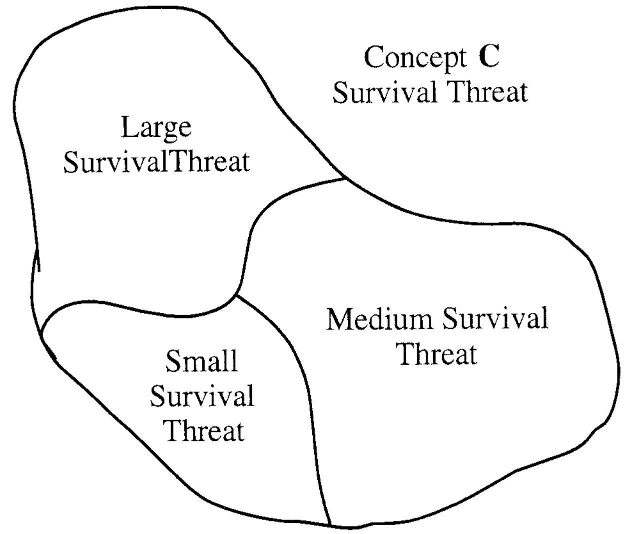


Figure 4. *Nested FCMs divide a concept into subconcepts.*

The output of a simple FCM is a binary limit cycle that describes actions or goals. This holds even if the binary concept nodes change state asynchronously. Each output turns a function on or off as in a robotic neural net (Brooks, 1989). This output can control smaller FCMs or fuzzy control systems. These systems can drive visual, auditory, or tactile outputs of the virtual world. The FCM can control the temporal associations or timing cycles that structure virtual worlds. The FCM state vector drives the motion of each character as in a frame in a cartoon. Simple equations of motion can move each actor between the states.

FCM nesting extends to any number of fuzzy sets for the inputs. A concept can divide into smaller fuzzy sets or subconcepts. The edges or rules link the sets. This leads to a discrete multivalued output for each node. Enough nodes allow this feedforward system to approximate any continuous function (Hornik, Stinchcombe, & White, 1989) for signal functions of the form (2.2). The subconcepts Q_{ij} partition the fuzzy concept $\mathbf{C}_j$

$$\mathbf{C}_j = \bigcup_{i=1}^{N_j} Q_{ij} \quad (2.5)$$

Figure 4 shows the concept of a SURVIVAL THREAT divided into subconcepts. Each subconcept is the degree of threat.

The FCM edges or rules map one subconcept to another. These subconcept mappings form a fuzzy system or set of fuzzy if-then rules that map inputs to outputs. Each mapping is a fuzzy rule or state-space patch that links fuzzy sets. The patches cover the graph of some function in the input–output state space. The fuzzy system then averages the patches that overlap to give an approximation of a continuous function (Kosko, 1993). Figure 5 shows how subconcepts can map to different responses in the FCM. This gives a more varied response to changes in the virtual world.

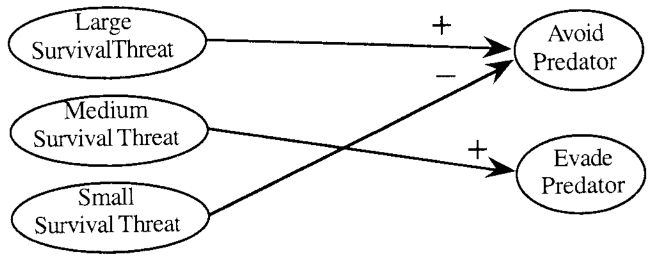


Figure 5. *Subconcepts map to other concepts. This gives a more varied response.*

3 Virtual Undersea World

Figure 6 shows a simple FCM for a virtual dolphin. It lists a causal web of goals and actions in the life of a dolphin (Shane, 1990). The connection matrix $\mathbf{E}_D$ states these causal relations in numbers as shown in equation (3.1) below.

$$
\mathbf{E}_D = \begin{array}{c} \\ D_1 \\ D_2 \\ D_3 \\ D_4 \\ D_5 \\ D_6 \\ D_7 \\ D_8 \\ D_9 \\ D_{10} \end{array}
\begin{array}{c} \begin{array}{cccccccccc} D_1 & D_2 & D_3 & D_4 & D_5 & D_6 & D_7 & D_8 & D_9 & D_{10} \end{array} \\
\left[\begin{array}{cccccccccc}
0 & -1 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & -1 & -1 & 0 & 0 & -1 \\
1 & 0 & -1 & 0 & 0 & -1 & -1 & 0 & 0 & -1 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\
0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
-1 & 1 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 1 & -1 & -1 & -1 & 0 & 1 \\
-1 & -1 & 1 & 0 & -1 & -1 & -1 & -1 & -1 & 0
\end{array}\right] \end{array} \qquad (3.1)
$$

The ith row lists the connection strength of the edges e_{ik} directed out from causal concept D_i and the ith column lists the edges e_{ki} directed into D_i. Row 9 shows how the concept SURVIVAL THREAT changes the other concepts. Column 9 shows the concepts that change SURVIVAL THREAT.

We can model the effect of a survival threat on the dolphin FCM as a sustained input to D_9. This means $D_9 = 1$ for all time t_k. $\mathbf{C}_0$ is the initial input state of the dolphin FCM:

$$\mathbf{C}_0 = [0 \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; 1 \;\; 0]$$

Then $\mathbf{C}_0\mathbf{E}_D \rightarrow \mathbf{C}_1$ where the arrow stands for a threshold operation with ½ as the threshold value. $\mathbf{C}_1$ keeps D_9 on since we want to study the effect of a sustained threat.

$\mathbf{C}_1$ shows that when threatened the dolphins cluster in a herd and flee the threat. The negative rules in the ninth row of $\mathbf{E}_D$ show that a threat to survival turns off other actions. The FCM converges to the limit cycle $\mathbf{C}_1 \rightarrow$

$\mathbf{C}_2 \rightarrow \mathbf{C}_3 \rightarrow \mathbf{C}_4 \rightarrow \mathbf{C}_5 \rightarrow \mathbf{C}_1 \ldots$ if the threat lasts as shown in equation (3.2) below.

$$
\begin{array}{llllllllllllllllllllll}
\mathbf{C}_0\,\mathbf{E}_D = [& 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & 0 & 1] \rightarrow \mathbf{C}_1 = [0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1] \\
\mathbf{C}_1\,\mathbf{E}_D = [& -1 & -1 & 2 & 0 & 0 & -2 & -2 & -2 & -2 & 1] \rightarrow \mathbf{C}_2 = [0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1] \\
\mathbf{C}_2\,\mathbf{E}_D = [& -1 & -1 & 1 & 1 & 1 & -3 & -3 & -2 & -1 & 0] \rightarrow \mathbf{C}_3 = [0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0] \\
\mathbf{C}_3\,\mathbf{E}_D = [& 1 & 0 & 0 & 1 & 2 & -3 & -3 & -1 & -1 & -1] \rightarrow \mathbf{C}_4 = [1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0] \\
\mathbf{C}_4\,\mathbf{E}_D = [& 1 & -1 & -1 & 0 & 1 & -1 & -2 & -1 & -1 & 0] \rightarrow \mathbf{C}_5 = [1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0] \\
\mathbf{C}_5\,\mathbf{E}_D = [& 0 & -1 & 0 & 0 & 1 & 0 & -1 & -1 & -1 & 1] \rightarrow \mathbf{C}_1 = [0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1]
\end{array}
\qquad (3.2)
$$

Flight causes fatigue ($\mathbf{C}_2$). The dolphin herd stops and rests staying close together ($\mathbf{C}_3$). All the activity causes hunger ($\mathbf{C}_4$, $\mathbf{C}_5$). If the threat persists, they again try to flee ($\mathbf{C}_1$). A threat suppresses hunger. This limit cycle shows a "hidden" global pattern in the causal virtual world.

The FCM converges to the new limit cycle $\mathbf{C}_6 \rightarrow \mathbf{C}_7 \rightarrow \mathbf{C}_8 \rightarrow \mathbf{C}_9 \rightarrow \mathbf{C}_{10} \rightarrow \mathbf{C}_{11} \rightarrow \mathbf{C}_{12} \rightarrow \mathbf{C}_{13} \rightarrow \mathbf{C}_6 \ldots$ when the shark gives up the chase or eats a dolphin and the threat ends ($D_9 = 0$) as in equation (3.4) below.

$$
\begin{array}{llllllllllllllllllllll}
\mathbf{C}_6 = [& 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0] & \\
\mathbf{C}_6\,\mathbf{E}_D = [& 1 & 0 & 0 & 1 & 1 & -2 & -2 & 0 & -1 & -2] \rightarrow \mathbf{C}_7 = [1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0] \\
\mathbf{C}_7\,\mathbf{E}_D = [& 1 & -1 & -1 & 0 & 0 & 0 & -1 & 0 & -1 & -1] \rightarrow \mathbf{C}_8 = [1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0] \\
\mathbf{C}_8\,\mathbf{E}_D = [& 0 & -1 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0] \rightarrow \mathbf{C}_9 = [0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0] \\
\mathbf{C}_9\,\mathbf{E}_D = [& 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0] \rightarrow \mathbf{C}_{10} = [0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0] \\
\mathbf{C}_{10}\mathbf{E}_D = [& 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0] \rightarrow \mathbf{C}_{11} = [0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0] \\
\mathbf{C}_{11}\mathbf{E}_D = [& -1 & 1 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0] \rightarrow \mathbf{C}_{12} = [0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0] \\
\mathbf{C}_{12}\mathbf{E}_D = [& 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & -1 & 0] \rightarrow \mathbf{C}_{13} = [0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0] \\
\mathbf{C}_{13}\mathbf{E}_D = [& 0 & 0 & 1 & 1 & 1 & -1 & -1 & 0 & -1 & -1] \rightarrow \mathbf{C}_6 = [0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0]
\end{array}
\qquad (3.4)
$$

The dolphin herd rests from the previous chase ($\mathbf{C}_6$, $\mathbf{C}_7$). Then they begin a hunt of their own ($\mathbf{C}_9$, $\mathbf{C}_{10}$). They eat ($\mathbf{C}_{11}$) and then they socialize and rest ($\mathbf{C}_{12}$, $\mathbf{C}_{13}$, $\mathbf{C}_6$). This makes them hungry and the feeding cycle repeats.

3.1 Augmented Virtual World

Figure 7 shows an augmented FCM for an undersea virtual world. It combines fish school, shark, and dolphin herd FCMs with (2.3): $\mathbf{F} = \mathbf{F}_{\text{fish}} + \mathbf{F}_{\text{shark}} + \mathbf{F}_{\text{dolphin}}$. The new links among these FCMs are those of predator and prey where the larger eats the smaller. The actors chase, flee, and eat one another. A hungry shark chases the dolphins and that leads to the limit cycle ($\mathbf{C}_1$, $\mathbf{C}_2$, $\mathbf{C}_3$, $\mathbf{C}_4$) above. Augmenting the FCM matrices gives a large but sparse FCM since the actors respond to each other in few ways. Figure 8 shows the connection matrix for the augmented FCM in Figure 7.

The augmented FCM moves the actors in the virtual world. The binary output states of this FCM move the actors. Each FCM state maps to equations or function approximations for movement.

We used a simple update equation for position:

$$\mathbf{p}(t_{n+1}) = \mathbf{p}(t_n) + \Delta t\mathbf{v}(t_n) \qquad (3.3)$$

The velocity $\mathbf{v}(t)$ does not change at time step Δt. The FCM finds the direction and magnitude of movement. The magnitude of the velocity depends on the FCM state. If the FCM state is "run away," then the velocity is

FAST. If the FCM state is "rest," then the velocity is SLOW. The prey choose the direction that maximizes the distance from the predator. The predator chases the prey. When a predator searches for food it swims at random (Koopman, 1980). Each state moves the actors through the sea.

The FCM in Figure 8 encodes limit cycles between the actors. For example, if we start with a hungry shark. Then the first state $\mathbf{C}_1$ is as in equation 3.9. This vector gives a 10-step limit cycle as in equation 3.10.

In this limit cycle a shark searches for food ($\mathbf{C}_1$, $\mathbf{C}_2$). The shark finds some fish ($\mathbf{C}_3$), chases the fish ($\mathbf{C}_4$), and then eats some of the fish ($\mathbf{C}_5$). To avoid the shark most fish run away and then regroup as a school ($\mathbf{C}_4$, $\mathbf{C}_5$, $\mathbf{C}_6$). Then the fish rest and eat while the shark rests ($\mathbf{C}_7$, $\mathbf{C}_8$).

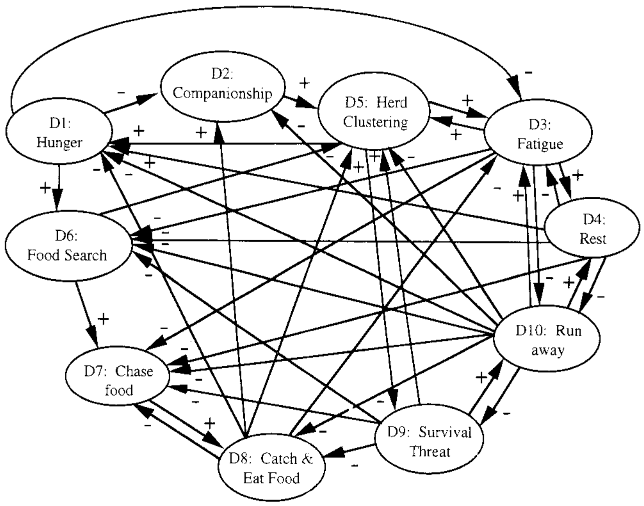


Figure 6. *Trivalent fuzzy cognitive map for the control of a dolphin actor in a fuzzy virtual world. The rules or edges connect causal concepts in a signed connection matrix.*

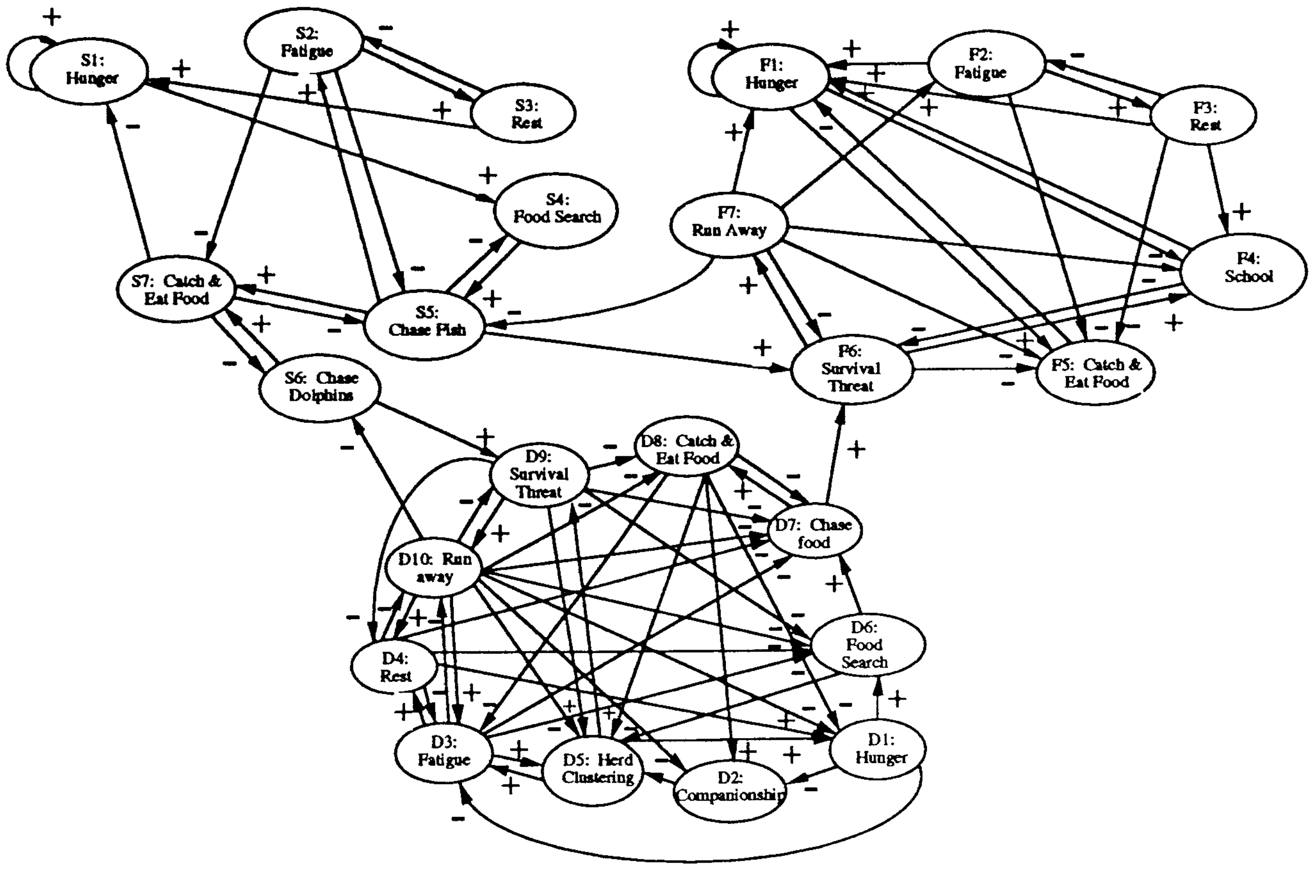


Figure 7. *Augmented FCM for different actors in a virtual world. The actors interact through linked common causal concepts such as chasing food and avoiding a threat.*

	Dolphin										Shark							Fish						
	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9	D_{10}	S_1	S_2	S_3	S_4	S_5	S_6	S_7	F_1	F_2	F_3	F_4	F_5	F_6	F_7
D_1	0	-1	-1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D_2	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D_3	0	0	0	1	1	-1	-1	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D_4	0	0	-1	0	1	-1	-1	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D_5	1	0	-1	0	0	0	0	0	-1	0	0	0	0	0	0	-1	-1	0	0	0	0	0	0	0
D_6	0	0	0	0	-1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D_7	0	0	0	0	1	-1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
D_8	-1	1	-1	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D_9	0	0	0	0	1	-1	-1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D_{10}	-1	-1	1	0	-1	-1	-1	-1	-1	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0
S_1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0
S_2	0	0	0	0	0	0	0	0	0	0	0	0	1	0	-1	-1	-1	0	0	0	0	0	0	0
S_3	0	0	0	0	0	0	0	0	0	0	1	-1	0	0	0	0	0	0	0	0	0	0	0	0
S_4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	-1	0	0	0	0	0	0	0
S_5	0	0	0	0	0	0	0	0	0	0	0	1	0	-1	0	0	1	0	0	0	1	0	1	1
S_6	0	0	0	0	0	0	0	0	1	0	0	1	0	-1	0	0	1	0	0	0	0	0	0	0
S_7	0	0	0	0	0	0	0	0	0	0	-1	1	0	-1	-1	-1	0	0	0	0	0	0	0	0
F_1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	-1	1	0	0
F_2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	-1	0	0
F_3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	-1	0	1	-1	0	0
F_4	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	-1	0
F_5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0
F_6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	-1	0	1
F_7	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	-1	0	0	1	1	0	-1	-1	-1	0

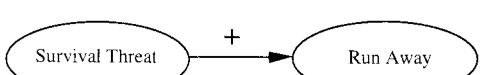
Figure 8. *Augmented FCM connection matrix for the dolphin herd, fish school, and shark. Figure 7 shows the nodes and edges. The lines show the FCMs of the actors. The sparse region outside the lines shows the interaction space of the FCMs.*

In time the shark gets hungry again and searches for fish ($\mathbf{C}_9$, $\mathbf{C}_{10}$, $\mathbf{C}_{11}$).

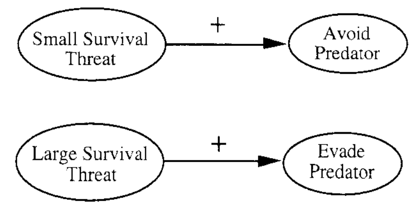
The result is a complex dance among the actors as they move in a 2-D ocean. Figure 9 shows these movements. The forcing function is a hungry shark ($\mathbf{C}_{11} = 1$). The shark encounters the dolphins who cluster and then flee the shark. The shark chases but cannot keep up. The shark still searches for food and finds the fish. It catches a fish and then rests with its hunger sated. Meanwhile the hungry dolphins search for food and eat more fish. Each actor responds to the actions of the other.

3.2 Nested FCMs for Fish Schools

In a simple FCM the threat response concepts link as a rule:



This rule can model the effects of different threats if we view it as a nested or embedded FCM:



This small survival threat may be a slow-moving predator that has not seen or decided to attack the fish. The large survival threat may be a fast predator such as a barracuda or shark that swims towards the center of the school. If we insert this new sub-FCM into the Fish FCM in Figure 7, we get the FCM in Figure 10. Different limit cycles appear for different degrees of threat. For a small threat (F_6) the fish avoid the predator (F_9) as they move out of the line-of-sight of the predator. Large threats (F_7) cause the fish to scatter quickly to evade the predator (F_8). This leads to fatigue and rest (F_2 and F_3).

A few simple spacing rules can model schooling behavior of fish (Reynolds, 1987). Each fish has a preferred angle and distance from the other fish in the school. The distance tends to be one body length (Partridge, 1982). The angle varies for each fish species.

Fish change their behavior as the degree of threat changes. The size of the threat is a function of the size, speed, and attack angle of the predator (Partridge, 1982). A small threat leads to avoidance behavior. Figure 11a shows how fish avoid a predator. The fish move in direction α to maximize their distance from the predator (Weihs & Webb, 1984):

$$\cot \alpha = \cot \alpha_m + \frac{V_p}{V_f \sin \alpha_m} \quad (3.5)$$

V_p and V_f are the velocities of the predator and the fish. α_m is the angle that minimizes the time in terms of the predator's sighting angle γ_p:

$$\tan \alpha_m = -\cot \gamma_p \quad (3.6)$$

A large threat causes the fish to evade the predator. The fish try to maximize the minimum distance from the predator D_p (Weihs & Webb, 1984):

$$D_p^2 = |(X_0 - V_p t) + V_f t \cos \alpha|^2 + (V_f t \sin \alpha)^2 \quad (3.7)$$

X_0 is the initial distance between predator and prey. α is the escape angle of the prey. V_p and V_f are the velocities of the predator and the fish. Figure 11b shows how fish evade a predator. The solution is a velocity ratio:

$$\cos\alpha = \begin{cases} \dfrac{V_p}{V_f} & \text{if } \dfrac{V_p}{V_f} \leq 1 \\[2ex] \dfrac{V_f}{V_p} & \text{if } \dfrac{V_p}{V_f} \geq 1 \end{cases} \tag{3.8}$$

These threat responses cause the "fountain effect" and the "burst effect" in fish schools (Partridge, 1982) as each fish tries to increase its chances of survival. The fountain effect occurs when a predator moves toward a fish school and the school splits and flows around the predator. The school reforms behind the predator. In the burst effect the school expands in the form of a sphere to evade the predator.

$$\mathbf{C}_1 = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0] \tag{3.9}$$

$$\begin{aligned}
\mathbf{C}_1\mathbf{E}_A &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0] \rightarrow \\
\mathbf{C}_2 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0] \\
\mathbf{C}_2\mathbf{E}_A &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 0\ {-1}\ 0\ 0\ 0\ 0\ 0\ 0\ 0] \rightarrow \\
\mathbf{C}_3 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0] \\
\mathbf{C}_3\mathbf{E}_A &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 1] \rightarrow \\
\mathbf{C}_4 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 1] \\
\mathbf{C}_4\mathbf{E}_A &= [0\ 0\ 0\ 0\ 0\ 0\ {-1}\ {-1}\ 0\ 0\ 1\ 0\ 0\ 0\ {-1}\ 0\ 1\ 2\ 1\ 0\ 1\ {-2}\ {-1}\ 2] \rightarrow \\
\mathbf{C}_5 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 0\ 1] \\
\mathbf{C}_5\mathbf{E}_A &= [0\ 0\ 0\ 0\ 0\ 0\ {-1}\ {-1}\ 0\ 0\ 0\ 1\ 0\ 0\ {-2}\ {-1}\ 0\ 3\ 1\ 1\ {-2}\ {-1}\ {-2}\ 0] \rightarrow \\
\mathbf{C}_6 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 0] \\
\mathbf{C}_6\mathbf{E}_A &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ {-1}\ {-1}\ {-1}\ 2\ {-1}\ 1\ 0\ {-1}\ 0\ 0] \rightarrow \\
\mathbf{C}_7 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0] \\
\mathbf{C}_7\mathbf{E}_A &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ {-1}\ 0\ 0\ 0\ 0\ 0\ 2\ {-1}\ 0\ 0\ 0\ 0\ 0] \rightarrow \\
\mathbf{C}_8 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0] \\
\mathbf{C}_8\mathbf{E}_A &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ {-1}\ 1\ 0\ 0] \rightarrow \\
\mathbf{C}_9 &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0] \\
\mathbf{C}_9\mathbf{E}_A &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 0\ {-1}\ 0\ 0\ 0\ {-1}\ 1\ 0\ 0] \rightarrow \\
\mathbf{C}_{10} &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0] \\
\mathbf{C}_{10}\mathbf{E}_A &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ {-1}\ 0\ 0\ 1\ 0\ 1\ 1] \rightarrow \\
\mathbf{C}_{11} &= [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 1]
\end{aligned} \tag{3.10}$$

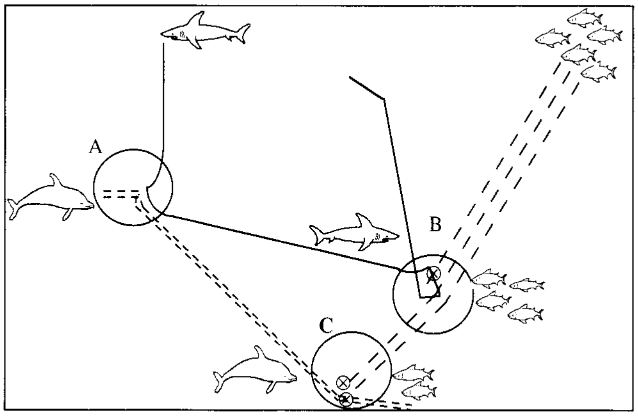


Figure 9. *FCMs control the virtual world. The augmented FCM controls the actions of the actors. In event A the hungry shark forces the dolphin herd to run away. Each dashed line stands for a dolphin swim path. In event B the shark finds the fish and eats some. Each dashed line stands for the path of a fish in the school. The cross shows the shark eating a fish. In event C the fish run into the dolphins and suffer more losses. The solid lines are the dolphin paths. The dashes are the fish swim paths. The cross shows a dolphin eating a fish.*

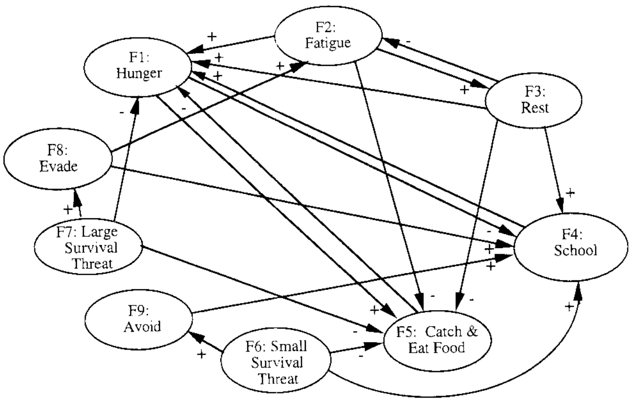


Figure 10. *Example of a nested FCM. The concept of a survival threat divides into two subconcepts that each map to a different survival tactic.*

4 Adaptive Fuzzy Cognitive Maps

An adaptive FCM changes its causal web in time. The causal web learns from data. The causal edges or rules change in sign and magnitude. The additive scheme (2.3) is a type of causal learning since it changes the FCM edge strengths. In general an edge e_{ij} changes with some first-order learning law:

$$\dot{e}_{ij} = f_{ij}(\mathbf{E}, \mathbf{C}) + g_{ij}(t) \qquad (4.1)$$

Here g_{ij} is a forcing function. Data fire the concept nodes and in time this leaves a causal pattern in the edge. Causal learning is local in f_{ij}. It depends on just its own value and on the node signals that it connects:

$$\dot{e}_{ij} = f_{ij}(e_{ij}, C_i, C_j, \dot{C}_i, \dot{C}_j) + g_{ij}(t) \qquad (4.2)$$

Correlation or Hebbian learning can encode some limit cycles in the FCMs or temporal associative memories (TAMs) (Kosko, 1988a). It adds pairwise correlation matrices in (4.1). This method can store only a few patterns. Differential Hebbian learning encodes changes in a concept in Eq. (4.2). Both types of learning are local and light in computation.

To encode binary limit cycles in connection matrix **E** the TAM method sums the weighted correlation matrices between successive states (Kosko, 1988a). To encode the limit cycle $\mathbf{C}_1 \rightarrow \mathbf{C}_2 \rightarrow \mathbf{C}_3 \rightarrow \mathbf{C}_1$ we first convert each binary state $\mathbf{C}_i$ into a bipolar state vector $\mathbf{X}_i$ by replacing each 0 with a -1. Then **E** is the weighted sum

$$\mathbf{E} = q_1\mathbf{X}_1^T\mathbf{X}_2 + q_2\mathbf{X}_2^T\mathbf{X}_3 + \cdots + q_{n-1}\mathbf{X}_{n-1}^T\mathbf{X}_n + q_n\mathbf{X}_n^T\mathbf{X}_1 \qquad (4.3)$$

The length of the limit cycle should be less than the number of concepts, or else crosstalk can occur. Proper weighting of each correlation matrix pair can improve the encoding (Wang, Cruz, & Mulligan, 1991) and thus increase the FCM storage capacity. Correlation learning is a form of the unsupervised signal Hebbian learning law in neural networks (Kosko, 1992):

$$\dot{e}_{ij} = -e_{ij} + C_i(x_i)C_j(x_j) \qquad (4.4)$$

A virtual world can encode an event sequence with (4.3) or (4.4). A simple chase cycle might be $\mathbf{C}_1 \rightarrow \mathbf{C}_2 \rightarrow \mathbf{C}_3$:

$$\mathbf{C}_1 = [1\ \ 0\ \ 1\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 1]$$

$$\mathbf{C}_2 = [1\ \ 0\ \ 1\ \ 1\ \ 1\ \ 0\ \ 0\ \ 0\ \ 1\ \ 0]$$

$$\mathbf{C}_3 = [1\ \ 0\ \ 0\ \ 0\ \ 1\ \ 0\ \ 0\ \ 0\ \ 1\ \ 1]$$

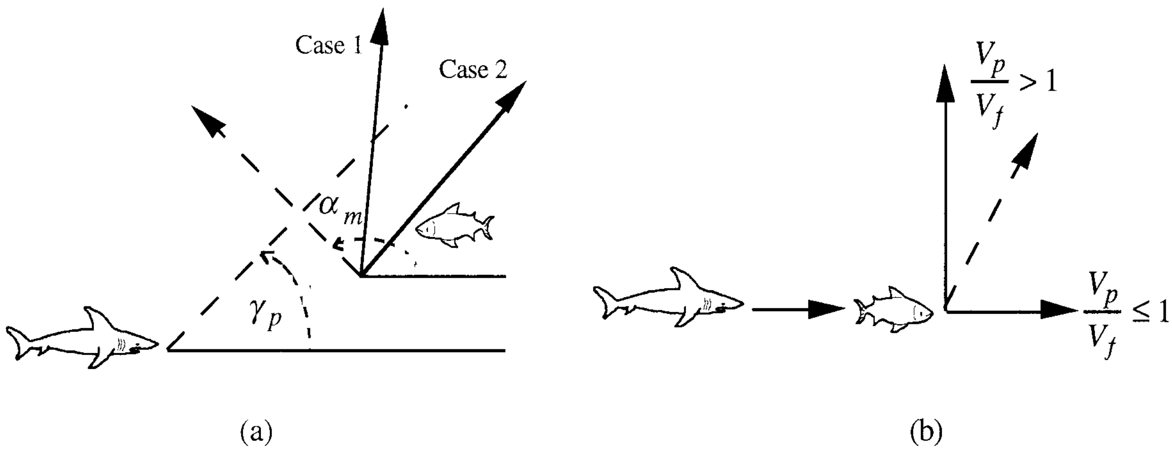


Figure 11. *Fish change their behavior as the degree of threat changes. (a) The fish minimize time within the sighting angle of the predator. Case 1 shows the angle of escape when the fish swim faster. Case 2 shows the desired angle when the predator swims faster. (b) The fish maximize the distance between themselves and the predator to evade the predator. The fish swim straight ahead when the fish swim faster than the predator. The fish swim away at an angle if the predator swims faster.*

Then (4.3) gives the FCM connection matrix **E** shown in equation (4.5).

$$\mathbf{E} = \begin{array}{c} \\ D_1 \\ D_2 \\ D_3 \\ D_4 \\ D_5 \\ D_6 \\ D_7 \\ D_8 \\ D_9 \\ D_{10} \end{array} \begin{array}{c} \begin{array}{cccccccccc} D_1 & D_2 & D_3 & D_4 & D_5 & D_6 & D_7 & D_8 & D_9 & D_{10} \end{array} \\ \left[\begin{array}{rrrrrrrrrr} 3 & -3 & 1 & -1 & 1 & -3 & -3 & -3 & 1 & 1 \\ -3 & 3 & -1 & 1 & -1 & 3 & 3 & 3 & -1 & -1 \\ 1 & -1 & -1 & -3 & -1 & -1 & -1 & -1 & -1 & -3 \\ -1 & 1 & 1 & -1 & -3 & 1 & 1 & 1 & -3 & 1 \\ 1 & -1 & 3 & 1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -3 & 3 & -1 & 1 & -1 & 3 & 3 & 3 & -1 & -1 \\ -3 & 3 & -1 & 1 & -1 & 3 & 3 & 3 & -1 & -1 \\ -3 & 3 & -1 & 1 & -1 & 3 & 3 & 3 & -1 & -1 \\ 1 & -1 & 3 & 1 & -1 & -1 & -1 & -1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 3 & -1 & -1 & -1 & 3 & -1 \end{array}\right] \end{array} \tag{4.5}$$

Then we get the limit cycle in equation (4.6).

$$\begin{aligned} \mathbf{C}_1\mathbf{E} &= [5 \quad -5 \quad 3 \quad 1 \quad 3 \quad -5 \quad -5 \quad -5 \quad 3 \quad -1] \rightarrow \mathbf{C}_2 = [1 \quad 0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0] \\ \mathbf{C}_2\mathbf{E} &= [5 \quad -5 \quad -5 \quad -7 \quad 3 \quad -5 \quad -5 \quad -5 \quad 3 \quad 7] \rightarrow \mathbf{C}_3 = [1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 1] \\ \mathbf{C}_3\mathbf{E} &= [6 \quad -6 \quad 2 \quad -6 \quad -2 \quad -6 \quad -6 \quad -6 \quad -2 \quad 6] \rightarrow \mathbf{C}_1 = [1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1] \end{aligned} \tag{4.6}$$

Correlation encoding treats negative and zero causal edges the same. It can encode "spurious" causal implications between concepts such as $e_{6,2} = 3$. This means searching for food causes a desire to socialize. Correlation encoding is a poor model of inferred causality. It says two concepts cause each other if they are on at the same time. Differential Hebbian learning encodes causal changes to avoid spurious causality. The concepts must move in the same or opposite directions to infer a causal link. They must come on and turn off at the same time or one must come on as the other turns off. Just being on does not lead to a new causal link. The patterns of turning on or off must correlate positively or negatively.

The differential Hebbian learning law (Kosko, 1988a) correlates concept changes or velocities:

$$\dot{e}_{ij} = -e_{ij} + \dot{C}_i(x_i)\dot{C}_j(x_j) \tag{4.7}$$

So $\dot{C}_i(x_i)\dot{C}_j(x_j) > 0$ iff concepts C_i and C_j move in the same direction. $\dot{C}_i(x_i)\dot{C}_j(x_j) < 0$ iff concepts C_i and C_j move in opposite directions. In this sense (4.7) learns patterns of causal change. The first-order structure of (4.7) implies that $e_{ij}(t)$ is an exponentially weighted average of paired (or lagged) changes. The most recent changes have the most weight. The *discrete* change $\Delta C_i(t) = C_i(t) - C_i(t-1)$ lies in $\{-1, 0, 1\}$. The discrete differential Hebbian learning can take the form

$$e_{ij}(t+1) = \begin{cases} e_{ij}(t) + c_t[\Delta C_i(x_i)\Delta C_j(x_j) - e_{ij}(t)] & \text{if } \Delta C_i(x_i) \neq 0 \\ e_{ij}(t) & \text{if } \Delta C_i(x_i) = 0 \end{cases} \tag{4.8}$$

Here c_t is a learning coefficient that decreases in time (Kosko, 1988b). $\Delta C_i\, \Delta C_j > 0$ iff concepts C_i and C_j move in the same direction. $\Delta C_i\, \Delta C_j < 0$ iff concepts C_i and C_j move in opposite directions. **E** changes only if a concept changes. The changed edge slowly "forgets" the old causal changes in favor of the new ones. This causal law can learn higher-order causal relations if it correlates multiple cause changes with effect changes.

We used differential Hebbian learning to encode a feeding sequence and a chase sequence in a FCM. The concepts in the ith row learn only when $\Delta C_i(x_i)$ equals 1 or -1. We used $c_t(t_k) = 0.1\,[1 - (t_k/1.1N)]$. The training data came from the dolphin FCM in Figure 6. This gave the $\mathbf{E}_D$ in equation (4.9).

$$\mathbf{E}_D = \begin{array}{c|cccccccccc|} & D_1 & D_2 & D_3 & D_4 & D_5 & D_6 & D_7 & D_8 & D_9 & D_{10} \\ \hline D_1 & 0.0 & 0.0 & -0.1 & -0.6 & -0.5 & 0.6 & -0.6 & 0.0 & 0.0 & 0.0 \\ D_2 & 0.0 & -0.5 & 0.5 & -0.5 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ D_3 & 0.0 & 0.0 & 0.0 & 0.9 & 0.5 & 0.0 & 0.0 & 0.0 & -0.1 & -0.1 \\ D_4 & 0.8 & 0.0 & -0.4 & 0.0 & 0.0 & -0.2 & 0.0 & 0.0 & -0.2 & -0.1 \\ D_5 & 0.4 & -0.3 & 0.3 & -0.1 & -0.2 & -0.2 & 0.0 & 0.0 & -0.1 & 0.1 \\ D_6 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & -0.5 & 1.0 & -0.6 & 0.0 & 0.0 \\ D_7 & 0.0 & -0.6 & 0.0 & 0.0 & -0.5 & 0.0 & -0.5 & 1.0 & 0.0 & 0.0 \\ D_8 & 0.0 & 1.0 & -0.5 & 0.0 & 0.5 & 0.0 & 0.0 & -0.5 & 0.0 & 0.0 \\ D_9 & -0.5 & 0.0 & 0.5 & 0.0 & 0.5 & 0.0 & 0.0 & 0.0 & 0.0 & 0.5 \\ D_{10} & -0.2 & 0.0 & 0.7 & 0.3 & -0.2 & 0.0 & 0.0 & 0.0 & 0.3 & -0.3 \\ \hline \end{array} \tag{4.9}$$

This learned edge matrix $\mathbf{E}_D$ resembles the FCM matrix in Figure 6. The causal links it lacks between $\mathbf{D}_{10}$ and $\{\mathbf{D}_6, \mathbf{D}_7, \mathbf{D}_8\}$ were not in the training set. The diagonal links terms for self-inhibition of each concept. This occurs since each concept is on for one cycle before the matrix transitions to the next state. The hunger input

$\mathbf{CL}_0 = [1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$ with a threshold of 0.5 now leads to the limit cycle shown in equation (4.10). This resembles the sequence of rest, eat, play, and rest from Section 3.1. Figure 12a shows the hand-designed

$$
\begin{aligned}
\mathbf{CL}_0\,\mathbf{E}_D &= [0.0\ \ 0.0\ {-0.1}\ {-0.6}\ {-0.5}\ \ 0.6\ {-0.5}\ \ 0.0\ \ 0.0\ \ 0.0] \rightarrow \mathbf{CL}_1 = [0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0]\\
\mathbf{CL}_1\,\mathbf{E}_D &= [0.0\ \ 0.0\ \ 0.0\ \ 0.0\ \ 0.0\ {-0.5}\ \ 1.0\ {-0.6}\ \ 0.0\ \ 0.0] \rightarrow \mathbf{CL}_2 = [0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0]\\
\mathbf{CL}_2\,\mathbf{E}_D &= [0.0\ {-0.6}\ \ 0.0\ \ 0.0\ {-0.5}\ \ 0.0\ {-0.5}\ \ 1.0\ \ 0.0\ \ 0.0] \rightarrow \mathbf{CL}_3 = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0]\\
\mathbf{CL}_3\,\mathbf{E}_D &= [0.0\ \ 1.0\ {-0.5}\ \ 0.0\ \ 0.5\ \ 0.0\ \ 0.0\ {-0.5}\ \ 0.0\ \ 0.0] \rightarrow \mathbf{CL}_4 = [0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0]\\
\mathbf{CL}_4\,\mathbf{E}_D &= [0.4\ {-0.8}\ \ 0.8\ {-0.6}\ {-0.2}\ {-0.2}\ \ 0.0\ \ 0.0\ {-0.1}\ \ 0.1] \rightarrow \mathbf{CL}_5 = [0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0]\\
\mathbf{CL}_5\,\mathbf{E}_D &= [0.0\ \ 0.0\ \ 0.0\ \ 0.9\ \ 0.5\ \ 0.0\ \ 0.0\ \ 0.0\ {-0.1}\ {-0.1}] \rightarrow \mathbf{CL}_6 = [0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0]\\
\mathbf{CL}_6\,\mathbf{E}_D &= [1.2\ {-0.3}\ {-0.1}\ {-0.1}\ {-0.2}\ {-0.4}\ \ 0.0\ \ 0.0\ {-0.3}\ \ 0.0] \rightarrow \mathbf{CL}_1 = [1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]
\end{aligned}
\tag{4.10}
$$

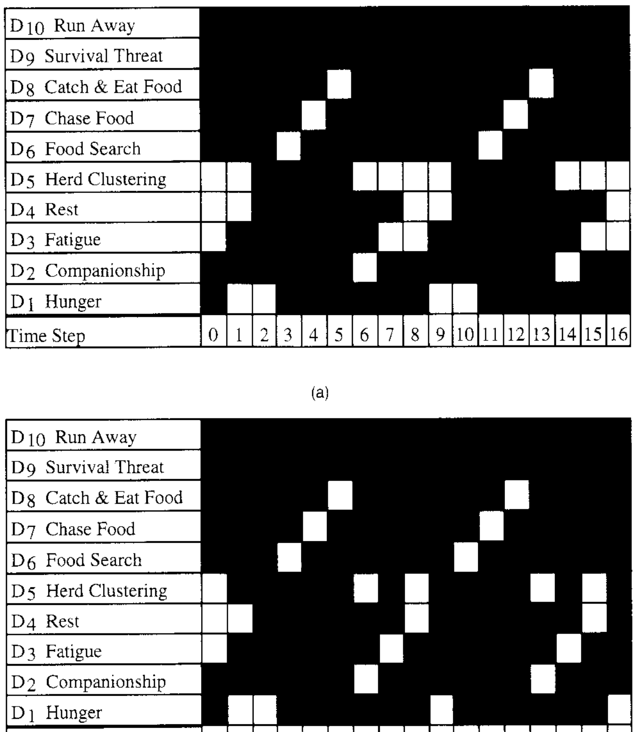


Figure 12. *Limit cycle comparison between the hand-designed system and the FCM found with differential Hebbian learning. Each column is a binary state vector. (a) Rest, feed, play, rest limit cycle for the FCM in Figure 6. (b) Limit cycle for the FCM found with (4.8).*

limit cycle from Section 3.1. Figure 12b shows the limit cycle from FCM found with differential Hebbian learning. The DHL limit cycle is one step shorter. Both FCMs have just one limit cycle and the null fixed point in the space of 2^{10} binary state vectors. The value of $\mathbf{E}_{D5}$ does not change over 2 intervals. The learning law in (4.8) learns only if there is a change in the node.

5 Conclusions

Fuzzy cognitive maps can model the causal web of a virtual world. The FCM can control its local and global nonlinear behavior. The local fuzzy rules or edges and the fuzzy concepts they connect model the causal links within and between events. The global FCM nonlinear dynamics give the virtual world an "arrow of time." A user can change these dynamics at will and thus change the causal processes in the virtual world. FCMs let experts and users choose a causal web by drawing causal pictures instead of by stating equations.

FCMs can also help visualize data. They show how variables relate to one another in the causal web. The FCM output states can guide a cartoon of the virtual world. The cartoon animates the FCM dynamics as the system trajectory moves through the FCM state space. This can apply to models in economics, medicine, history, and politics (Taber, 1991) where the social and causal web can change in complex ways that may arise from changing the sign or magnitude of a single FCM causal rule or edge.

The additive structure of combined FCMs permits a Delphi (Martino, 1972) or questionnaire approach to knowledge acquisition. These new causal webs can change an adaptive FCM that learns its causal web as neural-like learning laws process time-series data. Experts can add their FCM matrices to the adaptive FCM to initialize or guide the learning. Such a causal web can learn the user's values and action habits and perhaps can test them or train them.

More complex FCMs have more complex dynamics and can model more complex virtual worlds. Each concept node can fire on its own time scale and fire in its own nonlinear way. The causal edge flows or rules can have their own time scales too and may increase or decrease the causal flow through them in nonlinear ways. This behavior does not fit in a simple FCM with threshold concepts and constant edge weights.

A FCM can model these complex virtual worlds if it uses more nonlinear math to change its nodes and edges. The price paid may be a chaotic virtual world with unknown equilibrium behavior. Some users may want this to add novelty to their virtual world or to make it more exciting. A user might choose a virtual world that is mildly nonlinear and has periodic equilibria. At the other extreme the user might choose a virtual world that is so wildly nonlinear it has only aperiodic equilibria. Think of a virtual game of tennis or raquetball where the gravitational potential changes at will or at random.

Fuzziness and nonlinearity are design parameters for a virtual world. They may give a better model of a real process. Or they may be just more fun to play with.

References

Ackerman, E., Gatewood, L., Rosevear, J., & Molnar, G. (1969). Blood glucose regulation and diabetes. In F. Heinmets (Ed.), *Concepts and models of biomathematics.* New York: Marcel Dekker, 131-156.

Badler, N. I., Webber, B. L., Kalita, J., & Esakov, J. (1991). Animation from instructions. In N. I. Badler, B. A. Barsky, & D. Zeltzer (Eds.), *Making them move: Mechanics, control, and animation of articulated figures.* San Mateo, CA: Morgan Kaufmann, 51–98.

Brooks, R. A. (1989). A robot that walks: Emergent behaviors from a carefully evolved network. *Neural Computation, 1*(2), 253–262.

Brown, R. A. (1991). *Fluid mechanics of the atmosphere.* San Diego, CA: Academic Press.

Connell, J. H. (1990). *Minimalist mobile robotics: A colony-style architecture for an artificial creature.* San Diego, CA: Academic Press.

Craig, J. J. (1986). *Introduction to robotics.* Reading, MA: Addison-Wesley.

Dickerson, J. A., & Kosko, B. (1993). Fuzzy function approximation with supervised ellipsoidal learning. *World Conference on Neural Networks (WCNN'93), II,* 9–17.

Gibson, W. (1984). *Neuromancer.* New York: Ace Books.

Grossberg, S. (1982). *Studies of mind and brain.* Boston: Reidel.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayered feedforward networks are universal approximators. *Neural Networks, 2*(5), 359–366.

Kaelbling, L. P. (1987). An architecture for intelligent reactive systems. In M. P. Georgeff & A. L. Lansky (Eds.), *Reasoning about actions and plans.* San Mateo, CA: Morgan Kaufmann, 395–410.

Koopman, B. O. (1980). *Search and screening.* New York: Pergamon Press.

Kosko, B. (1986). Fuzzy cognitive maps. *International Journal Man-Machine Studies, 24,* 65–75.

Kosko, B. (1988a). Hidden patterns in combined and adaptive knowledge networks. *International Journal of Approximate Reasoning, 2,* 337–393.

Kosko, B. (1988b). Bidirectional associative memories. *IEEE Transactions Systems, Man, and Cybernetics, 18,* 49–60.

Kosko, B. (1992). *Neural networks and fuzzy systems: A dynamical systems approach to machine intelligence.* Engelwood Cliffs, NJ: Prentice-Hall.

Kosko, B. (1993). Fuzzy systems as universal approximators. *IEEE Transaction on Computers;* an early version appears in *Proceedings of the 1st IEEE International Conference on Fuzzy Systems (FUZZ-IEEE FUZZ92),* 1153–1162.

Krueger, M. (1991). *Artificial reality II* (2nd ed.). Reading, MA: Addison-Wesley.

Maes, P. (1990). Situated agents can have goals. *Robotics and Autonomous Systems, 6,* 49–70.

Martino, J. P. (1972). *Technological forecasting for decisionmaking.* New York: American Elsevier.

Minsky, M. L. (1985). *The society of mind.* New York: Touchstone Books, Simon & Schuster.

Partridge, B. L. (1982). The structure and function of fish schools. *Scientific American, 246*(6), 114–123.

Reynolds, C. W. (1987). Flocks, herds, and schools: A distributed behavioral model. *Computer Graphics, 21*(4), 25–34.

Shane, S. H. (1990). Comparison of bottlenose dolphin behavior in Texas and Florida, with a critique of methods for studying dolphin behavior. In S. Leatherwood & R. R. Reeves (Eds.), *The bottlenose dolphin,* San Diego, CA. Academic Press, 541–558.

Taber, W. R. (1991). Knowledge processing with fuzzy cognitive maps. *Expert Systems with Applications, 2*(1), 83–87.

Taber, W. R., & Siegel, M. (1987). Estimation of expert weights with fuzzy cognitive maps. *Proceedings of the 1st IEEE International Conference on Neural Networks (ICNN-87), II,* 319–325.

Wang, Y. F., Cruz, J. B., & Mulligan, J. H. (1991). Guaranteed recall of all training pairs for bidirectional associative memory. *IEEE Transactions on Neural Networks, 2*(6), 559–567.

Weihs, D., & Webb, P. W. (1984). Optimal avoidance and evasion tactics in predator-prey interactions. *Journal of Theoretical Biology, 106,* 189–206.

Winston, P. H. (1984). *Artificial intelligence* (2nd ed.). Reading, MA: Addison-Wesley.