SMART AMORPHOUS COMPUTATION

E.V.Krishnamurthy
Computer Sciences Laboratory, Australian National University,
Canberra, ACT 0200, Australia.
abk@mail.rsise.anu.edu.au

SUMMARY

An interacting multi - agent system in a network can act like a nature - inspired Amorphous Smart System (SS) exhibiting the four salient properties :

(i) Collective, coordinated and efficient

(ii) Self -organization and emergence

(iii) Power law scaling or scale invariance under emergence

(iv) Adaptive, fault tolerant and resilient against damage.

We explain how these basic properties can arise among agents through random enabling, inhibiting, preferential attachment and growth of a multiagent system.

The quantitative understanding of a Smart system with an arbitrary interactive topology is extremely difficult. Hence we cannot design a general purpose programmable Smart system.

However, for specific applications and a pre-defined static interactive topology among the agents, the quantitative parameters can be obtained through simulation to build a specific SS.

AMORPHOUS means shapeless or noncrystalline. This term is used as an adjective for a large collection of computing agents (not necessarily homogeneous),having no specific geometric or topological connective structure.
The amorphous computers, "evolve" these structural connections over time, by communicating with each other in a local neighbourhood depending upon their intera ction with their environment.

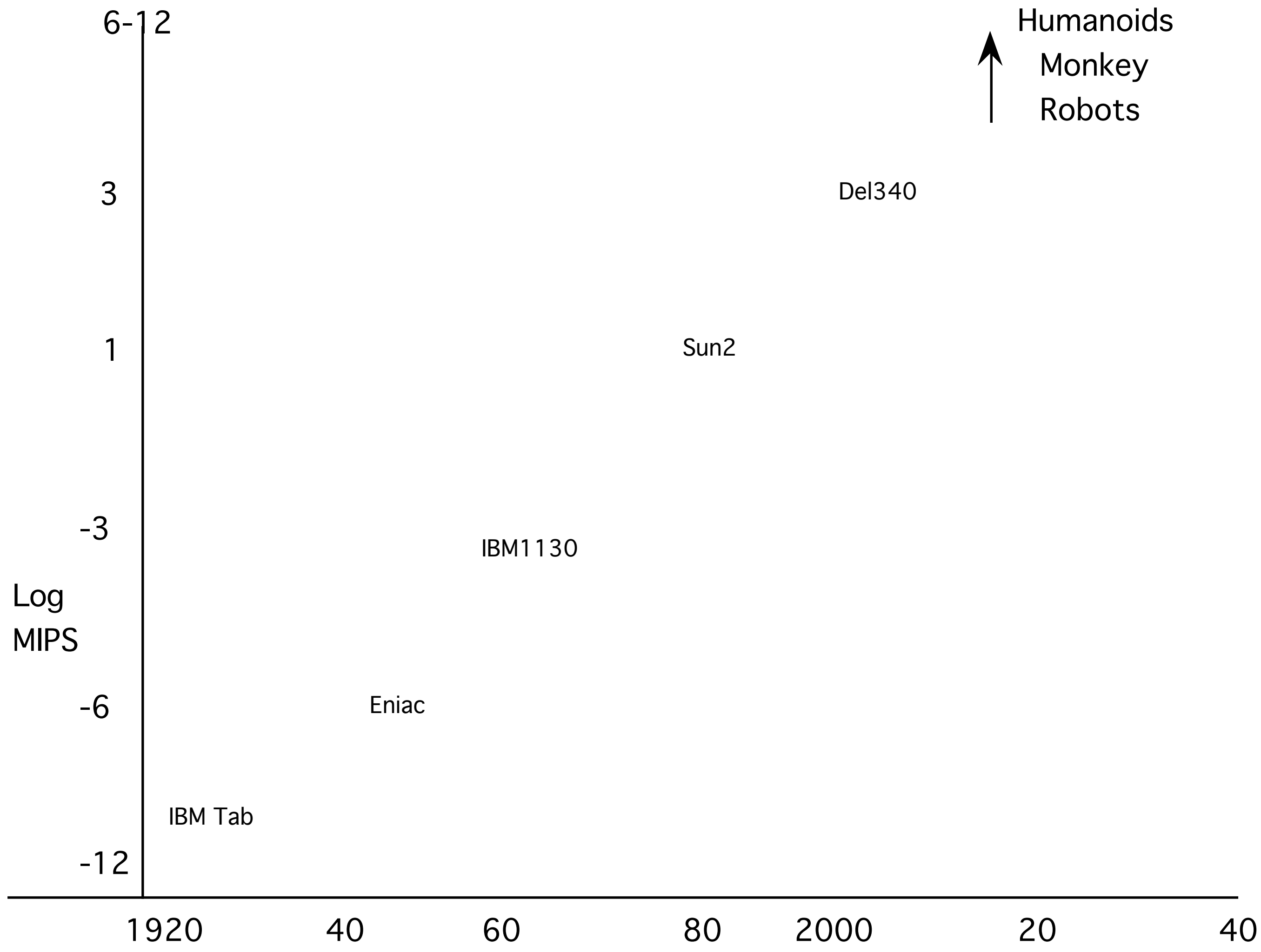
Amorphous computation is also called "Softwired computation"since these connections can change over time.

This lecture deals with some problems addressed in Roger Penrose's Book "Shadows of mind"!

These computations are similar to those computations in brain based on diffusive chemicals.They have slow acting electrical components (50 millisec and communication rate is 100Hz)- but make numerous variable Connections per second (TEN QUADRILLION) 10^{16}. They are not based on using universal circuits operating at high speeds and operating at 10^{9}Hz and do not use modular software . Yet the they are superior ,e.g., humans can recognize an object in 400-500 millisecs.

Efficient operation seems to come from customizing the hardware for the task.That is adaptation through softwiring for a specific task. This is done by the tip of "Axon " searching for chemical gradients arising from neural signals. This is very similar to swarming of ants. Thus the communication pathways are fractals arising from chemical diffusion. To match this we need to increase the transistor count to 10000 times more- that is a billion transistors in one sq.cm!Hence the search for alternative schemes to reach biological processing capability through bio-inspired evolution and development.

COMPUTATIONAL POWER



WHAT IS A SMART SYSTEM!
A smart system EXHIBITS the following important properties:

1.Interactive, Collective, Coordinated and Highly efficient Operation:

- **Consists of a large number of independent elements that communicate with their environment and with each other, through the interaction with the environment they share, and can perform actions collectively and in parallel, in a coordinated manner.**
- **The rules that govern their interaction and actions are essentially local to each element.**

2. Self organization and emergence:

• Some order arises spontaneously in the system as a whole (called "Emergence"). That is the behaviour of the system cannot be learned by reduction methods by treating the system as an asemblage of whole from parts (HOLISTIC).

3. Power-Law Scaling or Scale-Invariance Property

Power -law scaling relation arises for the emergent properties of the SS. This property is essential for fault resilience and self-organization.

4. Adaptive, fault-tolerant and flexible Operation:

• That is the whole system adjusts to new situations, so as bring about some advantageous realignment of the system.

It appears that these four properties are interrelated to self-organization, stigmergy and self-assembly, as well as, to systems exhibiting positive metric entropy, self-similarity and the small world phenomena. It is hard to separate these concerns as they form fuzzy boundaries .

1. HOW TO MODEL THEM?

We discuss three kinds of models that have a direct bearing on these issues :(i) Fractal and percolation model, (ii) Chaotic and nonlinear dynamical model and (iii) Topological (network) or graph model.

2.Is the emergence of SS analogous to the critical phenomenon in physics or percolation? What are the suitable parameters to describe this phenomenon?

We discuss this aspect from the point of view of geometric parameters such as: Lyapunov exponents , strange attractors , metric entropy, as well as, topological indices such as, Cluster coefficient, Average degree distribution and the correlation length of the interacting network.

3.Can we design a general purpose programmable smart system?

The answer is no, if we want to have an SS with exactly specified propeties. However, special purpose Smart systems with approximate statistical properties can be designed by using simulation procedures and parameter tuning.

RECALL: Smart systems possess the four properties:
1.Interactive, Collective and Parallel operation
2.Self organization through emergence, 3.Power-law scaling
4.Adaptive and flexible operation

Properties 1 and 2 hold near the critical point in a phase transition in a physical system or in a percolation modelnear the percolation threshold. These two models lie at the border of order and disorder and are concerned with the collective and parallel interaction among the microscopic objects that result in scale invariance (powerlaw behaviour with a critical exponent) and universality. Further, the percolation / Phase transition systems have the properties 3 and 4, namely, scaling, adaptive and flexible behaviour to tolerate failures, since they both deal with the formation of clusters or creating paths among distant neighbours, even if some of these neighbours are non-cooperative.

This is also called ”SOFT-WIRING”, since hardware is connected through modifiable wiring.

The percolation process provides for soft-wiring the hardware by providingcomunication pathways that can be altered- exmples: human visual system, lungs, swarms.

METRIC/ ALGORITHMIC ENTROPY/TURING COMPUTABILITY

- **Measure the disorder in a system.**
- **Metric entropy is defined as the average information per measurement obtained from an (denumerable) infinite sequence of finite precision - identical measurements made on a time evolving system from a time minus infinity to plus infinity.**

ZERO METRIC ENTROPY

implies that the measurement sequence, begun in the remote past, ceases to provide any more additional information after a finite time T. This means adequate information has been acquired to predict the properties of that system completely and unambiguously.

POSITIVE METRIC ENTROPY

- **Means the entropy in the system has not decreased as much as the information obtained by measurement and still the uncertainty remains.**
- **The system may be producing entropy faster than what we can measure.**

Here, we measure the bit-length of the minimal program (or functional rule) that generates a required output sequence from a given input sequence. e.g a totally erratic infinite sequence cannot be described as an output sequence of some input sequence by using a simple program whose length is smaller than the desired output length.

For positive K-entropy, in the long run the recording of information for evolution increases unbounded and the evolution cannot be followed deterministically, unless a disproportionately long or infinite time is devoted to this task beyond a critical time t = T(c).

As we approach T(c), the recording is critically slowed down -much like in phase transition. At this time the motion is no longer deterministic and the forward and backward evolution are not reversible resulting in a spontaneous breakdown of time - reversal symmetry. For K = 0 the evolution can be recorded deterministically.

Thus a phase -transition can arise between the Turing expressible and Turing non-expressible systems . By the universality principle we expect that critical exponents and scaling factors exist for such systems so that we can say when the system can become smart.

HIERARCHY AMONG MACHINES

Based on Metric-Algorithmic entropy we can establish a hierarchy among the computational systems . Here we classify the two major classes of machines, ordinary (O) and dissipative (P) based on metric entropy as below.

O. Ordinary or Zero Metric Entropy Machines:

Completely structured, Deterministic, Exact behaviour (or Algorithmic) Machines.This class contains: Finite State machines, Push down-stack machines, Turing Machines (Deterministic) that halt and Exactly integrable Hamilton flow machines. Such machines are information-lossless; their outputs contain all the required information as dictated by the programs and the information gained by running the program exactly equals the entropy decrease in the program.

P. Positive Metric Entropy Machines:

These are Partially Structured , Non-deterministic , Probabilisitic, , Kolmogorov -flow (K-flow) or chaotic systems with the motion in phase space unpredictable, nonequilibrium systems exhibiting macro and emergent behaviour - such as Chemical and Biological machines and living systems .

Without positive entropy evolution stops!
Distinguishability and diversity vanish!
MUTATION INCREASES ENTROPY BY PROVIDING DISSIMILARITY
SURVIVALREDUCES ENTROPY -HOMOGENIZING

Quantum systems?

INTEGRABILITY, COMPUTABILITY,CHAOS AND SMARTNESS

Integrable systems are associated with regular periodic motion. Nonintegrable systems show various degrees of irregular dynamics:

1.Ergodic:Here the set of points in phase space behave in such a way that the time-average along a trajectory equals the ensemble average over the phase space. Although, the trajectories wander around the whole phase space ,two neighbouring initial points can remain fully correlated over the entire time of evolution.

2. Mixing: The initial points in the phase space can spread out to cover it in time but at a rate weaker than the exponential (e.g.inverse power of time).

3. K-flow or Chaos: The trajectories cover the phase space in such a way that the initially neighbouring points separate exponentially and the correlation between two initially neighbouring points decays with time exponentially. It is with this class of irregular motion one can define classical chaos.

Each of the properties below imply all the properties above, e.g., within a chaotic region the trajectories are ergodic .

CHAOTIC SYSTEM:

Classical motion is chaotic if the flow of the trajectories in a given region of phase space has a positive definite K-entropy ,that is the average of sum of all positive Lyapunov characteristic numbers that measure the rate of exponential separation between two neighbouring points in the phase space and the average is taken over some particular neighbourhood. Chaos indicates hypersensitivity on the initial conditions; the system becomes inseparable (metric transitivity) .

and the periodic points are dense.Whole is simply not a sum of parts! (EMERGENCE!) .

Thus Smart systems are necessarily chaotic and governed by STRANGE ATTRACTORS having fractal dimension!

COMPUTABILITY THEORY AND INTEGRABILITY

- The recursive function over integers consists of some basic functions and closed under certain operations:

{ZERO, SUCCESSOR, PREDECESSOR, PROJECTION, COMPOSITION, PRIMITIVE RECURSION, BOUNDED MINIMALIZATION}.

- When generalised to the primitive real class this takes the form: {ZERO,SUCCESSOR, PREDECESSOR, PROJECTION, COMPOSITION, INTEGRATION, ZEROFINDING on REAL}.

Since Zero finding is equivalent to finding an attractor, Integrability and attractor finding in a dynamical system are equivalent to primitive recursivity and testing for zero in computability theory. Note that integration is not closed under minimalization like primitive recursion.Thus the routes to chaos and noncomputability are parallels in their respective domains . Such routes are important for understanding chaos and controlling them - at least individually!

SIMULATING A MIXTURE OF ENTROPY IN SYSTEMS

Thus, to enlarge the capabilities of class O machines by simulating special features of class P machines - using nondeterminism, randomness, approximation, probabilities, equilibrium-statistical mechanical (e.g. simulated annealing) and non-equilibrium statistical mechanical (e.g. genetic algorithms and Ant algorithm) .

SIMULATING SELF ORGANIZATION

Thus to simulate a Self-organizing system we need a Programming paradigm that includes on its rule-base deterministic, nondeterministic and probabilistic choices.

In distributed computing with agents, and amorphous computing, a very large disordered interconnection network having a positive metric entropy arises in the state space. Such networks have various statistical and nonlinear dynamical properties- random, small-world and scale free networks- having different fractal dimensions and emergent properties. To realize self-organization, therefore, the agents need to strategically control their links with other agents based upon some policy. Such policies are Turing non-expressible and are entirely empirical and statistical.

AGENT-BASED MODELING SMART SYSTEM

An agent is a system that is capable of perceiving events in its environment or representing information about the current state of affairs and of acting in its environment guided by perceptions and stored information . Here, we consider the evolution of a massively many (a multiset of) agents into an SS in a network.

That is we are interested in connected pathways among agents satisfying certain special properties, such as-minimal cost route, clustering of agents, resilience of conectivity among the agents under failure, attractors, phase transitions and so on. These properties depend upon the temporal dynamics of the system as embodied in the Lyapunov exponents and the spatial structure of the dynamical system's attractors.

Agent-based Modeling, Simulation and Animation

Three crucial properties of Agents make them suitable for the simulation of Nature Inspired Smart System (NISS):
(i) *Autonomy*: Agents can make decisions on actions they want to do without explicit control from the user,
(ii) *Reactive*: Agents can respond appropriately depending upon the context, and
(iii) *Proactive*: Agents can act in anticipation of future goals that can meet the specified objectives.

In reactive animation, the system has to react to various kinds of events, signals and conditions that are often distributed and concurrent. Also they can be time critical exhibiting both digital and analog (or hybrid) behaviour. In addition the reactive system, as in cell biological system can contain components that signal each other and also repeatedly created and destroyed.

Animation is useful for teaching purposes -for example, in cell biology in which the molecules are involved in various interactions with other

cells through cell-signalling cascades. Cell signalling cascades consist of several steps of activation and inhibition.

The animation emphasises the order of events; it can be achieved through the introduction of “Avatara” which is realised as an extension of the Agent model by including a body with a desired geometry and related functions.

This requires:
a. The object space- a multi set of agents
b. An environment- the interaction geometry
c. Interaction matrix
Of course, when we define objects and their intraction the scale becomes important-molecular (10^{-12}), nano (10^{-9}), micro (10^{-6}), macro (10^{-3}).
HENCE AGENT-BASED MODELING NEEDS TO BE CAREFULLY DESIGNED.
In the molecular scale atoms and molecules are autonomous objects and their interactions are as inchemical bonds. Also Brownian motion plays a role.
In the nano scale, interfaces are molecular lattice.
In the micro scale we have a solid state object nd a homogeneous media
In the macro scale the organization is less autonmous and can be controlled by other external factors (man,animals).
Thus any simulation process (like a model) is never a true representative of the real- but should reflect it closely!

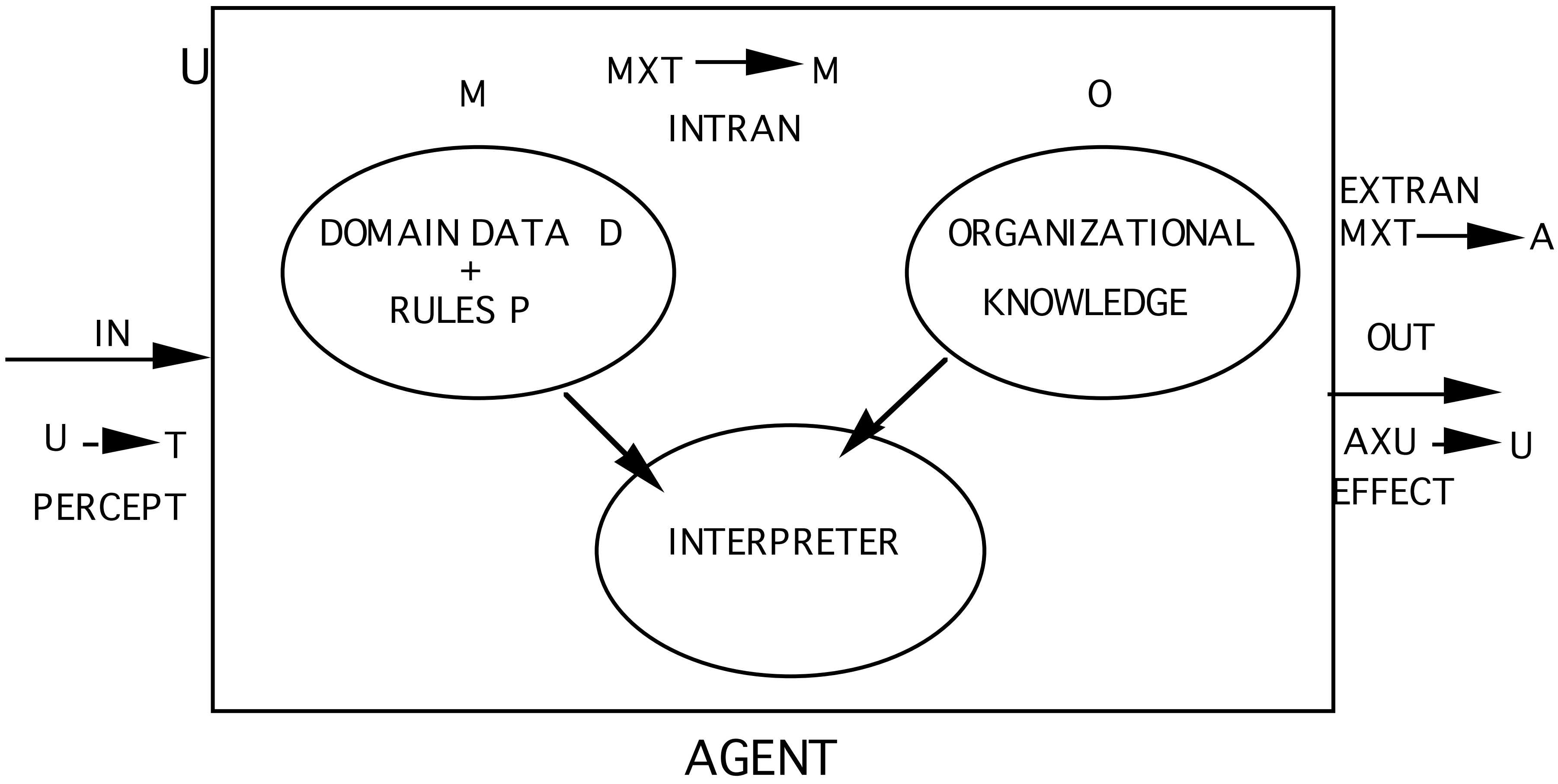

U
MXT M
M
O
INTRAN
DOMAIN DATA D
+
RULES P
ORGANIZATIONAL
KNOWLEDGE
EXTRAN
MXT A
IN
U T
PERCEPT
OUT
AXU U
EFFECT
INTERPRETER
AGENT

Figure 1

A multi-agent system is realised as a loosely coupled network of single agents; they interact among themselves and through the environment to solve a problem,. In order to use the multi-agent paradigm to realise cooperative and competitive computational tasks, we need to consider how the agents can interfere with each other. Such interferences can take place in four ways.

1. Enabling dependence (ED): Agent A(i) and agent A(j) are called enable dependent (or dataflow dependent) if the messages from A (i) creates the required precondition in A(j) to carry out a specific action .

2.Inhibit dependence (ID): Agents A (i) and A (j) are called inhibit dependent, if the actions of A (i) creates the required precondition in A(j) to prevent it from executing a specific action.

3. INTRAN Conflict (IC) : Agents A (i) and A (j) are opposition dependent (also called data-output dependent) through A(k)), if the order in which A (i) and A (j) enable A(k) and update A(k) produce different results in A(k); that is the objects A(i) and A (j) perform operations on A(k) that are not order reversible. That is, local serializability is not ensured in the INTRAN within A(k), if the actions are carried out within an agent in different order.

4. EXTRAN Conflict (EC): Agents A (i) and A(j) are data antidependent through A(k) if the order in which A(i) enables (inhibits) A(k), and A(j) enables (inhibits) A(k) result in different external actions (EXTRAN) by A(k) on the environment. That is the order in which information arrives from the environment and other agents affects the global serializability of the actions of an agent.

Remark: ED and ID:

The two properties ED and ID are crucial for modelling any life-like system which requires both positive and negative regulation and molecular switches. These rules permit an agent to enable itself (autocrine signalling for development or for autocatalysis), and also an agent A(i) to enable A(j) and A (j) to enable A(i) cyclically.

For example, A(i) can create the required precondition in A(k), so that A(j) can enable A(k). Also, A(i) can inhibit the required precondition in A(k) so that A(j) is prevented from enabling A(k).

Currently Petri net models are used in Systems Biology. The agent models are more general and provide greater flexibility and also help in animation.

Concurrency and Conflicts

In distributed computing and transaction processing: we require that the following two conditions are satisfied for global serialization when concurrent operations take place.

1. At each agent the actions in local actions are performed in the non-conflicting order (Local serializability).

2. At each agent the serialization order of the tasks dictated by every other agent is not violated. That is, for each pair of conflicting actions among transactions p and q, an action of p precedes an action of q in any local schedule, if and only if, the preconditions required for p do not conflict with those preconditions required for execution of the action q in the required ordering of all tasks in all agents (Global serializability).

The above two conditions require that the preconditions for actions in different agents A(i) and A(j) do not interfere or cause conflicts. These conditions are necessary for the stabilization of the multi-agent systems that the computations are locally and globally consistent.

Termination: For the termination of agent –based program, the interaction among the agents must come to a halt. When the entire set of agents halt we have an equilibrium state (or a fixed point) also called stability while dealing with exact computation in a deterministic system.

Non–termination, instability, multiple equilibria and chaos: These cases arise when the agents continue to interact indefinitely as in chemical oscillations, biological reactions, game theoretic problems, and cellular signal processing. Then the multiagent-system is sensitive to initial conditions leading to chaos having several attractors or equilibrium states. and self-organization.

Conflicts: Resolution or compromise? The conflicts arising in INTRAN and EXTRAN may require resolution or to an agreeable compromise. For example, the actions, "fold your right arm" or "stretch your right arm" are conflicting concurrent actions. We can achieve a compromise by keeping the arm folded half –way. Alternatively, one can blend the behaviour of actions, e.g., as in the actions "walk" and "run", if the

quantitative parameters can be suitably averaged over. These rules should be based on the context.

In order that a multiset of agents behaves like an SS, it should exhibit the four properties mentioned earlier. This will require that the agents are not centrally controlled, but can either enable (link) or inhibit (de-link) with other agents on their own,based on the information available from their environment.This will correspond to adaptation. This requires that the agents have adequate memory, intelligence,and pre-knowledge of the network of other nodes in the network.

We can use two different approaches inspired by Nature to enable (connect) or inhibit (disconnect) agents to form an interactive network:

1.*Use of markings similar to a chemical gradient or diffusion mechanism or a communication field* (Agents having simple intelligence, e.g ants, physical particles):

This provides a common spatial resource, where each agent can leave a mark that can be perceived by other agents. In essence, the agents use the space to construct a geometrical or a topological pathway that is preferentially traversed by many agents, resulting in a power-law scaling or small world beyond a critical threshold.

2.*Use of a positive feed-back or nonlinear response to the information available from knowledge sources, other agents may possess by connecting or disconnecting with other agents at random*:

This would require that *each agent knows what other agent knows, and how much they know measured in a taxonomic scale* (Agents with more complex intelligence) so that each agent can have a score about its neighbours to link, delink and form clusters.

Here, individual agents are points in space, and change over time is represented as movement of points, representing particles with different properties and the system dynamics is formulated using the rules:

(1) Stepping (or local coupling) rule:

The state of each individual agent is updated or revised in many dimensions, in parallel, so that the new state reflects each agent's previous best success.

(2) Landscaping (or global coupling) rule:

Each agent assumes a new best value of its state that depends on its past best value and a suitable function of the best values of its interacting neighbours,with a suitably defined neighbourhood topology and geometry.

All agents in the universe or selected chunks are updated using rules (1) and (2).

This would result in nonlinear iterative schemes among the agents, and can model Markovian random walks independent of the past history of the walk and non-Markovian random walks, dependent upon past history- e.g., self-avoiding, self-repelling and active random-walker . This can result in attractors having fractal dimensions presenting a swarm-like, flock-like appearances depending upon the Jacobian of the mapping

Role of Fractal , Percolation and Graph Models in NISS

An NISS has an associated data domain or space. In nature, this space is usually the three dimensional space; it is called the geometric dimension of the NISS. When the system is placed in an environment, it communicates through its surface area. Since the amount of communication is proportional to the surface area, a simple way to control the communication rate is to choose a suitable geometrical or topological structure that can spontaneously and easily be modified to vary the surface area. In biology, chemistry, and materials science, where surface phenomena play a crucial role, nature has preferred the evolution of fractal objects. Such a choice can optimize the interaction with the environment and provides for adaptation and the survival. In heterogeneous chemistry, the structure and geometry of the environment at which the reaction takes place plays an important role. It alone can dictate whether a reaction will take place at all. The geometric parameter is the fractal dimension. In fact environmental

interaction can change geometrical and topological features and conversely, those features can modify the interaction.

To model NISS using computing agents, therefore, the agents should be able to alter the pattern of their communication pathways, namely, the topology and geometry at will, resulting in the change of a fractal dimension.

Examples of such systems abound in nature: lungs, flock of birds, ant-swarms, formation of bacterial colonies,cellular signal propagation, Animal- human trails.

Also, the biological cells rearrange their internal components as they grow, divide and adapt to changing circumstances. For this purpose, they depend on a system of filaments called, cytoskeleton. The cytoskeleton is an interconnected network provides the machinery for the muscle cell contraction, and in the neuron to extend an axon and dendrites. Cytoskeletal networks are dynamic, adaptable, and are organized more like ant colonies and can respond rapidly by changing their structure and fractal dimension as required by external conditions for short periods or for a long period.

Here we need the Graph model model that can provide us a tool based on probabilities to compute the connectivity structure among the components in a network arising from cooperative and competitive interactions. We will breifly consider the graph model below for modeling the connectivity structure among agents..

Graph Connectivity Structure arising among Agents

The communication or interconnection patterns among the agents play a key role for applications to various modeling and simulation aspects of NISS. Three important statistical properties of the networks, namely average degree, characteristic path length and cluster coefficient, to be defined below are used as measures to distinguish the disordered networks such as:
(i) Random networks (ii) Power-law scaling networks, and (iii) Small World Networks.
from regular networks

MEASURES FOR DISORDERED NETWORKS

HOW WELL-CONNECTED YOU ARE?
***Average degree*:Sum over degrees/ of nodes /Number of nodes**

HOW COMPACT IS YOUR CONNECTION?
The *Characteristic path length L* measures the global property, namely, the average path length of the network.

HOW DENSE IS YOUR NEIGHBOURHOOD?
***The cluster coefficient* C is the average of C(i) ,where C(i) is defined by:**
Number of E(i) existing between k(i) neighbours of node i /Total number of possible edges .

Let us consider a finite graph G(V,E) where V is the set of n nodes and E the set of edges. Let us assume that the graph is represented as an adjacency matrix A with

A(i,j) =1, if there is an edge from node i to node j ; and A(i,j)=0, otherwise. We assume A(i,i) = 0 ,that is no self loops.

The following local and global parameters are derived from adjacency matrix:

(i)*Average degree*: $K = 1/n \sum_{i} k(i)$,and $k(i) = \sum_{j=1}^{n} A(i,j)$,or

k(i) is the degree of node , $0 \le K \le (n-1)$

(ii)The *Characteristic path length* L measures the global property, namely, the average path length of the network.

Given L(i,j) the shortest distance between nodes i and j,

L is defined by:

$L = 2/n(n-1) \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} L(i,j)$; $1 \le L \le (n-1)$

This is referred to as " global connectivity" in the context of lattice percolation, if we need to only infer whether they are connected or not. Thus the notion of percolation in lattice grid is closely related to the small-world and scale-free networks.

(iii)*The cluster coefficient* **C is the average of C(i) ,where C(i) is defined by:**

$$C(i) = 2\sum_{j=1}^{n}\sum_{k=1}^{n} A(i,j)A(i,k)A(k,j)/[k(i)k(i) - 1] =$$

Number of E(i) existing between k(i) neighbours of node i /Total number of possible edges [k(i)(k(i)-1]/2 .or, C=1/n $\sum_i C(i)$

Note that 0≤ C ≤1.

The above three properties serve as guidelines to roughly classify the three classes of disordered graphs:

(i) Random Network:

In Random network the degree distribution is a binomial or Poisson distribution in the limit of a large graph. Most of the nodes have the average degree and few nodes have more than average and few nodes have less than the average degree. Also L and C are small in random graphs.

(ii)Scale -free Network:

In this network, many nodes have a low degree (few links) and a few nodes have a high degree (many links). The distribution of the degree of the nodes has an unusual fat-tailed form or a power-law scaling property: namely the P(k) the degree distribution of a network is given by: $P(k) = k^{-g}$ where $2 < g < 3$.

This power-law degree distribution or scale-invariant property arises from two kinds of operations on a random graph.It has been experimentally observed that the biological networks have a range
$1 < g < 2.5$ and he social netwrks have a range $2 < g < 3$:
1. *Creating new nodes*: Growth of the graph by adding new nodes into an initial group of nodes as time progresses and
2. *Preferential attachment of Links*: The new nodes created are linked to old nodes, with a probability based on certain dominant properties the old nodes possess, e.g. the nodes having a higher degree (or attractiveness), chemical or physical interaction strength. In each case, the neighbourhood is appropriately defined as a conceptual graph. As the network grows the ratio of well-connected nodes to the number of nodes in the rest of the network remains nearly a constant,

Dorogovtsev et al. prove that the range 2 < g < 3 is crucial to have the following properties:

(a) Self-organization and (b) Resilience against random damage.

Also g is related to the fractal dimension; it has been for several types of networks: www, Actor, E *coli,* Scale free- tree and found to lie in the range 2 < g < 3.

(iii)Small-world graphs:

A graph is called a small -world graph, by Watts , if it exhibits the following two properties (when compared to a random graph of same number of nodes and average degree):

1. *Higher clustering coefficient C closer to unity:*

this implies that two nodes are more likely to be adjacent,if they share a common neighbour and

2. *Smaller average distance L between any two nodes:*

L scales logarithmically with the number of nodes. This measures a global property.

It results in a compact network- where the average length of the shortest directed path between two randomly chosen nodes is of the order of the logarithm of the size of the graph. This property ensures

that communication and interaction between distant neighbours can take place over long ranges. This is called the small world effect and the short path is also called a "worm-hole". Therefore, in agent based systems, where a very large number of agents are interconnected, the small-world network, can permit distant neighbours to interact with an efficient flow of information.

Assortative and Disassortative Mixing

In some networks ,the high degree nodes are connected to high degree nodes. These are called assortative or homophilic networks. In disassortative networks ,high degree nodes avoid being connected to high degree nodes.These two types of networks are dintingushed by using a degree-correlation coefficient that is positive for assortative networks and negatve for disassortative networks.
The assortative mixing results in larger positive Lyapunov exponents (eigenvalues) of the interacting matrix of the dynamical system. This means the system can quickly become unstable resulting in the

formation of giant components in graph networks or the phenomenon of percolation in a lattice.

In disassortative mixing high degree nodes avoid being connected to high degree nodes and result in a smaller positive Lyapunov exponent (or positive eigenvalues) and hence the dynamical fluctuation are not amplified and the system can reach stability more quickly.

It has been pointed out by Newman that such an assortative mixing is more prevalent among social networks, while disassortative networks seem to be common in biological networks. The assortative networks are less stable to random fluctuations leading to percolation like phenomena, while disassortative networks are more stable to fluctuations. The biological networks seem to self-organize themselves into assortative or disassortative networks according to their need to adapt themselves to their environment. We will illustrate this aspect in revisiting the ant swarms.

Stability aspects of the Graph dynamics

Consider the network of N nodes,each node i being associated with a variable x(i) . The time evolution of the graph can be represented by the equation:
$dX/dt = -X+F(X)$ where F is nonlinear. The equilibrium at a point X^* is determined by the Lyapunov exponents of the Jacobian that represents the interaction -this is a function of the size of the graph, nature of connectivity and rhe interaction strength between the nodes. As we observed earlier the fractal dimension is a function of the Lyapunov exponents or eigenvalues of the Jacobian at the equilibrium points and can change rapidly as the connectivity and interaction strength (e.g., attractive,repulsive) change, as for example in the bacterial colony. However, we can make the following observations that have been earlier suggested by Kauffman.
In a finite graph ,as the number of edges E increase with respect to the number of nodes N, in undirected graph, and $E/N > 0.5$ a giant connected component arises in the graph and as E/N increases beyond a threshold value 1, the graph becomes connected and cycles of all

length emerge ,indicating the presence of strange attractors of various cycle lengths (or ergodicity) indicating the presence of chaoticity of various kinds. For directed graph the all nodes get connected for E/N >1 and self-loops emerged when E/N>2. As the number of edges increase beyond this threshold,the shorter pathways emerge among the nodes and the mean distance among the nodes become shorter.In adition,many nodes have fewer links and few nodes have many links indicating the presence of scale-free distribution and the presence of self-loops and multiple loops interlock themselves so that there are minimal pathways among the nodes of the order of the logarithm of the number of nodes. However, clustering is related to percolation and is dependent upon the intrinsic geometry, as well as topology.

This reasserts the fact that a complex system lies in between the totally random system and the algorithmic system; the presence of numerous strange attractors and their basins draw the system to an equilibrium state rather quickly to adapt to the environment. Since the system is nonalgorithmic, there is no way we can control the time dependent parameters effectively.

Examples

(i) Metabolic networks

In metabolic networks the number of nodes is of the order of 800 and edges 3600; thus they contain many cycles and have a mean degree 10,node-to node distance 3;they are disassortative.

(ii) Protein Networks

In protein networks, the number of nodes are of the order of 2000, number of edges of the order 2000, the mean degree is 2 and node to node distance is 6.8, and cluster coefficient is 0.07, Newman [2004 .The degree distribution satisfies $P(k) = k^{-g}$ with g = 2.4 showing that they are scale-free, self-organizing and resilient against damge. Note that the ratio of cluster coefficient of protein networks to random networks is large and the average path length scales logarithmically with the number of nodes.

(iii) Neuronal Networks

The neuronal network develops in three phases:

1.In the first phase the different parts develop according to their own local programs.

2.In the second phase axons and dendrites grow along specific routes setting up a provisional, but orderly network of connections between different parts of the system.

3. In the final phase, the connections are adjusted and refined thro interactions among far-flung components in a way that depends upon the electrical signals that pass between them.

These indicate that the scale –free and small world networks play a role in the design of nervous system and Hebbian networks.

A. Swarm of Agents and self-organization

The multiset of agents paradigm to realise particle swarms has many advantages.

(i) The fundamental principle in swarm intelligence is cooperation and knowledge sharing. The multiset of agents approach can realise both exploration and exploitation by grouping the particles in different zones of the search space to set up cooperation and competition among members of same group and different groups. Using different grouping prevents possible quasi-ergodic behaviour that can arise resulting in the entrapment of the orbit in isolated regions, since the grouping amounts to using several Markov chains, with the initial states reasonably apart, in the search space.

(ii) Different types of interaction topologies (such as the wheel, the ring, the star and other lattice structures, can be assigned within and among different groups and their influence can be studied .

(iii) If we need to optimize several objectives the multiset of agents swarming can be useful .

(iv) Since there are different groups, evolutionary paradigm approach can be tried, by annihilating unfit populations and creating new populations.

The Ant Heuristics are based on the model of real ants finding optimal solutions to the shortest path between their nests and the food sources. Each ant leaves a trail of pheromone, thereby creating a new landscape so that other ants are attracted towards the food source by the scent. This problem is analogous to modifying a random graph that grows with time with preferential attachment to those strong pheromonal nodes leading to the food source from the nest. The optimal path turns out to be the path with a maximum density of the scent allowing for evaporation of the scent with time and degradation of the scent with distance. The communication among the ants take place through a shared landscape that establishes a link with the other agent nodes with the same belief and forming clusters.

To start with, the agents initialize their beliefs, by randomly linking for information.Then with time they update their beliefs through preferential linking to nodes with maximal pheromone intensity and forming local (groups) clusters. Then they modify the landscape further, by contacting other clusters to obtain collective intelligence, and reach an equilibrium state resulting in an optimal path from the nest to the food source.

The above search heuristic was simulated with a multiset of agents consisting of 100 groups each with 500 agents and the performance studied . In comparison to using a single group of 50000 agents, the multi-group performance was markedly better. Using multi-swarms rather than a single swarm and setting up competition within a swarm and between different swarms results in a more efficient heuristic in which preferential attachment happens to high degree nodes, leading to "*scale free distribution*".

In this set-up ; if some of the individual nodes or ants are randomly destroyed, the heuristic is not affected at all exhibiting the resilience of the scale-free distribution under random failure. However, if some of the hubs of well-connected group of nodes are selectively destroyed, the heuristic fails.

Also in this heuristic, since the initial tendency is to form local links, this results in higher clustering and if many nodes find the same nodes to link with no degree restriction, we obtain a correlation length smaller than the random network.

Simulation results show that the swarm network topology is very sensitive to the nature of interaction and threshold values, cost and aging of nodes. This emphasizes the fact that the algorithmic structure has broken down and the system is self-organizing exhibiting both the small-world and scale-free properties, or changing from one class to another, yet preserving self similarity. It is also possible for the networks to change their fractal dimensions and from assortativity to disassortativity and conversely, depending upon the environmental influence as in the bacterial colonies.

For practical applications we can use different scent types, different scent strengths, use special purpose agents for exploration and trail selection and allow non-cooperative and lazy agents to die. These depend upon the level of intelligence and nature of the agents.
Our estimate of the box-counting fractal dimension of the swarm lies between 1.7 and 1.9 .

B Simulating Animal-Human Trails

Active walker models in which animals or humans interact through indirect communication mediated by the environment and leave a purposeful trail either from nest to the food source or between other points of interest, can be interpreted qualitatively as a small world phenomena on graphs, or as percolation phenomena in lattices, by choosing a proper lattice structure (square or hexagonal) tiling the space.

These trails result due to agglomeration process of several walkers moving arbitrarily leaving markings (clearing vegetation or leaving chemicals), but eventually producing an attractive effect on another walkers. Clearly, there is a preferential choice among the possible trails and the most frequently used trails combine to become popular. Also rarely used old trails disappear and frequently used trails are reinforced although many new entry points may arise and destinations may branch off. Also fitness is evaluated for each trail as to its cost and utility .

C. Genetic Programming approach to Graph Growth

The problem of growing a graph until it reaches self-organized criticality through interaction is closely related to Genetic Programming (GP), Koza . In GP each program construct is a tree constructed from tasks, functions and terminal symbols.Then we perform crossover and mutation by swapping program sub-trees leading to feasible programs, taking care of the nature and type of the task. These operations resemble Metropolis-Hastings-Monte-Carlo methods , to create transitivity in a graph from a given node to a desired attractor node.The GP operations correspond to an ergodic move-set in the space of graphs with a given set of parameters and repeatedly generating the moves and accepting them with probability p or rejecting them with probability (1-p).

Suitable move-sets are: creation of new nodes, aging and annihilation of nodes, Mutation -movement of edges from one place to another, mating -swapping edges of the form (s,t),(u,v) to (s,u),(t,v), adding new edges based on a cost function. Such moves can create a phase transition (or percolation) to reach a global goal through successive local goals . An important aspect in GP is the fitness of the individual program generated locally and globally.In self-organization, ideally, one requires that the fitness is a *self-awareness* function i.e. the individual who does the work evaluates itself, ensuring that the global fitness is guaranteed. This is widely prevalent in Nature for activities such as: nest building (stigmergy), food searching (foraging).

D. Simulating Stigmergy

In Stigmergy (a term coined by French biologist Pierre-Paul Grasseis), two organisms interact indirectly through the environment. When one of them modifies the environment, the other responds to the new environment at a later time, e.g., nest building in insect societies. Starting with a basic pattern the nest grows by adding new material. The resulting structure produces different types of stimulus and responses from other members to build the nest further. This has all the features of a co- evolutionary algorithm, in the

sense it is a mutually bootstrapping process, that is driven by relative fitness (not absolute fitness). (i) We have a population of competing and cooperating individuals who exchange local information and interact. (ii)The shared fitness of the individuals piece of contribution is evaluated. (iii)Fitness based selection of sites for building the walls of a nest. (iv)Probabilistic variation of the nest based on builders.

The Wasp nests, are built by preferential attachment of a new part with the old assembly with probabilities 0.55, 0.06,0.39 respectively to fill in 3 walled hexagon, 2 walled hexagon, and 1 walled hexagon. That is, the sites are rank ordered and the local selection principle and preferential attachment ensures local fitness and promises global fitness and results in a scale invariant property for the shape. This whole process of nest building resembles a competitive game with a limited local information. The local fitness ensures global fitness because of the geometrical scale invariance.

Wasp Nest Evolution : Use of GP and Shape Grammar

The nest construction rules used by wasps,, can be simulated through Multiset of agent based Genetic programming. It seems that the insects and animals use their perceptive skill to evaluate the compactness of the resulting construction by using the Isoperimetric Quotient(IQ) to measure the fitness of their construction. In self-organization the nests are evalauted locally and that should ensure that it is globally fit . IQ tells us how to maximize surface area for a given perimeter in a 2-d object and how to maximize the volume in a 3-d object given the surface area. For a 2- dimensional object, the IQ is given by 4 p A/P^2 ,where A is the area and P is the perimeter (number of sides for a regular polygon). This ratio is a maximum equal to unity for the circle and its near approximants . Here we omit the constants and compute the reciprocal of IQ, namely, the square of the perimeter divided by area of the hexagon (assumed to be unity), which we call Quality Index (QI).

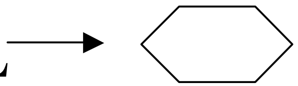
For a 3- dimensional object the IQ = $6 V p^{1/2} / S^{3/2}$ where V is the volume and S is the surface area. This is a maximum equal to unity for the sphere or closer or its closer approximants. This measure is useful when dealing with polyhedral grammars (termite mounds.

In self- organization, a global pattern emerges solely from the interactions among the lower level components of the system. The rules specifying the interactions among the components use only local information without reference to the global pattern.

In the following wasp-nest rules, we denote an assembly of hexagons by (p,q) where p is the number of sides in the assembly (perimeter), and q is the number of hexagons (area),QI=p^2/q. The graph isomorphic variants of these rules are omitted.

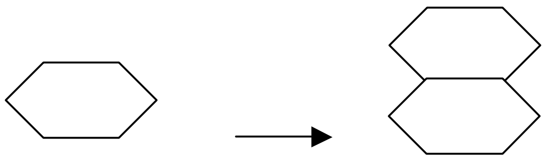
Rule 0: (Probability 1) **(6,1),QI= 36**

NULL

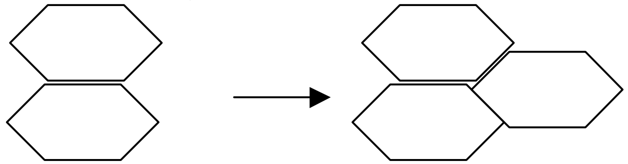


Rule 1:Probability 0.39

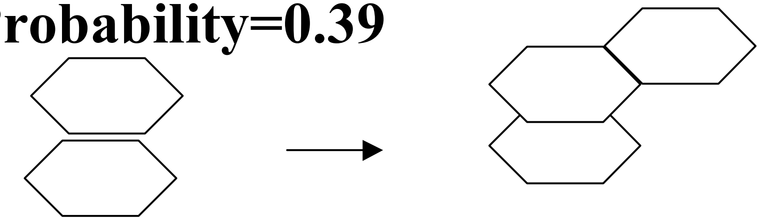
(6,1),QI=36 **(10,2),QI=50**



Rule 2a: (10,2), QI=50 (12,3),QI= 48
Probability = 0.06

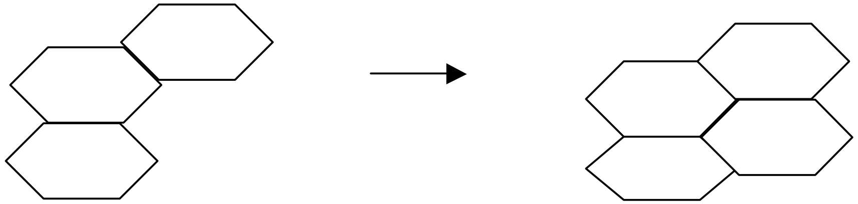


Rule 2b: (10,2),QI=50 (14,3),QI=65
Probability=0.39



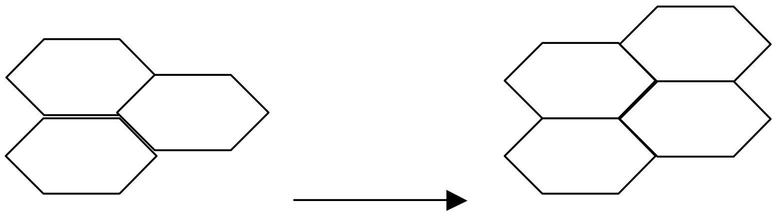
Rule 3a: (14,3),QI=65 **(14,4),QI=49**

Probability =0.55



Rule 3b:(12,3),QI=48 **(14,4),QI= 49**

Probability=0.06



Since the rules are based on local information, it is sufficient to consider sharing of 1,2, 3 edges among the hexagons. This is because each time a wasp leaves the best possible configuration with the least QI.; it will not leave an open configuration with 4 unfilled sides with perimeter 18 , QI= 81 or 5 unfilled sides with perimeter 22 with QI = 96. Another wasp would have completed the earlier configuration having a lower QI , since evolution prefers the higher probability rules 2 a, 3a and 3b .

This resembles Nash Equilibrium, in which each one of the several agents perform its action so that it is optimal, assuming that the other agents do not alter their own actions. Nash equilibrium arises out of merely local combination of myopic actions among agents, where each agent is happy with its own strategy as long as other agents do not deviate from their own fixed strategy.

This is not a particularly good scheme as the lack of communication makes explicit coordination and planning impossible. However, in large swarms communication bandwidth and cost constraints make stigmergy the only feasible option, since simple agents do not have communication devices and can only use diffusive signalling.

In wasp nests, the walls are corrugated to achieve self-similarity of the whole structure, since hexagons cannot tile the plane. Further, wasps use the preferential attachment rule that achieves both the self-similarity of the structure and geometrical compactness to have the largest possible area (volume) with the least perimeter (surface area).

E. Synchronization among agent population

A simplest adaptive system arises in 'Synchronization". Here two oscillatory systems (or repetitive systems) adjust their behaviours relative to each other so as to attain a state where they work in unison. This is a universal phenomenon. When two nonlinear oscillators interact, mode locking may occur whenever a harmonic frequency of one mode is close to a harmonic of other. As a result nonlinear oscillators tend to lock to one another so that one system completes precisely p cycles each time when another system completes q cycles (p,q integers). Agents provided with nonlinear oscillatory capabilities can couple through different choice of interacting functions. This kind of coupling can result in the emergence of dynamical order and the formation of synchronous clusters and swarming. Such SS can function like Chemical systems, Biological systems, Molecular machines and will have applications in designing task specific biomorphic robotics.

ENGINEERING SS

Since an SS is inherently nonlinear and hysteretic, agents made out of Piezoceramics, magnetostrictives, Shape-Memory Alloys (SMA), Electrostrictive polymers, Ferromagnetic SMA, can serve as components for specific applications. However, the central question in designing SS is how to program the components so that the system as a whole self organizes. This is the basic question addressed in the design of Amorphous computers and Spray computers, as well as in the synthesis of heterogeneous materials with certain specified macro properties from a knowledge of the properties of the microsturctures.

Emergence is a global behaviour (or a goal) that evolves from the local behaviours (goals) of components The evolutionary rules for arriving at the global goal is non-computable, since it cannot be expressed as a finite composition of computable deterministic function of local goals for any arbitrary problem domain. Thus we cannot design a general purpose programmable Smart system if we want to have an SS with exactly specified propeties.

However, for specific applications and a pre-defined interactive topology among the agents, the geometric, dynamical, topological parameters, and statistical properties can be obtained through simulation and tuned to build a specific SS.

Conclusion and Future Research

We described the important properties of a Nature -Inspired Smart System (NISS) and the models needed to understand, simulate and animate NISS.
Multi- agent systems in a network can exhibit the properties of both the computational and dynamical systems at the macro-level and can undergo phase transition and emerge into an NISS. However, to understand NISS quantitatively, we need to obtain the spatial structure of attractors and the temporal aspects of the trajectories, and topological or graph parameters. The first two aspects provide information about phase transitions and tells us whether a trajectory governed by a positive Lyapunov exponent falls in a given attractor.

The topological aspects give us a statistical behaviour of the network of connectivity among the agents. Real systems including biological systems are nonstationary, in which the parameters are changing with time. Thus quantitative understanding of an NISS using the above parameters with arbitrary time varying interactive topology seems

difficult. These difficulties arise due to the statistical nature of the models available and inability to compute the dynamical parameters.

Protein-based computational networks have an important role to play in the design of NISS- particularly, in Genetic regulatory systems, Prosthetics, Systems Biology and understanding the organization of the living cells. Simulation and animation have important roles to play both in research and teaching in these areas, although *in silico* simulation is never a good substitute for *in vivo* studies.

Modelling power and decision power work in opposition !
Chaotic sequences are as we know have greater MODELLING POWER (more complex) than Type 0 (Turing Machine) grammar. Working with them is difficult indeed - since the decision power is weak!
Self organization is at a higher level of hierarchy .
It has higher modeling power than a Turing machine in producing high adaptability and efficiency, rather than programmability!
Programmability and adaptability seems to be mutually exclusive and whenever maximum efficiency exists adaptive systems can do better.

Real systems including biological systems are nonstationary, in which the parameters are changing with time. Thus quantitative understanding of a Smart system using the above parameters with arbitrary time varying interactive topology seems not possible. These negative results arise due to the statistical nature of the models available and inability to compute the exact parameters numerically. However, for specific applications and a pre-defined static interactive topology among the agents, the statistical parameters can be obtained through simulation and tuned to build a special purpose SS.

Quantum Mechanics, Smart System and Artificial Life

We assumed that the real world is at the macro scale while simulating with agents. All our arguments are not valid below micro -nano-and molecular scales . Here quantum mechanical effects (QM)takes over . The geometry, nature on interaction and other details are not easy to model by classical computational tools.

1.In QM there is no chaos and no trajectories and no time arrow . Quantum mechanics is a linear theory and no chaos exists. There is no exponential time divergence of trajectories.The quantum mechanical view is time -reversible. However, in classical chaos, the existence of positive Lyapunov exponent prevents the recovery of initial state and so there is a time-arrow and our smart systems will age.

2. No exponential complexity:
Since there is no chaos and no Lypaunov exponents, metric entropy is not definable and quantum world is computational complexity-free. All problems are solvable in a polynomial time -no exponential time problems. Quantum mechanics seems to contain randomness in itself.

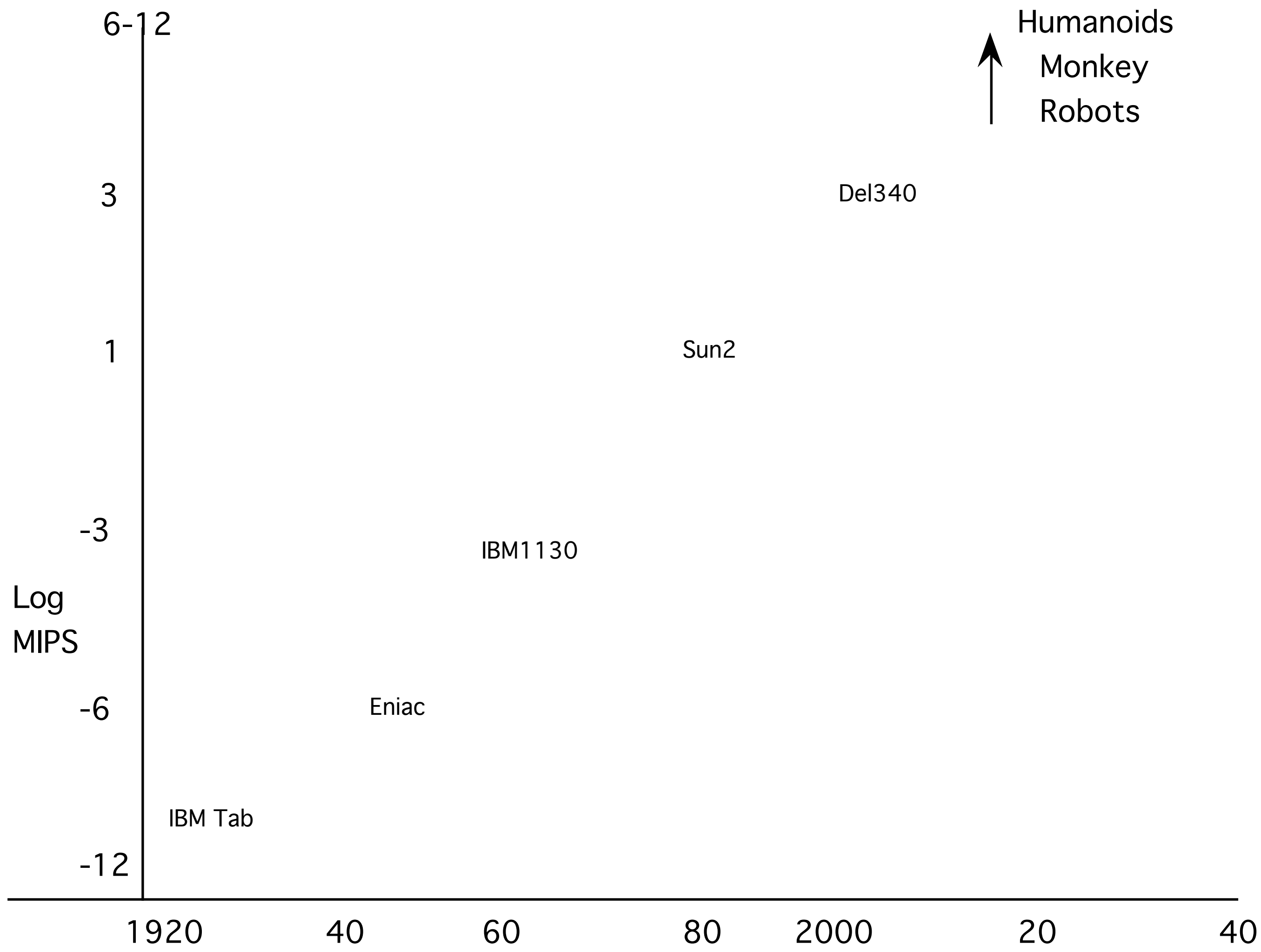
3.Built-in-ergodicity:
Wheras a classical agent will travel precisely through a specific deterministic path, if there is no added randomness, inherently a quantum mehanical agent explores all posible pathways in space time to reach a given final point producing an exponentially growing information. This means the Quantum mechanical agent is endowed with the ability for exploration and exploitation without adding any randomness- a kind of self-awareness!

4. Quantum Systems exhibit Emergence and Decoherence:
Quntum mechanics always works on a holistic view . The quantum system exhibits both kinematic and measurement-induced nonlocality. In a composite system composed of parts and the state of whole is not definable in terms of the states of its parts. There are quantum correlation between different subsystems due to the superposition principle. In fact, the whole is more than its parts, even if we assume there is no interaction among the subsystems (non-separability). Quantum sytems already exhibits emergence leading to many different views. Decoherence provides a particular view.

5. Quantum Systems are not programmable due to inherent failure of modularity.Does it imply that they are always adaptive?

Thus in QM, all computations are complexity -free, all QM agents have a built in self-awareness to use exploration and exploitation for solving problems, emergence is always present,adaptive, and the smart systems remembers past for ever and, never age!

The study of complex networks is in its early stage. The interaction of agents within this framework is very complex and requires the availability of tools for creating a very large number of agents and dealing with them in a suitable environment. This would provide us an insight into the behaviour of natural systems we see around us.


Humanoids
Monkey
Robots
6-12
3
1
-3
-6
-12
Log
MIPS
Del340
Sun2
IBM1130
Eniac
IBM Tab
1920
40
60
80
2000
20
40