(Chapter 7 in)
Computational Intelligence for Agent-based Systems
(Studies in Computational Intelligence)
Vol.72, (Hardcover) by [Raymond S.T. Lee](#) (Editor),
[Vincenzo Loia](#) (Editor), Springer verlag, New York,
September 2007,
ISBN-10: 354073175X, **ISBN-13:** 978-3540731757

Interacting Agents in a Network for *in silico Modeling of Nature-Inspired Smart Systems**

V.K.Murthy ¹, E.V.Krishnamurthy ²

¹School of Business Information Technology, R.M.I.T University ,
Melbourne ,Victoria 3000, Australia
kris.murthy@rmit.edu.au

²Computer Sciences Laboratory, The Australian National University,
Canberra, ACT 0200, Australia.
abk@mail.rsise.anu.edu.au

CONTENTS

Abstract.

- 1 Introduction
- 2 Multi-set of Agents Transactional Paradigm
 - 2.1 Operational Aspects of Multiagent System
 - 2.2 Interaction among Agents
 - 2.3 Concurrency and Conflicts
3. Agent-based Modeling and Simulation
 - 3.1. Advantages of Agent based simulation
 - 3.2 Special Purpose Agents in Simulation
 - 3.3 Agents and Avatars in Animation
 - 3.4 Agent-based Models in Development Biology
 - 3.5 Cell Motility
 - 3.6 Pattern formation Mechanisms
 - 3.7 Signal Processing among Cells
- 4 Role of Fractal and Percolation Models in NISS
- 5 Injecting Positive Entropy
- 6 Graph Connectivity Structure arising among Agents
 - 6.1 Random, Scale-free and Small World Networks
 - 6.2 Assortative and Disassortative Mixing
 - 6.3 Stability Aspects of the Graph Dynamics

6.4	Examples
7	Nature -Inspired Smart Computations
7.1	Muliset of Agents Based Ant Colony Heuristic
7.2	Smart Bacterial Colony
7.3	Simulating Animal-Human Trails
7.4	Simulating Attractor Dynamics using Exploration
7.4.1	Bistability in Perception
7.5	Oscillatory Chemical Reactions
7.6	Graph Growth Models
7.7	Stigmergy and Games
7.7.1	Wasp Nest-like Growth
7.7.2	Cell diversification through Inductive Interaction
7.8	Synchronization among Agent Population
8	Engineering NISS
8.1	Quantum Mechanics and Smart System
9	Multi-agent Architecture and Toolkits
10	Summary and Directions for Research
	References

Interacting Agents in a Network for *in silico** Modeling of Nature -Inspired Smart Systems

V.K.Murthy ¹ , E.V.Krishnamurthy ²

¹School of Business Information Technology, R.M.I.T University , Melbourne ,Victoria 3000,
Australia

kris.murthy@rmit.edu.au

²Computer Sciences Laboratory, The Australian National University, Canberra, ACT 0200,
Australia.

abk@mail.rsise.anu.edu.au

Abstract. An interacting multi- agent system in a network can model the evolution of a Nature - Inspired Smart System (NISS) 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. 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 NISS. Further developments of agent technology will be of great value to model, simulate and animate, many phenomena in Systems biology- pattern formation, cellular dynamics, cell motility, growth and development biology, and can provide for improved capability in complex systems modelling. Also agents will serve as useful tools to model, design and develop biomorphic robots and neuromorphic chips.

**in silico = In or by means of a computer simulation; in silico* is closely based on two older Latin phrases that are key terms in the jargon of every biologist and biochemist: *in vivo* and *in vitro*, both of which came into use at the end of the nineteenth century. The first translates as “in that which is alive”, and refers to some experiment carried out within a living organism, such as a drug test on an animal. The second means “in glass” and is used for experiments that take place in an artificial environment, such as a test tube or culture dish (WIKIPEDIA)

1 Introduction

The study of computation and intelligence on the one hand, and the study of life processes and the related conscious thinking on the other, have been engaging philosophers and scientists since ancient times. The phenomenal developments in computer science, robotics, physical, biological, neuro-sciences and theoretical analysis of the properties of complex systems, have recently shed new light on the different aspects and have established very strong links between these disciplines resulting in the Science of Nature-Inspired Smart Systems (NISS). The main thrust of research in this area is to explore how insights from these studies may have positive and beneficial impact in the design of newer computational models and associated techniques. NISS is a highly interdisciplinary area and the definitions have not yet been well-formulated and vary between fields, although the basic properties of these system are nature-inspired.

From the point of view of Physics, NISS are ensembles whose dynamics can be monitored or modified by distributed sensors and actuators, in accordance with an integrated control law to accommodate time-varying exogenous inputs or changing environmental conditions. NISS exchanges energy, entropy and matter with the environment. On an average NISS exports entropy, it operates far from equilibrium, beyond a critical distance from the equilibrium state, positive and negative feedback are present and the system is nonlinear.

In Engineering, an NISS is one that monitors itself and/or its environment in order to respond to changes in its conditions. This can result in (i) Adaptation (ii) Self-repair and (iii) Optimisation of resources.

Biosystems characterize NISS by the properties:

1. The processing of information is distributed and is monitored
2. The communication system is reconfigurable.
3. The system is highly adaptive and acquires knowledge through interaction with its environment.
4. The energy consumption is optimized.
5. The system is self-organizing; it exhibits emergent properties which are not predictable, in advance from the properties of the individual local interactions among its components and the environment.

Here, we define NISS with the following four properties that are reflected in each of the above viewpoints.

1. Interactive, Collective, and efficient

An NISS consists of a large number of components and interacting with the environment. They collectively and cooperatively perform actions, coordinating their actions if there is competition, to obtain maximal efficiency.

2. Self-organization and emergence

The total dynamic behaviour arising due to cooperation between different parts of the NISS can lead to a coherent behaviour of the entire system that can change by a large amount when the values of a control parameter changes by a small amount

(sensitivity). These changes have similarities to the phase transition encountered in physics, where local changes result in a global change in which new properties emerge abruptly. These new properties of the NISS are not predictable, in advance from the properties of the individual interactions. In particular, under emergence, the many degrees of freedom arising due to its component parts collapse into a fewer new ones with a smaller number of globally relevant parameters. This means the properties of self-organization cannot be predetermined and the system evolves on a qualitatively new level with respect to its global constraints.

3. Power-Law Scaling or Scale-Invariance Property

Power-law scaling relation arises for the emergent properties of the NISS. Dorogovtsev and Mendes [27] prove that this property is essential for fault resilience and self-organization.

4. Adaptive, fault-tolerant and Resilient against damage

NISS are always flexible to change, Rose and Lauder [82]-they can self-modify their past behaviour and adapt to environmental changes, available resources, as well as, tolerating failures or non-cooperation of some of their components. We can include mutation and reproduction in this property.

It appears that these four properties are interrelated to self-organization, stigmergy and self-assembly, Boncheva et al. [15] Minelli [67], 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, Serugendo [85], Serugendo et al. [86], Serugendo et al [87].

This chapter describes the application of a Multi-set of Agents Paradigm (MAP) to model, simulate and animate NISS (Nature-Inspired Smart Systems) and related computational schemes: for example, Genetic algorithms, Goldberg [35], Michalewicz and Fogel [66], Genetic Programming, Koza [53], Immunocomputing, Gonchorova et al [36], Stepney et al [92], de Castro and Timmis [25], Self-organized criticality, Morovitz [70], Camazine et al. [20], Guerin and Kunkle [40], and Active Walker models (ants with scent or multiwalker-paradigm, where each walker can influence (repel or attract) the other through a shared landscape based on probabilistic selection, Bonabeau et al. [14], Chu et al. [23], Dorigo et al. [26], Kennedy and Eberhart [50], Kennedy [51], Biomimicry, Benyus [10], Pacino [76], Bio-inspired robotics and computing, Bar-Cohen and Breazeal [7], Kumar and Bentley [57], Bullock [18], Werfel et al. [108], Ayers et al. [4], Brownian Agents, Ebeling and Schweitzer [28], Game-theoretic problems, Osborne and Rubinstein [75], Fudenberg and Tirole [32], and *in silico* integrative biological modeling, McCulloch and Huber [64], Keele and Wray [49].

Three crucial properties of Agents make them suitable for the simulation of 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 order to realize the above computational schemes using the Multi-set of Agents Paradigm (MAP) we need to consider the following issues:

1. What kind of mathematical models are needed to study NISS and for implementing within the framework of MAP?

We discuss three interrelated mathematical 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. These models have been very well studied in recent years and are closely related to the properties such as: self-similarity, scale-free, resilience to damage, self-organization and emergence.

2. Is the emergence of NISS 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 using MAP?

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

Section 2 describes the multi-set of agents-based paradigm (MAP). In Section 3 we describe the general strategies for agent-based modelling and the role they can play in modeling, simulation and animation with examples from cellular biological systems. Section 4 describes the fractal and percolation models. Section 5 explains the role of positive metric entropy condition in NISS. Section 6 outlines the statistical properties of disordered networks and their role in NISS. Section 7 deals with the macro-level simulation of NISS; these are illustrated by a few examples-ant colony heuristic based on group swarming, Bacterial colonies, human-animal trails, Chaotic attractors, Bistability in perception, Oscillatory chemical reactions, Graph-growth, and Stigmergy. Section 8 briefly considers the engineering of NISS, the role of scales, and our inability to simulate nano and molecular levels due to the limitations arising from quantum mechanics. Section 9 provides a brief description of currently available agent-tool kits. Section 10 summarizes the results and the directions for research in this area.

2 Multi-set of Agents Transactional Paradigm

The AOIS (Agent-Oriented System Community) defines an agent as 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, Woolridge [111], Tianfield [101], Lucena et al [60], Zhang and Zhang [114], Hopgood [44]. The multiset-agent system consists of several single agent-systems, Fig.1. Thus if N agents are involved $i = 1, 2, \dots, N$, each of the agents will be denoted with a label (i) . Since the term “Agent” is used in a variety of ways in Artificial intelligence and social sciences, we will restrict ourselves to the definition that the agent is a software module having the transactional programming model with all the important features-such as: atomicity of commitment, short duration transactions, sensing and aborting, and its own thread of control. Further, we will not use psychological terms such as intention and desire, although we will be using the term “revision of belief” in the sense of rewriting (updating or deleting) existing information (or knowledge) in its memory state.

(1) Worldly states or environment U:

All those states which completely describe the universe containing all the agents.

(2) Percept: Depending upon the sensory capabilities (input interface to the universe or environment) an agent can receive from U an input T (a standard set of messages), using a sensory function Perception (PERCEPT): $PERCEPT : U \rightarrow T$.

PERCEPT can involve various types of perception: see, read, hear, smell. The messages are assumed to be of standard types based on an interaction language that is interpreted identically by all agents. Since U includes both the environment and other agents the input can be either from the agents directly or from the environment that has been modified by other agents. This assumption permits us to deal with agents that can communicate directly, as well as, indirectly through the environment as in active walker model; this is called “stigmergy” in Ant colony where one ant can modify its environment and affect the behaviour of another ant (see also EFFECT).

(3) States of Mind M:

The agent has a mind M (essentially a problem domain knowledge consisting of an internal database D for the problem domain data and a set of problem domain rules P) that can be clearly understood by the agent without involving any sensory function. Here, D is a set of beliefs about objects, their attributes and relationships stored as an internal database and P is a set of rules expressed as preconditions and consequences (conditions and actions). When T is input, if the conditions given in the left-hand side of P match T , the elements that correspond to the right-hand side are taken from D , and suitable actions are carried out locally (internally in M) as well as externally on the environment.

The nature of internal production rules P , their mode of application and the action set determines whether an agent is deterministic, nondeterministic, probabilistic or fuzzy. Rule application policy in a production system P can be modified by:

- (1) Assigning probabilities/fuzziness for applying the rule
- (2) Assigning strength to each rule by using a measure of its past success
- (3) Introducing a support for each rule by using a measure of its likely relevance to the current situation.

The above three factors provide for competition and cooperation among the different rules. Such a model is useful for many applications, e.g., Active -walker, Self-organization and swarm models, Chemotaxis. Accordingly, we assume that each agent can carry out other basic computations, such as having memory, simple addition capability, comparison, simple control rules and the generation of random numbers, Pacino [76]. These mechanisms enable us to simulate tumbling, as well as running of organisms for foraging.

Figure 1. Structure of an Agent

(4) Organizational Knowledge (O): Since each agent needs to communicate with the external world or other agents, we assume that O contains all the information about the relationships among the different agents, e.g., the connectivity relationship for communication, the data dependencies between agents, interference among agents with respect to rules and information about the location of different domain rules.

(5) INTRAN:

On the receipt of T, the action in the agent M is suitably revised or updated by the function called Internal transaction (INTRAN).

Revision: Revision means acquisition of new information about the environment, that requires a change in the rule system P. This may result in changes in the database D. In a more general sense, revision may be called “mutation” of the agent since the agent exhibits a mutation in behaviour due to change of rules or code, Boloni et al [13].

Example: The inclusion of a new tax-rule in the Tax system.

Update: Update means adding new entries to the database D; the rules P **are not changed**. In a more general sense, the update may be called “Reconfiguration” since the agent’s behaviour is altered to accommodate a change in the data.

Example: Inclusion of a new tax -payer in the Tax system. Both revision and update can be denoted in set-theoretic notation by:

INTRAN: $M \times T \rightarrow M(D,P)$

Both mutation and reconfiguration play important roles in Nature-inspired computing. These are achieved in our model by introducing changes in D and P as required. This can be interpreted as updating or revising a set of database instances. Hence, if one or several interaction conditions hold for several non- disjoint subsets of objects in the agent at the same time, the choice made among them can be nondeterministic or probabilistic. This leads to *competitive parallelism*. The actions on the chosen subset are executed atomically and committed. In other words, the chosen subset undergoes an 'asynchronous atomic update'. This ensures that the process of matching and the follow-up actions satisfy the four important ACID properties: Atomicity (indivisibility and either all or no actions or carried out), Consistency (before and after the execution of a transaction), Isolation (no interference among the actions),Durability (no failure). Once all the actions are carried out and committed the next set of conditions are considered.

As a result of the actions followed by commitment, we may revise or update and obtain a new database for each agent; this may satisfy new conditions of the text and the actions are repeated by initiating a new set of transactions. These set of transformations halt when there are no more transactions executable or the databases does not undergo a change for two consecutive steps indicating a new consistent state of the databases.

However, if the interaction condition holds for several disjoint subsets of elements in the database at the same time, the actions can take place independently and simultaneously. This leads to *cooperative parallelism*; e.g. vector parallelism, pipeline parallelism.

(6) EXTRAN: External action is defined as an external transaction (EXTRAN) that maps a state of mind and a partition from an external state into an action performed by the agent. That is: EXTRAN: $M \times T \rightarrow A$. That is, the current state of mind and a new input activates an external action from the action set A.

(7) EFFECT: The agent also can affect U by performing an action from a set of actions A (ask, tell, hear, read, write, speak, send, smell, taste, receive, silent), or more complex actions. Such actions are carried out according to a particular agent’s role and governed by an etiquette called protocols. The effect of these actions is defined by a function EFFECT that modifies the world states through the actions of an agent:

EFFECT: $A \times U \rightarrow U$;

EFFECT can involve additions, deletions and modifications to U. Thus an agent is defined by a set of nine entities, a 9-tuple:

$(U,T, M(P,D),O,A,PERCEPT,INTRAN,EXTRAN,EFFECT)$.

The interpreter repeatedly executes the rules in P until no rule can be fired.

Thus, we can interpret all the abstract machine models (finite state machines, state charts, Turing machines) and concurrent computational models (Petri nets) as subclass of the agents, by a suitable choice of the entities in the 9-tuple.

The multiset of agents paradigm (MAP) consists of the following features Murthy and Krishnamurthy [71], Krishnamurthy and Murthy [56]:

(i) A multiset M that contains evolving agents (called the agent-space) whose information is structured in an appropriate way to suit the problem at hand. Conventionally the elements of the multiset are passive datastructures, Krishnamurthy et al [55]. Here each element of the multiset is an agent. Thus apart from the advantages resulting from the use of intelligent agents as elements, we have a strong theoretical background from the abstract rewriting systems on multisets, Suzuki et al. [97].

(ii) A set of interaction rules that prescribes the context for the applicability of the rules to the agents. Each rule consists of a left-hand side (a pattern or property or attribute) describing the conditions under which the agents can communicate and interact, and a right hand side describes the actions to be performed by the agents, if the rule becomes applicable, based on deterministic, fuzzy or probabilistic criteria, Murthy and Krishnamurthy [71].

(iii) A control strategy that specifies the manner in which the agents will be chosen and interaction rules will be applied, the kinetics of the rule- interference (inhibition, activation, diffusion, chemotaxis) and a way of resolving conflicts that may arise when several rules match at once.

(iv) An agent by itself or another coordinating agent evaluates the performance of the agents to determine the effectiveness of rule application. This agent ensures that the contract among the different agents hold; if the contract fails, the coordinator can rescue, abort or restart as in i-Contract or in Eiffel, Meyer [65].

(v) Interaction -Based: The computations are interpreted as the outcome of interacting agents to produce new agents (or same agents with modified attributes) according to specific rules. Hence the intrinsic (genotype) or genetic constitution of an organism and acquired properties due to interaction (phenotype) with the environment can both be incorporated in the agent space. Since the interaction rules are inherently parallel, any number of actions can be performed *cooperatively or competitively* among the subsets of agents, so that the new agents evolve toward an equilibrium or unstable or chaotic state.

(vi) Content-based activation of rules: The next set of rules to be invoked is determined solely by the contents of the agent-space, as in the context of chemical reactions.

(vii) Pattern matching: Search takes place to bind the variables in such a way to satisfy the left hand side of the rule. It is this characteristic of pattern (or attribute) matching that gives the agent-based paradigm its distinctive capabilities for nature-inspired computing.

(viii) Choice of objects, and actions:

We can several types of objects, as the basic elements of computation to perform suitable actions on them by defining a suitable topology, geometry or a metric space.

2.1 Operational Aspects of Multiagent System

A multi-agent system is realised as a loosely coupled network of single agents shown in Figure 1; they interact among themselves and through the environment to solve a problem. Operationally, the multiagent system uses certain protocols having the following features or their variants:

1. There is a seeding agent who initiates the solution process.
2. Each agent can be active or inactive.
3. An active agent can do local computation, send and receive messages (through message-passing or shared memory) and can spontaneously become inactive.
4. An inactive agent becomes active, if and only if, it receives a message
5. Each agent may retain its current belief or revise its belief as a result of receiving a new message by performing a local computation. If it revises its belief, it communicates its revised state of belief to other concerned agents; else it does not revise its solution and remains silent.

2.2 Interaction among Agents

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. These interference rules are similar to those enunciated by Turing [103] to describe the development of shape, form and pattern in organisms (chemical morphogenesis rules). 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).

Example: In cellular signal processing proteins whose conformation can be altered by the equilibrium binding of a regulatory ligand, switches between active and inactive states when the concentration of the regulator varies. Also situations such as- two ligands whose binding sites in a protein are coupled can reciprocally affect each others binding or proteins that can simultaneously bind two ligands - can be modelled by ED and ID, Graham and Duke [38].

Note that various types of Petri net models, Krishnamurthy [54], are used currently in Systems Biology, Cardelli [21], Pinney et al. [79], Effroni et al [29], Harel [41], Aviv and Shapiro [3]. The agent models are more general and provide greater flexibility and also help in animation.

2.3 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 can have several attractors or equilibrium states. It is also possible that the evolution of the agent system is sensitive to initial conditions leading to chaos and self-organization.

Conflicts: Resolution or compromise? While using agents for modelling behaviour, under concurrency, the conflicts arising in INTRAN and EXTRAN may require resolution or to an agreeable compromise. This situation can arise in animation and robotics. 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, Murthy et al. [72]

3. Agent-based Modeling and Simulation

Agent-based modelling 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, their interaction scale becomes crucial : Molecular (10^{-12}), Nano (10^{-9}), Micro(10^{-6}), Macro (10^{-3}) (all in meter). Hence agent-based modeling needs to be carefully designed. Further, any simulation process (and a model) should not be treated as an end product. They can only be regarded as tools to gain insight rather than a true representative of the real and experimental validation should be followed up.

In the molecular scale, atoms and molecules are autonomous objects and their interactions are as in chemical 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 and a homogeneous media. It is only in the macro scale, the organization is less autonomous and can be controlled by other external factors (man, animals and their control strategies).

In macro-scale 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, Watts [104], Strogatz [95], Dorogovtsev and Mendes [27], Barabasi et al.[6], Ben Naim et al [9], Newman [73], Zaslavsky [113].

To realize agent-based simulation, therefore, the agents need to strategically control their links with other agents based upon some policy. These policies result in nonlinear

dynamical systems and hence we need to consider the fractal and related models as the basis.

3.1. Advantages of Agent based simulation

(i) Traditional simulation ignores layers of complexity to achieve a simplified description of overall complexity. Agent-based simulation (ABS), Sichman et al. [89] starts with simple rules of interaction among the individual components that drive the system to the complex behaviour observed. It works bottom up by examining what low-level rules and what kind of heterogeneous, autonomous agents are required to synthesize the required higher level behaviour. Thus, ABS is useful for realising conventional algorithms, Evolutionary algorithms, Genetic algorithms, Genetic Programming, Immunocomputing, Self-organized criticality, Active walker models where each walker (e.g. ants with scent) can influence (repel or attract) other walkers using a shared landscape.

(ii) Agent-based simulation enables us to make predictions about the processes occurring at the intermediate mesoscopic scale due to the interplay between the microscopic dynamics and the macroscopic environment.

(iii) The simulation of deterministic, chaotic and stochastic systems are possible.

(iv) Different models where iterated application of simple rules can generate complex forms can be studied.

(v) Agent-based hybrid simulation (for soft and hard computation) can be used to study the global behaviour of a complex adaptive system from the local interactive behaviour of its components. The fundamental building block of most models of a complex adaptive system is the adaptive autonomous agent. An adaptive autonomous agent is characterized by:

1. Interaction with the environment
2. Goal driven motivation
3. Intelligence
4. Adaptive dynamics

Also an agent's goals can take various forms:

1. Desired local states
2. Desired end goals
3. Selective rewards to be maximized

3.2 Special Purpose Agents in Simulation

Agents can be designed to realize special purpose functionalities. These functionalities depend on the problem domain. For example, in information technology, we need collaborative agents that compete or cooperate; interface agents that act as personal assistants; mobile agents that migrate among hosts to enhance efficiency of computation and improve the network throughput; information agents that manage,

manipulate and collate information from many distributed sources; reactive agents that respond to stimulus and respond in an environment where they are embedded; smart agents that learn from their actions.

In building cellular biological models based on agents, however, it will be more convenient to use special purpose agents –such as sensor agents, goal agents, skill agents and compose these modules to build reusable, composable, blendable behaviours. In this context, we may recall that in cell system biology, the protein networks organize themselves into special purpose proteins thus : (Alberts et al [1], Cardelli [21], Wooley and Lin [110], Lauffenberger et al.[58]).

1. Relay proteins: simply pass the message to the next signalling component
2. Messenger proteins: carry the signal from one part of the cell to another.
3. Adaptor proteins ; Link one signalling protein to another
4. Amplifier proteins: Increase the signal levels causing a signal cascade
5. Transducer proteins: Convert the signal from one for to another
6. Bifurcation protein: Spread the signal from one pathway to another
7. Integrator proteins: Receive two or more signals and integrate them and transmit them.
8. Latent gene regulatory proteins: These are activated at the cell surface by activated receptors and then migrate to the nucleus to stimulate gene transcription.
9. Modulator proteins: These modify the intracellular signalling proteins and thereby regulate the strength of the signal along the pathway.
10. Anchoring proteins: Maintain a specific signalling proteins at a precise location in the cell by tethering them to a membrane of the cytoskeleton.
11. Scaffold proteins: These are adaptor or anchoring proteins that bind multiple signalling proteins together in a functional complex and often hold them at a specific location.

In animating protein -based computing, it is required to perform simple self-assembly, in which each agent has one or more binding sites to which ligands (a molecule that binds to specific site on a protein) can bind , when they have complementary matching shapes. Once they match they can either inhibit or enable the action of the agents deterministically or stochastically. Signalling proteins can bind to one another in multiple combinations like Lego-bricks with the protein forming a 3-Dimensional network of interactions that determines the route followed by the signalling pathways. Agent-based animation can be helpful to understand specific sequence of reactions. Also the above special purpose agents can be useful.

3.3 Agents and Avatars in Animation

Agents consist of information objects and an associated script that knows what to do with the information and how to deal with the environment. They behave like actors and have intentions and actions. Agents are autonomous and they have a built in control to act, only if they want to. In addition, agents are flexible, proactive and have multithreaded control. The above aspects make agents suitable for reactive animation.

In reactive animation, Efroni et al.[29], Harel [41], Szarowicz et al. [98], Szarowicz et al. [99], WavesWorld [105], 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.

Expansion of simulation to animation can be 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, Alberts et al. [1]. These are usually illustrated in static form as a series of steps. Animation can introduce each step individually and in order to emphasize multiple effects of one protein and the cellular location of each effect.

Example. The Wnt pathway begins with the extracellular ligand Wnt binding to its membrane receptor (frizzled) followed by activation of the receptor. The receptor in turn activates the intercellular protein disheveled which then inhibits glycogen synthase kinase 3 (GSK-3). Once GSK-3 is inhibited, it can no longer induce the destruction of beta-catenin. Stith [93].

This pathway is represented by using the symbols ~E(enable);~I (inhibit):

Wn ~E frizzled ~E Disheveled ~I GSK-3 ~I Beta-catenin

In static display this pathway will be shown, as though, Wnt leads to inhibition of beta-catenin, while in animation we can see the full reactive process.

The animation emphasises the order of events, the binding of ligands and the accompanying reactions and the movements along with the timing of the events. Ideally, such a realization would require a total manifestation of the real object and the events that are occurring in time and the reactions of the associated objects. This can be achieved through the introduction of “Avatara” which is realised as an extension of the Agent model (Figure 1) by including a body with a desired geometry and related functions. The body provides for the visualization part of the events. The actions or animation on the body are carried out by a scheduler that is controlled by one or more agents. In this sense Avatara provides a visible part of the environment; however, it cannot perceive or react to events on its own.

Remark:

In Hindu scriptures, Avatara is a manifestation of a personality suited to the time, task, circumstance and the need. It assumes a form or shape in accordance with the purpose or goal in hand. A partial realisation of the personality (or a projection) of Avatara is called “Amsa”. Avatara requires a body besides the mind of an agent, to provide the visual realisation of its actions.

3.4 Agent-based Models in Development Biology

In development biology several basic kinds of models are studied Keele and Wray [49]:

1. *Subcellular models*: Here differential equations of enzyme production and utilization form the basis.
2. *Cellular interaction*: These involve exchange of information between neighbouring cells
3. *Cell development*: Five important properties possessed by cells during development are studied.
 - (i) Cell division: This is important for two reasons: It provides a motive force for differentiation which normally occurs after cell division. Also the increase in the number of cells changes the morphology of the whole organism and so may affect the input received by any cell.
 - (ii) Genetic information: The control at the level of genes in higher organisms is largely unknown. This is an important area of study
 - (iii) Ability to sense information from environment and neighbouring cells. Cells often need to receive information from their neighbours in order to differentiate. The positional information may be in the form of a simple signal molecule or may involve a complex signal field, where cells at different positions differentiate according to the signal received at that point. We do not understand, as yet, how such fields are produced and maintained.
 - (iv) Ability to change state (differentiation): any state change can be termed as differentiation and the differentiation defines the changes within the cells (mainly at chromosome level) which eventually result in observable differentiation. Differentiation may involve shape changes, production of signals transmitted to other cells, or production of a new biochemical not previously synthesized by the cell.
 - (v). Cell movement: This occurs passively, as cells grow and divide; but many types of cells have genuine motile ability. This can be by means of pseudopoidal extensions by which cells drag themselves along. A second form of movement involves cytoplasmic streaming within the cell.

3.5 Cell Motility

Many animal cells possess the fundamental ability to crawl ; this form of locomotion is essential for morphogenesis and wound healing. This is done through: protrusion of cell's leading edge, adhesion of this edge to the substratum, forward translocation of the cell body, detachment of the cell's rear edge and retraction of the cell's rear edge. The front part of the migrating cell is called "Lamellipod". This is the basic engine for gliding and crawling- the cell body and the rear are mechanically passive. The engine

is powered by polymerization of a monomer for expansion and depolymerization for contraction. Such protein based engines can be useful to build biomimetic micro-robots, softwiring among neural cells in order to metamorph the neural connections, Jacobson [47].

Further developments in agent technology may provide for agent creation, destruction, motility and migration in virtual space, for modelling cell-biology.

Boahen [12] describes how the axon toed by an amoeboid structure at its tip senses chemical gradients laid down by neural communication cells , thus guiding the axon to appropriate locations. Agent-based animation is useful to view this phenomenon.

Cell movement behaviour results in rearranging the pattern of organization to a convergent extension in which the cells are pulled inward and elongated. Here the process occurs through frizzled-disheveled polarity-signalling pathway. This results in a cooperative behaviour in which cells that are aligned exert forces on their neighbours to align them. The animation can provide an excellent approach to understand cell movements, Bray [16], Lee et al. [59], Strossel [96].

In many modelling situations involving particles (for example, when “Help yourself and suppress thy neighbour”(competitive policy), or attraction and repulsion dependencies are used. These dependencies can be modelled using a combination of Enable (Activation) and Inhibit dependence rules.

Also percolation-like situation (cooperative transition- see Section 4 in which the sharpness of response increases (decreases) with an increase in the number of effector (inhibitor) molecules that must bind to a target molecule can be modelled, Alberts et al [1]. In interactive animation of biological molecules static illustrations are not satisfactory. Transactional agents can enable us to animate the binding of certain molecules, accompanied by other actions. In particular, cell signalling pathways that involve inhibitory, binding, and activation steps can be animated without the inherent ambiguities arising in the order , timing and the interpretation of the events.

3.6 Pattern formation Mechanisms

There are two fundamental mechanisms which utilise all the development parameters in one form or another to produce shapes and patterns in developing organism. Biochemical instability gives rise to patterns of concentration of one or more chemicals in the system. The pattern specification specifies the way in which cells receive information to differentiate. It takes the form of an interaction between external signals and the inherent genetic information in the cells. Morphogenesis (a word derived from shape and generation), on the other hand, is the process by which cells differentiate using the specified pattern to form new shapes and structures during development. Cellular principles of morphogenesis of even simple organisms can provide some information on this process, Ransom [81]. Agent-based simulations can be useful in these areas.

3.7 Signal Processing among Cells

Signalling between cells, Monk et al. [68], is an essential process in the development of multi-cellular systems. A variety of signals are used depending upon the production of the ligand (the mediator of the signal) by signalling cells and detection of this ligand by specific receptors expressed by the receiving cells.

Three kind of signalling are used:

- (i) Autocrine Signalling: A Cell signalling to itself
- (ii) Paracrine Signalling: Signalling between distinct cells that are spatially separated and depends on secreted diffusive ligands such as growth factors and hormones.
- (iii) Juxtacrine signalling: Signalling that is mediated by ligands and receptors that are both anchored to the cell membranes. This signal is a direct signalling between cells and play an important role in spatial pattern generation and wound healing process (Monk et al [68])

The main question is how a nearest neighbour signalling mechanism can account for the observed increase in cell proliferation many cell diameters away from the wound edge. Monk et al [68] shows that Juxtacrine signalling with a positive feedback can create a travel front (similar to an avalanche or percolation) above a critical threshold, thereby establishing long-range connection among cells.

Agent-based model is well-suited to study inter-cellular dynamics on patterning and juxtacrine signalling.

4 Role of Fractal and Percolation 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, Bunde and Havlin [19] Grimmer [39], Hilborn [42], Moon [69], Torquato [102], Wolfram [109]. 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 features and conversely, geometrical feature modify the interaction; see Chapter on “Fractal Analysis” by Avnir et al. in Bunde and Havlin [19], Torquato [102].

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, Maini and Othmar [61], cellular signal propagation, Monk et al [68], Weiss [107], Animal- human trails, Schweitzer[84], and Smart materials, Smith [90]. 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 and pragmatically broken into three distinct cytoplasmic systems: microfilaments, microtubules and intermediate filaments. Cytoskeleton enables cell to swim, crawl and 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.

However, we still need yet another 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. This model is the Percolation model, Grimmett [39], Torquato [102]. The percolation theory is concerned with the study of the diffusion / penetration of certain materials from an environment into the material of system placed in that environment through the boundary or surface of the system. It has been found experimentally that such a penetration is determined by a single parameter p . We call a particular value $p = p(c)$ the "threshold" for percolation, if the pathways are infinite when $p > p(c)$ and the pathways are nonexistent or have limited access within the material of the system when $p < p(c)$. The value $p(c)$ is called the critical point. Also we say that there is a percolation above $p(c)$. The region above $p(c)$ is called the supercritical phase and the region below $p(c)$ is called the subcritical phase. Also when $p = p(c)$ the infinite open cluster or pathway is called an incipient infinite cluster.

Percolation theory is studied using an internal connected structure of the material of the system, either as a lattice or as a connected graph. Also it is studied as a "bond percolation" in which we study the percolation through the edges of the lattice or a graph; or as a "site percolation" in which we study the percolation through the lattice points or vertices of the graph or as a "mixed percolation" involving edges and vertices.

For simplicity, consider a square lattice, where each site is occupied randomly with probability p or empty with probability $(1-p)$. At low concentration p the occupied states are either isolated or forms small clusters. Two occupied sites belong to the same cluster if a path of nearest neighbour occupied sites connects them. When p is increased, the average size of clusters increases. At a critical concentration $p(c)$ a large cluster appears which connects the opposite edges of the lattice. This cluster is called the infinite cluster since it diverges when the size of the lattice is increased to infinity. When p increases further the density of the cluster increases, since more and more

sites become part of the infinite cluster, and the average size of the finite clusters decreases.

The value of the percolation threshold $p(c)$ depends explicitly on the type of lattice and also whether it is site or bond percolation. In site percolation $p(c) = 0.593$ for square lattice and 0.70 for hexagonal lattice. This means, if a square of unit area is filled in with agents whose range of influence is in a square lattice, then it is adequate that about 60% of the lattice is filled so that there will be a percolation path between the edges of the square. The fractal dimension $d(f)$ of the percolation of the square lattice of dimension 2 is $d(f) = 91/48 = 1.9$.

The average density of the agents $\langle n \rangle = \text{Number of agents} / \text{lattice size}$ versus Clustering obtained through simulation is given in Figure 2. Here we see that below the critical density the connectivity is zero due to lack of enough agents in the lattice. But above the critical value the connectivity turns out to be one. Note also that the connectivity is robust above the critical density since the failure of some of the agents can maintain the connectivity. In this sense, they are self-organizing. This property is useful in the cost-effective design of special purpose sensors. We will again deal with this aspect in dealing with resilient networks in Section 6.

In this simulation, we start with a square lattice of side length L and some number p , $0 \leq p \leq 1$; we generate a random number in $[0,1]$ for each lattice cell and say a cell is empty if the random number is less than p . The patterns can be generated for various values of p .

Denoting by $M(L)$ the number of cells in the largest cluster, $M(L)$ scales as:

(i) $\ln(L)$ for $p < p(c)$;

(ii) L^d for $p = p(c)$ where $d = 1.9$ and power-law scaling holds at the critical threshold,

and (iii) L^2 for $p > p(c)$

Random fractals can be highly multiply connected and the topological structure can change dramatically, when a continuously varying parameter increases through a critical value resulting in a simply connected path. Thus self-organized unforeseen new paths can emerge between points of interest beyond a critical threshold, e.g., nest to the food when communication takes place among agents in an environment. In pattern formation theory based on diffusion, such percolation paths that arise above a critical threshold, are also called “Travelling fronts or waves” or “avalanches”, Kauffman [48], Klusch et al [52].

Also percolation-like cooperative threshold phenomenon arises in cell biology; for example, the sharpness of response of cells increases (decreases) with an increase in the concentration (number) of effector (inhibitor) molecules binding to a target molecule or extra-cellular signalling, Alberts et al.[1].

Figure 2. Percolation Threshold

Thus the fractal and percolation models play important roles in understanding a Nature-Inspired Smart System. However, the dynamics of the resulting probabilistic system is in general, highly nonlinear implying that it is not possible to obtain analytical expressions and need to resort to Monte Carlo simulation based methods to compute the percolation threshold and other parameters.

5 Injecting Positive Entropy

The properties of NISS mentioned in Section 1 and the requirement mentioned in Section 4 having a readily modifiable connectivity pattern among its components with a fractal structure, belong to those of both the computational and the nonlinear dynamical systems. The fractal communication structure is very sensitive to small parameter changes and can lead to an abrupt change in behaviour, e.g bifurcation. Fractal dimension, metric entropy, and Lyapunov exponent serve as important invariants to quantify the behaviour of such a non-linear system, Falcioni et al.[30]. A system with negative Lyapunov exponents imply stability and a system with the positive Lyapunov exponents imply instability and the presence of positive feedback. Thus positive feedback (cooperativity) in a nonlinear system implies a positive metric entropy. This is analogous to positive thermodynamical entropy export to the environment in a living system leading to far-from equilibrium states, Prigogine [80], and to self-replication of molecules. Equilibrium Systems have zero entropy and they

do not export positive entropy and correspond to nonliving system. In fact, evolution stops if entropy is zero, Brooks and Wiley [17].

Also, a chemical reaction cannot happen spontaneously unless the entropy turns positive. For example, the conversion of ADP (Adenosine Di-Phosphate) to ATP (Adenosine Tri-Phosphate) is not spontaneous. So the conversion requires another reaction to provide energy. This is Glycolysis-conversion of Glucose to the Lactate anion. This is a spontaneous reaction exporting positive entropy, Petrucci et al [77]. The large positive entropy keeps the reaction far away from equilibrium. As far as biological systems are concerned, reactions reaching equilibrium will result only in dead cells.

Thus the introduction of positive entropy or randomness seems essential in any computational and dynamical system to model the far-from equilibrium systems. Mathematically speaking, such a positive entropic system turns out to be topologically transitive and sensitive to initial conditions, or chaotic. This means, that the system is no longer expressible as a sum of its component parts and the system functions as a whole leading to what is known as "Emergence". In such a case, the algorithmic independence among the subsystems is lost, Chaitin [22].

In general, the evolutionary rules of systems with zero and negative entropy are predictable (or Lyapunov exponent less than or equal to zero) and such rules are non-algorithmic, or not exactly predictable for positive entropy machines. Well-structured objects - such as: Context free grammars, regular grammars and serial-parallel graphs, regular graphs, lattice graphs that are generated using deterministic algorithms provide for easy description through functional rules, and hence have zero metric entropy. Such systems with non-positive metric entropy are "Turing or algorithmically expressible" with a finite set of evolutionary rules. Systems with a positive metric entropy represent "disordered" machines and their evolutionary rules are in general, "Turing non-expressible". The metric entropy can distinguish the two major classes of machines: ordered (O) and disordered (D) machines.

1. Ordered or Zero Metric Entropy Machines (O)

These are completely structured, Deterministic, Exact behaviour (or Algorithmic) Machines. This class contains, the machines in Chomskian hierarchy:

- (i) Finite state machines (regular grammar),
- (ii) Push-down stack machine (context-free grammar),
- (iii) Linear bounded automata (context sensitive grammar)
- (iv) Turing machines that halt
- (v) Exactly integrable Hamiltonian flow machines.

These machines are in principle, information loss-less and instruction obeying; their outputs contain all the required information, as dictated by the programs.

2. Positive Metric Entropy or Disordered Machines (D)

Nonintegrable positive entropy machines exhibit various degrees of irregular dynamics:

(i) *Ergodicity*: 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.

The term “Ergodicity” means statistical homogeneity; here, the trajectory starting from any initial state can access all other states in the phase space.

(ii) *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).

(iii) *Bernoullicity, 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 we define classical chaos. These lie in the border and beyond the Turing computable region; that is they belong to partial recursive schemes leading to undecidability.

(iv) *Far-from--equilibrium systems*: Systems exhibiting emergent behaviour - such as Chemical and Biological machines and living systems, Prigogine [80], Forbes [31], Horgan [43], Marcus [63], Zak [112].

Each of the above properties imply that all the preceding properties hold; that is, within a chaotic region the trajectories are ergodic on the attractor and wander around the desired periodic orbit.

The properties of an NISS are governed by three important parameters: Metric Entropy, Lyapunov exponents, and Fractal dimension. There is a close relationship between positive Lyapunov exponents (or equivalently the eigenvalues of the interaction matrix) and fractal dimensions.

If $h(1), h(2), \dots, h(j)$ are Lyapunov exponents in ascending order that are greater than zero, and $h(j+1), \dots, h(k)$ are less than zero in descending order, then we have j stretching and $(k-j)$ contracting directions in the k dimensional trajectories, then the quantity, Lyapunov dimension, Moon [69], Baker and Blackburn [5] :

$$d(L) = j + 1/h(j+1) \sum_{i=1}^j h(i).$$
 It is conjectured that $d(L) = d(b)$ the box-counting dimension.

As a result $d(L)$ can change rapidly depending upon $h(i)$ as a function of the network size, the connectivity structure and the interaction among the agents (e.g. ant and bacterial colony, Sections 7.1 and 7.2).

Hence, to understand an NISS quantitatively, we need to know the spatial structure of attractors and the temporal aspects of the trajectories. The former provides information about phase transitions, while the latter tells us whether a trajectory governed by a positive Lyapunov exponent falls in a given attractor. Real systems including biological systems are nonstationary in which the parameters are changing with time. Thus quantitative understanding of a NISS is extremely difficult except through simulation. To simulate NISS, we need to combine zero and positive entropy machines; that is we introduce positive entropy through the chaotic (deterministic randomness) or stochastic (statistical randomness) means, in appropriate decision or control steps of a deterministic machine.

The positive entropy endows NISS with several powerful features:

- (i) It provides ergodicity of search orbits. This means every point in the set of accessible states is approached arbitrarily closely in its dynamical regime. This property ensures that searching is done through all possible states of the solution space since there is a finite probability that an individual can reach any point in problem space with one jump.
- (ii) It provides solution discovery capabilities (as in genetic programming-GP) due to embedded randomness. This property arises due to the fact that chaotic orbits are dense and has positive Lyapunov exponent and two initially close orbits can separate exponentially from each other with a possibility of falling into a desired attractor.
- (iii) It provides a common linking thread among several disciplines; e.g., Genetics (genetic-algorithms), Diffusion, Thermodynamics (simulated-annealing), Statistical Mechanics (Particle transport); Complex Systems (Active -walker, Self- organization and percolation models).

6 Graph Connectivity Structure arising among Agents

In the last section we discussed the role of randomness or chaoticity (for introducing positive entropy) in computational systems. In this section we will discuss the role of disorder or randomness in interconnection network among interacting agents. These disorderly interconnection patterns play a key role for applications to various modeling and simulation aspects of NISS, Strauss [94]. 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 three typical disordered networks such as:

(i) Random networks (ii) Power-law scaling networks, and (iii) Small World Networks. from regular networks. For an excellent survey of complex networks, Newman [73].

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 i , $0 \leq k(i) \leq (n-1)$

(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:

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

This is referred to as “ global connectivity” in the context of lattice percolation (Fig.2), 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 $0 \leq C \leq 1$

6.1 Random, Scale-free and Small World Networks

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 the social networks have a range $2 < g < 3$, Chung and Lu [24], Barabasi et al [6].

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.[27] 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 [104], 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.

6.2 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 distinguished by using a degree-correlation coefficient that is positive for assortative networks and negative 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 [73] 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.

6.3 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 the 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 [48].

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 addition, 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, Gell-Mann [34]; 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.

6.4 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 [73]. The degree distribution satisfies $P(k) = k^{-g}$ with $g = 2.4$ showing that they are scale-free, self-organizing and resilient against damage. Similar results have been reported by Atilgan et al [2]; namely, 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, Boahen [12], Jacobson [47] :

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 through interactions among far-flung components in a way that depends upon the electrical signals that pass between them.

The central challenge of neural development is to explain how the axons and dendrites grow out find their right partners and synapse with them selectively to create a functional network. This growth is very sensitive to diffusible chemotactic factors (as in ant swarms). The different pathways grow out as in the movement of cars in a rush-hour traffic in highway junctions. At the same time the component failures are tolerated. These indicate that the scale-free and small world networks play a role in the design of nervous system and Hebbian networks; see Szirtes et al [100].

7 Nature -Inspired Smart Computations

We now describe a multiset of agents based self-organizing heuristic that uses inferences, beliefs and computation. We assume that the number of agents is very large to exhibit the statistical features. The agents use randomization in the application of the rules and interconnectivity among them that can alter the important neighbourhood properties; this results in different kinds of disordered networks with fractal dimensions.

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, Kennedy [51] 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.

(v) In the agent model, shared landscape can be simulated through a blackboard and self-avoiding walks can be simulated using a table that stores the locations visited earlier.

(vi) Also MAP can handle Parallel ant colony system simulation suggested by Chu et al. [23], Blackwell and Branke[11]. If many different equally likely independent modes m are to be searched for, it is more efficient to choose a multiset of agents with m sets, each with k agents so that $mk = N$ where N is the total number of agents; i.e., the agents are partitioned into distinct groups each with their own speciality.

In order that a multiset of agents exhibit the four properties of NISS mentioned earlier, we require that the agents are autonomous (not centrally controlled), and can either enable (link) or inhibit (de-link) other agents on their own, based on the information available from their environment to reach a specified goal (proactive). 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 interactive networks, exhibiting the properties of small world graphs or scale-free property or their combined properties.

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. From a practical viewpoint, in the agent model, the shared landscape can be simulated through a blackboard and self-avoiding walks can be simulated using a table that stores the locations visited earlier.

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 agents know, 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, de-link and form clusters. This would result in nonlinear iterative schemes among the agents. 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).

The above two rules permit us to model Markovian random walks which is independent of the past history of the walk and non-Markovian random walks, dependent upon past history- such as self-avoiding, self-repelling, communicating, and active random-walker models. This can result in various kinds of attractors having fractal dimensions presenting a swarm-like, flock-like, bacterial colony-like appearances depending upon the Jacobian of the mapping, Ilachinski [46], Wolfram [109], Ben- Jacob et al [8].

7.1 Multiset of Agents Based Ant Colony Heuristic

The Ant Heuristics, Dorigo et al [26], Chu et al., [23], Blackwell and Branks, [11], 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 pheromone 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 resulting swarm (ant-network) can exhibit features of both the small world graphs and scale-free graphs and can change from one class to another.

The formalized agent-based heuristic for a shortest path is given below:

Step 1:

Generate a multiset of agents each agent representing an ant. Let there be G sets contained in the multiset, each set containing identical agents $N(j)$ for $j = 0, 1, 2, \dots, (G-1)$.

Randomly select an initial node r for each agent. The initial pheromone level on every edge of the graph is set to a small positive quantity $f(0)$ in the blackboard. Set the cycle counter to zero.

Step 2: Initialization of beliefs:

Let $J(i,j,r)$ is the set of cities that remain to be visited by the i th agent in the j th group starting from node r ; let $f(j,r,s)$ is the pheromone level between node r and node s , for the j th group.

Let $v(r,s) = 1/D(r,s)$ where $D(r,s)$ is the distance between the nodes r and s ; let α be a parameter which determines the relative degradation of pheromone level inversely with the distance.

Stepping Rule: Starting from r , calculate the next visited node s , for the i th agent in the j th group as per the rule:

(i) To begin with choose q , a random number in $0 < q < 1$; let $q(0)$ be a constant chosen in the interval between 0 and 1.

(ii) **If** $q < q(0)$, **then**

$$s = \text{Arg Max } u \in J(i,j,r) [f(j,r,u)]^\alpha \cdot [v(r,u)]^\alpha,$$

else $P(i,j,r,s)$.

Here $P(i,j,r,s)$ is the transition probability from node r to node s for the i th agent in the j th group, given by:

$$\text{If } s \in J(i,j,r), \text{ then } f(j,r,s) \cdot (v(r,u))^\alpha / \sum_{\{u \in J(k,r)\}} [f(j,r,u)]^\alpha \cdot [v(r,u)]^\alpha,$$

else 0;

Updating the beliefs from Intra-group knowledge:

Step 3: Local Landscaping Rule:

Update the pheromone level between nodes for each group using:

$$f(j,r,s) = (1-\gamma) \cdot f(j,r,s) + \gamma \Delta f(r,s) \text{ where } \Delta f(r,s) = f(0) = 1/[nL(n,n)]$$

where $f(j,r,s)$ is the pheromone level between nodes r and s for the agents in the j th group. and $L(n,n)$ is the approximate distance of the route between all nodes. Here n is the number of nodes and $0 < \gamma < 1$ is the decay parameter of pheromone.

Continue Stepping and local landscaping until each agent in each group completes the route. In this manner, each agent updates its beliefs and communicates the beliefs to concerned agents.

Step 4: Evaluation:

Calculate the total length travelled by each agent in each group.

Step 5: Global Landscaping Rule:

Let $L(j)$ is the shortest length for the agents in the j th group and α is pheromone parameter.

Update the pheromone level between the nodes for each group using the information on the best available route of the j th group thus:

$$f(j,r,s) = (1-z).f(j,r,s) + z. \Delta f(j, r,s)$$

where we set

$\Delta f(j, r,s) = \text{If } (r,s) \text{ belongs to the best route of the } j^{\text{th}} \text{ group, then } (1/L(j))$
else ZERO.

Step 6 : Updating belief from Inter-group Knowledge:

Let $L(\text{best})$ be the best length among the best route of all the groups, i.e., $L(\text{best}) < L(j)$, for all j , $j = 0,1,2,...,(G-1)$

Reassign

$$f(j,r,s) = f(j,r,s) + w. \Delta f(\text{best}, r,s)$$

where : $\Delta f(\text{best}, r,s) = \text{if } (r,s) \text{ belongs to the best route of all groups then } 1/L(\text{best}),$
else ZERO.

Step 7: Stability:

Perform all steps 2 to 6 **until the system reaches stability**, when **almost all** the agents travel the same route.

Simulation Results

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*”.

Also it turns out that the many different equally likely independent modes m are to be searched for, and there are N agents, it seems to be more efficient to choose a multiset each with k elements such that $mk = N$ or $k = N/m$. If there are correlations between different modes, it will be helpful to use this knowledge with advantage. Such a correlation leads to selective linking or assortative or homophilic mixing.

It has been pointed out by Newman [73] that such an assortative mixing is more prevalent among social networks, while disassortative networks seem to be common in biological networks. It appears that 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.

Since the scale-free networks and small world networks exhibit self similarity (in the sense their highly connected components themselves belong respectively to the same class), the hierarchical decomposition arising due to multiswarming results in each group linking to another group through a short path length, Song et al [91].

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.

Also (i) if some agents are assumed to be aging, non-cooperative and they are allowed to die, and (ii) we limit the number of links to each agent by a cost factor, then the scale-free distribution is markedly affected. Our estimate of the box-counting fractal dimension of the swarm lies between 1.7 and 1.9 (resembling Diffusion limited aggregation, Grimmer [39], Torquato [102]). This value seems to be lower than the expected value 2 of the power law exponent for metabolic networks but consistent with the expected value the power law exponent for biological networks, Newman [73]. It may be due to the approximate procedure used to compute the fractal dimension and the relatively small number of agents used in the estimation.

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 colony. To model such systems Chung and Lu [24] suggest methods that involve both geometry and topology as well as a combination of preferential attachment and duplication of nodes and edges.

For practical applications the above heuristic can be improved in several ways. 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.

7.2 Smart Bacterial Colony

Microorganisms often need to cope with hostile environment. To do this they have developed sophisticated cooperative behaviour and intricate communication capabilities- direct cell-to cell physical interactions via extra membrane polymers, collective production of extracellular wetting fluid for movement on hard surfaces, long range chemical signalling, such as quorum sensing and chemotactic (bias of movement according to chemical gradient) signalling, collective activation, deactivation of genes and even exchange of genetic material. Utilising these capabilities the bacterial colonies develop complex spatio-temporal patterns in response to adverse growth conditions (Ben Jacob et al [8])

In particular, a variety of branching and chiral (handedness) fractal patterns emerge . Invoking the ideas from pattern formation in nonliving systems and using generic modelling, the studies of Ben Jacob et al [8] reveal novel survival strategies which account for the evolved patterns. Using the communicating walkers model (Discrete model) in which bacteria are represented by discrete, random walking entities consuming nutrients, reproduce , perform random biased movement, and produce and respond to chemicals whose time evolution is described by reaction-diffusion equations they show how communication leads to self-organization via a cooperative behaviour of the cells. In this regard, pattern formation in microorganisms can be viewed as the result of the exchange of information between the microlevel (individual cells) and the macro-level (colony).

For colonial adaptive self-organization the bacteria employ three kinds of chemotactic responses, each dominant in a different regime of the morphology:

1. Food chemotaxis for nutrient intake
2. Repelling chemotactic signalling, a long-range signalling, by starved bacteria at the inner parts of the colony.
3. A short range attractive signal

The attractive and repulsive signalling is called two-component signalling and is widely used in animal and plant kingdom; it is determined by the diffusion constant of the chemical agent and the rate of spontaneous decomposition. If nutrient is deficient for a long time, the bacterial cells may enter a state of spore which enables them to survive much longer without food. Since these spores have no metabolism the bacteria entering into the spore state emit a wide range of waste materials acting as a chemical repellent so that other bacteria can escape the dangerous situations.

Bacterial colonies alter their fractal patterns so as to have different dimensions depending upon the growth conditions. The communicating walkers model reveals that in the fractal bacterial colony, the cells are active at the outer parts, and closer to centre they are stationary and some of them form spores. It is known that some bacteria respond to adverse growth conditions by entering a spore stage until more favourable conditions return. Such spores are metabolically inert and exhibit a marked resistance to the lethal effects of heat, drying, freezing, deleterious chemicals and radiation.

We can learn a lot from the smart bacterial colony as to how to build agent-controlled nano, micro and milli robots for special purpose missions.

7.3 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 Schweitzer [84], Ebeling and Schweitzer [28]. Similar approach can be useful for the study of traffic patterns and crowd dynamics and the way to control them.

7.4 Simulating Attractor Dynamics using Exploration

For many fractal problems, multi agent simulation can speedup the procedure of understanding attractor dynamics by plotting the landscape. We give an example of attractors obtained in the Newton method for solving the complex polynomial $f(z)$ is defined by $z = N(z)$, where

$N(z) = z - f(z)/f'(z)$. Let us consider the computation of the cube root of unity, i.e., solving the equation $f(z) = z^3 - 1$. Then $N(z) = (2z^3 + 1)/3z^2$.

This has attractors (fixed points) at 1 , $\omega = \exp 2\pi i/3$, and $\omega^2 = \exp 4\pi i/3$. The attractor landscape turns out to be very complex for a polynomial with many roots.

Multiple agents Simulation Procedure: Select a grid of points covering the square region whose corners are at $2 + 2i$, $2 - 2i$, $-2 - 2i$, and $-2 + 2i$. Here we employ four agents. Each agent calculates the value of the 100th iterate of a randomly chosen point and perform the iteration :

$z = N(z)$ and colour the points as follows.

- (i) If the distance from the 100 th iterate to 1 is less than $1/4$, then it is assumed that the point is in the stable set of 1 and the agent colours it blue (in black and white, light gray).
- (ii) If the distance from the 100 th iterate to $\exp 2\pi i/3$ is less than $1/4$, then it is assumed that the point is in the stable set of $\exp 2\pi i/3$ and the agent colours it green (in black and white, dark gray).
- (iii) Finally, if the distance from the 100 th iterate to $\exp 4\pi i/3$ is less than $1/4$, then we assume the point is in the stable set of $\exp 4\pi i/3$ and the agent colours it red (in black and white, black)..
- (iv) All the points which are not within $1/4$ of one of the roots after 100 iterations of N are left white.

For a 251 by 251 grid the graph produced by the above process is shown in Fig. 3. Observe that we have three regions — one region defining the stable set of 1 , the second one containing the stable set of ω while the third one consisting of the stable set of ω^2 .

Figure. 3. Attractor Sets of Newton iteration

7.4.1 Bistability in Perception

In visual perception, as for example, in the Necker cube illusion, the attention is switched between two images. This switching is chaotic and corresponds to two different chaotic attractors, Hoppensteadt and Izhikevich [45]. Also we can simulate multistable perception arising due to deterministic chaos, in which many ambiguous interpretation of the image alternate spontaneously, Nishimura et al. [74].

7.5 Oscillatory Chemical Reactions

We mentioned in Section 2 that the rule processing does not terminate if and only if rules provide new conditions to fire indefinitely - that is actions of a rule R_i create the right conditions for another rule R_j to fire. Such a cyclic system occurs commonly in biology and seems to form the basis of all living systems. For example, the energy-rich molecule adenosine triphosphate (ATP) is produced through a succession of reactions

in the glycolytic (sugar-splitting) cycle that involves ATP at the start. To produce ATP we need ATP. These reactions correspond to catalytic reactions that arise in nonequilibrium systems where a set of reactants produce a set of products that react with some of the reactants and continues indefinitely as an oscillatory reaction. These reactions, for example, can be of the form in which one of the initial reactant A and another reactant X produces Y, and Y produces Z and Z produces X. This is an interacting and competing three-rule system. Such reactions have rates that are determined by the concentration of the reactants and the products. Thus depending upon the concentration and the frequency or probability of application of the rules the system can lead to convergence, divergence or oscillations. Such oscillatory systems are nonlinear and are called non-equilibrium systems or Brusellator Model, Suzuki et al [97], and can be simulated using MAP.

The ATP synthesis is a reversible coupling device that can convert energy of the electrochemical proton gradient into chemical bond energy or vice-versa. The former takes place through hydrolysis and the latter results in synthesis.

$ATP \rightarrow ADP + P \rightarrow ATP$.

Normally for each molecule of ATP split to ADP and phosphate ,three molecules of sodium are pumped out and two molecules of potassium are pumped in. Such reaction rules can be incorporated in a specified agent for simulation or reactive animation.

Also other reversible reactions such as; first order of the form:

(Reactants) $R \rightarrow P$ (Products) ; second order of the form $S + E \rightarrow ES$ where S is bound to E resulting in the bound complex ES along with kinetic equations ,reaction rates, and equilibrium conditions.

Also inhibitory reaction mechanisms such as: A can combine with B ,only when B is not bound to C can be modelled.

The brusselator model consists of the four rewriting rules::

(1) $A \rightarrow X$; (2) $BX \rightarrow YD$; (3) $XXY \rightarrow XXX$; (4) $X \rightarrow E$

Thus the multi-agent system is a multiset $M = \{A, B, X, Y, E\}$ and a set of reaction rules R defined above. among the agents and within. Note that an agent can undergo self- mutation on its own (autolysis), or due to interaction with the environment; e.g., in the rules for $A \rightarrow X$ or $X \rightarrow E$. We can also incorporate the rate of reaction within each rule that can be either deterministic or probabilistic. By varying the reaction rates or rates of application of rules the system the number agents X and Y diverge, become unstable, oscillate or converge.

For example, a simple simulation shows that the relative frequencies of application of rule (1) and rule (2) can produce varying kinds of effects in the population (Numbers) of X and Y.

7.6 Graph Growth Models

The problem of growing a graph until it reaches self-organized criticality through interaction is closely related to Genetic Programming (GP), Koza [53]. 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, Newman [73], Strauss [94] 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).

As mentioned earlier in Section 7.1 other probabilistic models can be used for graph generation that involve both geometry and topology, as well as a combination of preferential attachment and duplication of nodes and edges, Chung and Lu [24].

7.7 Stigmergy and Games

In Stigmergy (a term coined by French biologist Pierre-Paul Grasse), two or more organisms collectively and indirectly interact through the environment. This has all the features of a multi-player game - in which players have common knowledge-that other players know that their co-players know so that the game reaches an equilibrium. In this game, one of the players modify the environment, the other responds to the new environment at a later time, e.g., nest building in insect society. Collective construction of a nest is an important activity for survival of these organisms when provided with adequate building material and subject to the following constraints:

1. Economy: Getting a maximum benefit out by spending minimum energy.
2. Global plan for building the nest is unavailable.
3. Environmental structure is not known.
4. Communication with the environment is essentially myopic (localized) .
5. A basic template forms the basis; it is replicated and used in construction.
6. There is a finite number of iterations and the game of construction comes to an end reaching a compact structure keeping the variation to a minimum

The above six constraints are basic to all self-replicating evolutionary mechanisms, including the very basic life-generating RNA molecule, Alberts et al [1].

Starting with a basic template (pattern) the nest grows by adding new material, Werfel et al. [108], Boncheva et al. [15], 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, Koza [53] in the sense it is a mutually

bootstrapping process 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 individual 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, Camazine et al. [20], 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 preferentially ordered and the local selection principle and attachment ensures local fitness for each partner consistent with the local fitness for all the other partners., and promises global (landscape) fitness resulting in a scale invariant property for the shape. This whole process of nest building resembles a strategic game with a limited local information. The local fitness ensures global fitness because of the geometrical scale invariance. It is amazing that wasps use the preferential attachment rule that results in self-similarity. However, here we are dealing with Euclidean geometry in building compact geometric shapes that occupy the largest possible area (volume) with the least perimeter (surface area).

The wasp nest building is related to achieving a Nash equilibrium in strategic games, Kauffman [48], Klusch and Gerber [52], Fudenberg and Tirole [32] Osborne and Rubinstein [75], Weirich [106]. In fact, Darwin observes in his “The Origin of Species” that “ the work of construction seems to be a sort of balance struck between many bees..”. By equilibrium, we mean that, there is a profile of strategies that are self-supporting, assuming that the profile is realisable. A Nash equilibrium is a profile of strategies such that each player’s strategy is an optimal response to other players’ strategies. Nash equilibrium arises out of a local combination of myopic actions among agents, where each agent chooses its own strategy, as long as other agents do not deviate from their own fixed strategy. Here there is a large group of players who are matched at random to build a specific nest. The players are not allowed to communicate directly and each player chooses her own 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. This scheme is widely prevalent in natural computing, as we saw in the case of bacterial colonies. This example emphasises the fact that, Game theory will play increasing roles in agent-based models for natural computing.

Further, in wasp nests, the walls are corrugated to achieve self-similarity of the whole structure, since hexagons cannot tile the plane forming another hexagonal shape again! 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). Self-similarity makes nest building robust with respect to cross-over and mutation because it favours positive correlations between the form and quality of the possible solutions.

Here we use the Quality index (QI) derived from geometric compactness, as the fitness metric for self-similarity. Thus a partial structure is self-similar to the total structure if their fitness match approximately. It appears that all the evolutionary processes self-organize themselves into a self-similar state (w.r.t to some suitably defined metric) as we observed earlier in the context of ant swarming. Self-similarity reduces the probability of being disrupted when using building blocks. Building block disruption occurs when a cross-over operation destroys useful schemata. The QI provides proportionality representation for the self-similar structures and forms the basis for self-organization, Garibay et al [33].

The nest construction rules used by wasps, Camazine et al. [20], can be modeled through Multiset of agents- based Genetic programming (GP). The wasps 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, Sankar and Krishnamurthy [83]. In self-organization the nests are evaluated 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 $4 \pi A/P^2$, 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). This index provides a similarity measure or self-similarity metric for the pattern generated. Here a pattern is an assemblage of regular hexagons. This can be defined for all connected pattern of hexagons.

Self similarity holds when for all connected sub-patterns the quality Index is approximately equal to that of a full pattern; that is average fitness=entire fitness. The QI permits the evaluation of the raw fitness of each worker's construction and the shared fitness.

For a 3- dimensional object the $IQ = 6 V \pi^{1/2} / S^{3/2}$ 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, Camazine et al [20]).

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. Each wasp chooses an action (either consciously or based inherently in their genetic make-up) that is best possible in that context for their survival, from a set of possible actions. Such actions ultimately reach one or more layers of nest resulting in equilibrium states (depending upon the requirement to accommodate the entire colony).

In the following wasp-nest -like growth 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 rule- variants are omitted. 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 rank orders the rules 3a, 2b and 2a. We will show that $QI = 48$ (**64**) is a global convergent value or an equilibrium state for a large nest with an odd (**even**) sequence of hexagons with the specified symmetry. In this growing nest, the terminals are hexagons; the functions are composition of hexagons, the control parameters are probabilistic and fitness is evaluated by QI.

7.7.1 Wasp Nest-like Growth

Step 1: Create the template hexagon ; its fitness $QI = 36$

Step 2: For every hexagon created, catalyze the creation of new walls with the following probabilistic preferential ranking rule that locally ensures fitness.

Rank 1: Add 3 walls (for existing 3 walls) with a probability 0.55.

Rank 2: Add 5 walls (for existing 1 wall) with a probability 0.39.

Rank 3: Add 4 walls (for existing 2 walls) with a probability 0.06.

Step 3: Repeat until a desired size is reached; global fitness is ensured.

The preferential ranking is similar to generating a scale-free graph, except that we are in the domain of geometry and generating a scale invariant shape through growth and preferential attachment, rather than in the domain of topology.

Rule 2b: (10,2),QI=50
Add 5 walls; Probability =0.39

(14,3),QI=65

Rule 3a: (14,3),QI=65
Add 3 Walls; Probability = 0.55

(14,4),QI=49

Rule 3b:(12,3),QI=48
Probability =0.06

(14,4),QI= 49

Global Fitness test for Wasp Nest

We now give examples of two type of layered sequences (odd and even) of hexagons built using the above rules locally, and how the global fitness scales as a power-law.

(i)Odd Sequence of Hexagons:(More-circular; Globally more fit): for layers $k=0,1,2,\dots$

Area q : $1,7,19,37,61,91,127,169,217,\dots, (3k^2+3k+1)$

Perimeter p : $6,18,30,42,54,66,78,90,102,\dots, 6(2k+1)$

$p^2/q = 48 - 12/(3k^2+3k+1)$ or p^2/q is approximately equal to 48 for a large q .

The log p -log q plot is a straight line with a slope $1/2$, indicating that it is scale-free.

(ii)Even Sequence: (More like diamond-Globally Less fit)

Here we have, for layers $k=1,2,\dots$

Area: Hexagons q : 4,16,36,64,100,144,196,256,..., $4k^2$,

Perimeter p : 14,30,46,62,78,94,110,126,... $(16k-2)$,...

$p^2/q = 64 - (16k+15)/(k+1)^2$, or p^2/q is approximately equal to 64 for a large q .

The log p -log q plot is a straight line with a slope $1/2$, indicating that it is scale-free.

Note that both the types of nests shown above belong to the same universal class with the same power-law exponent $1/2$.

7.7.2 Cell diversification through Inductive Interaction

While stigmergy is a cooperative development, in cell biology a competitive development leads to specialized cells with a change in character. This is called **inductive interaction**. Here, adjacent cells that were similar to begin with, compete so as to adopt a special character. Any cell that gains the advantage in this competition strongly signals and inhibits its neighbours from sending similar signals to others in return. This is called lateral inhibition. It leads to the creation of a specialized cell surrounded by inhibited cells that can adopt to a new character, Alberts et al [1].

7.8 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, Pikovsky et al. [78], Manrubia et al. [62], Strogatz [95].

The synchronization of two pendulums from unsynchronized to total synchronized state beyond a critical threshold of the coupling strength. Using a simple two-state model, Baker and Blackburn [5] show that as the coupling strength increase beyond the threshold value 0.6 the two pendulums can be synchronized. This graph is similar to the percolation model studied in Fig.2.

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 systems can function like Chemical systems, Biological systems, Molecular machines and will have applications in designing task specific biomorphic robotics, Bar Cohen et al.[7], Boahen [12], Manrubia et al. [62].

8 Engineering NISS

Since an NISS 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 NISS 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 microstructures, Torquato [102]. Emergence is a global behaviour (or a goal) that evolves from the local behaviours (goals) of components, Serugendo et al. [85], [86], [87]. 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 NISS with exactly specified properties.

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 NISS. Recent advances in Cell biology reveals that Nature has always concentrated on devising special purpose computing elements having both the analog and digital features, and exploiting cooperative and competitive schemes using percolation like threshold phenomena, positive feedback leading to self-enforcement, nonlinearity, and short and long range signalling to facilitate interaction using disordered protein networks. For accomplishing any special task, the protein machines are interconnected through softwiring, and are temporally and spatially coordinated through linked processes to achieve maximal efficiency.

8.1 Quantum Mechanics and Smart System

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 because:

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 Lyapunov 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. The complexity of chaotic dynamics manifests itself in that the evolved pattern lives in space that is far larger than the original space of coarse grained cells.

For example if we use floating point computation, we need to extend our precision indefinitely to capture the chaotic evolution, since the number of bits required grows exponentially with the positive Lyapunov exponent resulting in an enormous algorithmic cost. However, if indeed we can build quantum registers, we can capture this information without extra cost as quantum dynamic evolution opens up the Hilbert space of states. But we need to prevent the destruction due to decoherence ! Accordingly, ill-conditioned systems due to near singularity are in principle solvable by quantum means, assuming that the coherence is not destroyed.

For example , non-invertible maps such as the Arnold cat map:

$$x(t+1) = x(t)+y(t) \mod 1$$

$$y(t+1) = x(t)+2y(t) \mod 1$$

where x,y are real numbers in $[0,1]$ can run forwards and backwards in Quantum computing recovering the initial states, whereas in classical computing we will find the information destroyed.

3. Built-in-ergodicity:

Whereas a classical agent will travel precisely through a specific deterministic path, if there is no added randomness, inherently a quantum mechanical agent explores all possible 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:

Quantum 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 systems already exhibits emergence leading to many different views. Decoherence provides a particular view.

5. Quantum Systems are not easily amenable to simulation

It is claimed that, quantum computations are complexity-free, quantum agents have a built in self-awareness to use exploration and exploitation for solving problems. Also the quantum smart systems seem to remember past forever and never age. However, there seems to be no way to simulate quantum mechanical agents. This is a grand challenge!

9 Multi-agent Architecture and Toolkits

Shakshuki et al. [88], evaluate multiagent tool kits, such as: Java Agent development framework (JADE), Zeus Agent building toolkit and JACK Intelligent Systems. They consider Java support, and performance evaluation. The number of agents they consider is of the order of 32. For the implementation of the paradigm described here, further developments are needed in Agent technology, since we need a very large number of agents to simulate many real-life scientific applications.

Gorton et al [37] have evaluated agent architectures: Adaptive Agent architecture (AAA), Aglets developed by IBM, and the Java based architecture Cougaar. The paradigm described here is well-suited for implementing in Cougaar, a Java based agent architecture, since Cougaar is based on human reasoning. A Cougaar agent consists of a blackboard that facilitates communication and operational modules called plug-in that communicate with one another through the blackboard and contain the logic for the agent's operations. The use of blackboard and direct communication are useful for simulating the problems described in this chapter. Further developments in agent-based languages and architecture are needed to simulate very large scale complex systems.

10 Summary and Directions for Research

We described some important properties a Nature -Inspired Smart System (NISS) need to possess 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 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 macro scale NISS.

Modelling power and decision power work in opposition! Chaotic sequences have greater modelling power (more complex) than Type 0 (Turing Machine) grammar. Working with them is difficult indeed - since the decision power is weak! Also, self organization is at a higher level of hierarchy and requires a higher modeling power than a Turing machine in producing high adaptability and efficiency, rather than programmability! Thus programmability and adaptability seem to be mutually exclusive and whenever maximum efficiency exists adaptive systems can do better.

In looking forward to future, much work needs to be done. 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 complexity of behaviour of natural systems we see around us.

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.

Another challenging problem is to consider the role of scales and the role of quantum computational systems in the design of NISS. This will require understanding and reconciling several fundamental issues of classical nonlinear dynamics and biodynamics with the philosophy of quantum mechanics. The science of NISS is a fascinating area that can keep the upcoming generation of computational biologists active for many decades.

References

- [1] Alberts,B et al.(2002)*The Molecular Biology of the Cell*, Garland Science, New York.
- [2] Atilgan,A.R. et al.(2004) A, *Small -world Communication of Residues and Significance in Protein Dynamics*, Biophysical Journal , Vol. 86(1), 85-91.
- [3]Aviv,R and Shapiro,E.(2002) *Cellular Abstractors: Cellular computation*, Nature, Vol 419, 343.
- [4]Ayers,L et al. (2002) *Neurotechnology for Biomimetical Robots*, M.I.T.Press, Cambridge, Mass.
- [5] Baker,G.L.and Blackburn,J.A.(2005) *The Pendulum*, Oxford University Press, Oxford.
- [6]Barabasi,A and Albert,R(1999) *Emergence of Scaling in Random Networks*, Science, Vol.286,pp.509- 512.
- [7] Bar-Cohen,Y and Breazeal,C.(2003) *Biologically-Inspired Intelligent Robotics*, S.P.I.E. Press, Bellingham, Washington.
- [8]Ben-Jacob,E.et al.(1997)*Smart bacterial colonies ,in Physics of Biological systems: From Molecules to Species*, Lecture Notes in Physics,Vol.480, pp.307-340, Springer Verlag, New York.
- [9] Ben-Naim,E.et al.(Eds) (2004) *Complex Networks*, Lecture Notes in Physics, Vol.650, Springer Verlag, New York.
- [10] Benyus,J.M.(2002) *Biomimicry*,Harper, New York..
- [11]Blackwell,Tand J.Branke,J.(2004) *Multi-swarm Optimization in Dynamic environments*, Lecture Notes in Computer Science,Vol.3005,Springer Verlag, New York, pp.489-500..
- [12] Boahen,K.(2005) *Neuromorphic microchips*, Scientific American, Vol.292, May, pp.38-41
- [13]Boloni,L et al (2004) *Software Engineering Challenges for mutable-agent systems*, Lecture Notes in Computer Science, Vol.2940,pp.149-166, Springer Verlag, New York.
- [14] Bonabeau,E.et al. (1999) *Swarm Intelligence :From Natural to Artificial systems*, Oxford University Press, London, U.K.
- [15] Boncheva, M. et al., (2003) *Millimeter scale self-assembly and its applications*, Pure and Applied Chemistry,Vol.76(5),pp.621-630.
- [16] Bray,D(1996). *Cell movements*, Garland, New York, 1996.
- [17] Brooks,D.R and Wiley, E.O (1988) *Evolution as Entropy*, University of Chicago Press, Chicago, Ill.
- [18]Bullock,S.(2006)*The invention of algorithmic biology*, in Richard Hawkins, pp.116-124, A.Griffen and M.Ridley (Eds).Oxford University Press, Oxford
- [19]Bunde,A and Havlin,S. (1994) *Fractals in Science*, Springer Verlag, New York.
- [20]Camazine,S. (2002) *Self-Organization in Biological Systems*, Princeton University Press, Princeton.
- [21]Cardelli,L. (2005) *Abstract Machines in Systems Biology*, Springer Transactions on Biological Systems, Springer Verlag, New York.
- [22]Chaitin,G (2003) *Two Philosophical Applications of Algorithmic Information Theory*, Lecture Notes in Computer Science, Vol.2731, Springer Verlag, New York, pp.1-10.
- [23]Chu,S et al. (2003) *Parallel Ant colony Systems*, Lecture Notes In Artificial Intelligence,Vol.2871, Springer Verlag, New York.pp.279-284.
- [24]Chung,F and Lu, L.(2006) *Complex Graphs and Networks*, American Mathematical Society, CBMS, Vol.107,Providence, Rhode Island.

- [25] deCastro,L.N. and Timmis, J.I. (2002) *Artificial Immune Systems: A New computational Intelligence Approach*, Springer Verlag, New York.
- [26]Dorigo,M et al.(2002) *Ant Algorithms*, Lecture Notes in Computer Science, Vol.2463, Springer Verlag, New York.
- [27]Dorogovtsev,S.N and Mendes,J.F.F. (2003) *Evolution of Networks*, Oxford University Press, Oxford.
- [28]Ebeling,W and Schweitzer,F (2003) Self-organization, Active Brownian dynamics and biological Applications , *Nova Acta Leopoldina* , Vol.88, No.332,pp.169-188.
- [29] Effroni,S et al. (2005) *Reactive animation: Realistic Modeling of Complex Dynamic Systems*, IEEE Computer, January, pp.33-46.
- [30] Falcioni,M et al. (2003) *Kolmogorov's legacy about entropy, Chaos and Complexity*, Lecture Notes in Physics, Vol.636, Springer Verlag, New York, pp. 85-108.
- [31] Forbes,N.(2004) *Imitation of Life*, M.I.T.Press, Cambridge, Mass.
- [32] Fudenberg, D. and J.Tirole,J. (1991) *Game Theory*, MIT Press, Cambridge, Mass.
- [33] Garibay,I et al..(2006) *Emergence of Genomic self-similarity in location independent representations* , Genetic programming and evolvable Hardware,Vol.7, pp.55-80.
- [34]Gell-Mann,M.(1994) *The Quark and the Jaguar*, W.H.Freeman, New York.
- [35] Goldberg,D.E..(1989) *Genetic algorithms in search, optimisation and machine learning*, Addison Wesley, Reading, Mass.
- [36] Goncharova, L. et al. (2003) *Biomolecular Immunocomputing*, Lecture Notes in Computer Science, Vol.2787, pp. 102-110, Springer Verlag, New York..
- [37] Gorton,I. (2004) *Evaluating agent Architectures: Cougaar, Aglets and AAA*, Lecture Notes in Computer Science, Vol.2940, Springer Verlag, New York,pp.264-274.
- [38] Graham,I and. Duke,T(2005) *The logical repertoire of ligand-binding proteins*, Physical Biology, Vol.2, pp.159-165.
- [39] Grimmett,G.(1999) *Percolation*, Springer, New York,1999.
- [40] Guerin,S and. Kunkle,D. (2004). *Emergence of Constraint in Self-organizing Systems*, Nonlinear dynamics, Psychology and Life Sciences, Vol. 8, No.2, pp.131-146.
- [41]Harel,D.(2003) *A grand challenge for computing: towards full reactive modeling of a multicellular animal*, EATCS Bulletin, [http:// www. wisdom. weizmann. ac. il/ ~dharel / papers/grandchallenge.doc](http://www.wisdom.weizmann.ac.il/~dharel/papers/grandchallenge.doc).
- [42]Hilborn,R.C. (2003)*Chaos and Nonlinear Dynamics*, Oxford University Press, Oxford.
- [43]Horgan,J.(1995) *From complexity to perplexity*, Scientific American, June,pp.74-79.
- [44]Hopgood,A.A.(2005)*The State of Artificial Intelligence*,pp.3-77,Advances in Computers,Vol.65, Elsevier, New York.
- [45]Hoppensteadt, F.C and Izhikevich,E.M. (1997) *Weakly Connected Neural Networks*, Springer, New York.
- [46]Ilachinski,A.(2001). *Cellular Automata*, World Scientific, Singapore.
- [47]Jacobson,M.(1992). *Developmental Neurobiology*, Plenum Press, New York.
- [48] Kauffman,S.A.(1993). *The origins of Order*, Oxford University Press, Oxford.
- [49]Keele,J.W and Wray, J.E. (2005). *Software Agents in molecular computational Biology*,Briefings in Bioinformatics, Vol.6, No.5,December, pp.370-379.
- [50]Kennedy,J and Eberhart,R.C (2001).*Swarm Intelligence*, Morgan Kauffman, London.
- [51]Kennedy,J. (2006). *Swarm Intelligence*, in *Handbook of Nature-Inspired & Innovative Computing*, Ed: A.Zomaya, Springer Verlag, New York, pp.187-221.
- [52]Klusch,M and Gerber,A.(2002), *Dynamic coalition formation among Rational*

- Agents*, IEEE Intelligent Systems, May/Jun pp.42-47..
- [53] Koza, J.R. (1999) *Genetic programming III*, Morgan Kaufmann, San Francisco.
 - [54] Krishnamurthy, E.V. (1989) *Parallel Processing*, Addison Wesley, Reading, Mass.
 - [55] Krishnamurthy, E.V. and Krishnamurthy, V. (2006) *Multiset Rule-based-Programming Paradigm for Soft Computing in Complex Systems*, in *Handbook of Nature-Inspired & Innovative Computing*, Ed: A.Zomaya, Springer Verlag, New York, pp.77-109.
 - [56] Krishnamurthy, E.V. and Murthy, V.K. (2006). *Distributed Agent Paradigm for soft and hard computation*, Journal of Network and Computer Applications, Vol.29, pp.124-146
 - [57] Kumar, S and Bentley, P.J. (2003), *Biologically inspired Evolutionary Development*, Lecture Notes in Computer science, Vol.2606, pp.57-68, Springer Verlag, New York.
 - [58] Lauffenburger, D.A. and Linderman, J.L. (1993) *Receptors*, Oxford University Press, Oxford.
 - [59] Lee, J et al. (1993) *Principles of locomotion for simple-shaped cells*, Nature, Vol.362, pp.467-471..
 - [60] Lucena, C et al. (2004) *Software Engineering for Multi-agent Systems*, Lecture Notes in Computer Science, Vol.2940, Springer Verlag, New York.
 - [61] Maini, P.K. and Othmar, H.G. (Eds.). (2001) *Mathematical models for biological pattern formation*, Springer Verlag, New York.
 - [62] Manrubia, S.C et al. (2004) *Emergence of dynamical order*, World Scientific, Singapore.
 - [63] Marcus, G. (2004) *The birth of mind*, Perseus Group, New York, 2004
 - [64] McCulloch, A.D. and Huber, G. (2002) *Integrative Biological modelling in silico*, pp.4-19, in *in silico biological processes*, No.247, Novartis Foundation symposium, G.Bock and A.Goode, Eds, John Wiley and Sons, Chichester, U.K.
 - [65] Meyer, B. (1992) *Applying design by contracts*, IEEE Computer Vol.25(10), pp.40-52.
 - [66] Michalewicz, Z. and Fogel, D.B. (2000). *How to Solve it: Modern Heuristics*, Springer Verlag, New York.
 - [67] Minelli A. (2003), *The Development of Animal Forms*, Cambridge University Press, Cambridge, UK.
 - [68] Monk, N.A.M et al. (2001) *Spatio-temporal patterning in models of juxtacrine intercellular signalling with feedback*, in *Mathematical Models for Biological Pattern Formation*, Ed: P.K.Maini and H.G.Othmar, Springer Verlag, New York.
 - [69] Moon, F.C. (1999) *Chaotic and Fractal Dynamics*, John Wiley, New York.
 - [70] Moravitz, H.J. (2002) *The emergence of Everything*, Oxford University Press, Oxford.
 - [71] Murthy, V.K. and Krishnamurthy, E.V. (1995) *Probabilistic Parallel Programming based on multiset transformation*, Future generation Computer systems Vol. 11, pp.283-293
 - [72] Murthy, V.K. and Krishnamurthy, E.V. (2005) *Contextual information Management using Contract-based workflow*, Proc. ACM Computing Frontiers, CF'05, Ischia, Italy.
 - [73] Newman, M.E.J. (2004) *The Structure and Function of complex Networks*, Santa Fe Institute.
 - [74] Nishimura, H et al. (2003) *Neural chaos scheme for perceptual conflicts*, Lecture Notes in Artificial Intelligence, Vol. 2773, Springer Verlag, New York, pp.170-196.
 - [75] Osborne, M.J. and Rubinstein, A. (1994) *A course on Game Theory*, MIT Press, Cambridge, Mass.
 - [76] Pacino, K.M. (2002) *Biomimicry of bacterial foraging for distributed optimisation*

- and control IEEE Control System Magazine, Vol.22(3), pp.52-68
- [77] Petrucci, R.H. et al (2002) *General chemistry*, Prentice Hall, NJ.
 - [78] Pikovsky, A et al., (2003) *Synchronization*, Cambridge University Press, Cambridge.
 - [79] Pinney, J.W et al. (2003) *Petri net representations in systems biology*, Biochemical Society Transactions, Vol.31, Part 6.
 - [80] Prigogine, I. (1980) *From being to becoming*, W.H. Freeman and Co, San Francisco.
 - [81] Ransom, R. (1981) *Computers and Embryos*, John Wiley, New York
 - [82] Rose, M.R and Lauder, G.V (1996) *Adaptation*, Academic Press, New York.
 - [83] Sankar, P.V. and Krishnamurthy, E.V. (1978) *On the compactness of subsets of digital pictures*, Computer Graphics and image Processing, Vol. 8, pp.136-143.
 - [84] Schweitzer, F (2002). *Brownian Agents and Particles*, Springer Verlag, Berlin, 2002.
 - [85] Serugendo, G.D.M. et al. (2004) *Self Organization : Paradigms and Applications*, Lecture Notes in Artificial Intelligence, Vol.2977, Springer Verlag, New York, pp.1-19.
 - [86] Serugendo, G.D (2003) *Engineering Emergent Behaviour: A Vision*, Lecture Notes in Artificial Intelligence, Vol.2927, Springer Verlag, New York, pp.1-7.
 - [87] Serugendo, G.D.M et al (2006). *Self-organization and Emergence in MAS: An overview*, Informatica, Vol.30, pp.45-54.
 - [88] Shakshuki, E and Jun, Y (2004) *Multi-agent development toolkits: An Evaluation*, Lecture Notes in Artificial intelligence, 3029, Springer Verlag, New York, pp. 209-218.
 - [89] Sichman, J.S et al. (2003) *Multi-agent- based simulation II*, Lecture Notes In Artificial Intelligence, Vol.2581, Springer Verlag, New York.
 - [90] Smith, R.C (2005) *Smart Material Systems*, SIAM, Philadelphia.
 - [91] Song, C et al. (2004) *Complex Networks are Self-similar*, Preprint, Levich Institute, New York.
 - [92] Stepney, S et al. (2003) *Artificial Immune System and the grand challenges for non-classical computation*, Lecture notes in Computer Science, Vol. 2787, Springer Verlag, New York, pp.204-216.
 - [93] Stith, B.J. (2004) *Use of animation in teaching cell biology*, Cell. Biology Education, Vol.3(3), Fall, pp.181-188.
 - [94] Strauss D (1986) *On a general class of models for interaction*, SIAM Review, Vol.28, pp.513-527.
 - [95] Strogatz, S.H. (2003) *Sync: The emerging science of spontaneous Order*, Hyperion Press, New York.
 - [96] Strossel, T. (1993) *On the crawling of animal cells*, Science, Vol.260, 1086-1094.
 - [97] Suzuki, Y et al (2001) *Artificial Life applications of a class of P systems: Abstract rewriting systems on Multisets*, Lecture notes in Computer Science, Vol.2235, pp.299-346, Springer Verlag, New York.
 - [98] Szarowicz, A et al. (2001) *The application of AI to automatically generated animation*, Lecture Notes in Artificial Intelligence, Vol.2256, pp.487-494, Springer Verlag, New York.
 - [99] Szarowicz, A. et al. (2006) *Combining intelligent agents and animation*, see web.
 - [100] Szirtes, G et al (2001) *Emergence of Scale-free Properties in Hebbian Networks*, International Journal of Neural Systems, preprint..
 - [101] Tianfield, H. (2003) *A Study on the Multi-agent Approach to Large Complex Systems*, Lecture Notes in Artificial Intelligence, Vol.2773, pp.438-444, Springer Verlag, New York.
 - [102] Torquato, S. (2002) *Random Heterogeneous Materials*, Springer, New York.

- [103] Turing,A.M.(1952)*The Chemical Basis for Morphogenesis*, Phil. Trans. Roy.Soc London, Vol.237, pp.,37-79.
- [104] Watts, D.(1999) *Small Worlds*, Princeton University Press, Princeton.
- [105] Waves World (2004) *An agent- based approach to animation*, Xenia. media.mit.edu/a_wave/Ph.D.Thesis/ch4.
- [106] Weirich,P.(1998) *Equilibrium and Rationality* , Cambridge University Press, Cambridge, U.K.
- [107] Weiss,T.F(1996) *Cellular Biophysics*, Vols1 and 2, MIT Press,Cambridge,Mass.
- [108] Werfel, J et al. (2005) *Construction by robot swarms using extended stigmergy*,*Technical Report*, AI Memo AIM-2005-011,MIT,C S and AI Lab.
- [109] Wolfram,S (2002) *A New kind of Science*, Wolfram Media Inc., Champaign, Ill.
- [110] Wooley,J.C and.Lin,H.C. (Eds.) (2005) *Catalyzing inquiry at the interface of computing and biology*, National Research council of the National Academies, National Academies Press, Washington, DC.
- [111] Woolridge,M(2002) *Introduction to Multi-Agent systems*, John Wiley, New York.
- [112] Zak, M, et al. (1997) *From Instability to Intelligence*, Springer Verlag, New York..
- [113] Zaslavsky, G.M.(1999). Chaotic Dynamics and the origin of Statistical Laws, *Physics Today*, Vol.52(3),pp.39-45.
- [114] Zhang,Z and Zhang, C (2004) *Agent-Based Hybrid Intelligent Systems*, Lecture Notes in Artificial Intelligence, Vol.2938, SpringerVerlag, New York.

INDEX TERMS

AAA (Adaptive agent architecture)
Agent
ABS (Agent-based simulation)
Adaptive
Agent-based simulation (ABS)
Aglets
Animation
Avataras
AOIS
AOIS (agent-oriented System Community)
Assortative mixing
Ant colony heuristic
Animal human trails
Attractor dynamics
Autocrine signalling
Autonomous
Avalanches
Average degree

Bacterial colony
Bernoullicity
Bistability in perception

cell mobility
Cell system biology
Cell-diversification
Characteristic path length
clustering
clustering coefficient
concurrency
conflicts
correlation
Cougaar

Decoherence
 Developmental biology
 Disassortative mixing

 Enabling
 Enabling
 Emergence
 Ergodicity
 EXTRAN
 EXTRAN conflict

 Far from equilibrium System
 Fault-tolerant
 Fractal model
 Fractal dimension

 Graph growth
 Game theory
 global coupling rule

 Inductive interaction
 Inhibiting
 interaction
 INTRAN conflict
 Isoperimetric Quotient (IQ)

 Jack
 JADE
 juxtacrine signalling

 Landscaping rule
 local coupling rule
 Lyapunov Exponents

 metabolic networks
 Metric entropy
 Mixing
 Multi-agent
 multi-agent tool kits
 multiset

 Nash Equilibrium
 Nature Inspired Smart systems (NISS)

neuronal networks
NISS (Nature-Inspired Smart system)

Oscillatory Chemical Reaction

paracrine signalling
Pattern formation
Percolation
Percolation threshold
Positive entropy
Power-law scaling network
Proactive
Protein networks

Quantum mechanics
Quality Index

Random fractals
Random network
Reactive
Resilient
Revise

Scale-free distribution
Scale-free network
Self-awareness
Self-organization
Shape-memory alloys(SMA)
Signal processing in cells
Simulation
Small world network
Stability of Graph dynamics
Special purpose agents
Stepping rule
Stigmergy
Structure of an Agent
Synchronization

Tool kits(Multi-agent)
Transactional paradigm
Turing -Expressible (-non-expressible)

Wasp-nest growth

Zero entropy

Zeus