

NÉLITON ANTÔNIO CAMPOS

**ESTUDO SOBRE TEORIA, ADEQUAÇÃO E FERRAMENTAS DE
SISTEMAS COMPLEXOS COM VISTA ÀS SIMULAÇÕES
COMPUTACIONAIS**

Monografia de final de curso apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação, para a obtenção do título de Bacharel em Ciência da Computação

Orientador

Prof. José Monserrat Neto

LAVRAS
MINAS GERAIS - BRASIL
2002

NÉLITON ANTÔNIO CAMPOS

**ESTUDO SOBRE TEORIA, ADEQUAÇÃO E FERRAMENTAS DE
SISTEMAS COMPLEXOS COM VISTA ÀS SIMULAÇÕES
COMPUTACIONAIS**

Monografia de final de curso apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação, para a obtenção do título de Bacharel em Ciência da Computação

APROVADA em __ de _____ de _____.

Prof. _____

Prof. _____

Prof. José Monserrat Neto
UFLA
(Orientador)

LAVRAS
MINAS GERAIS – BRASIL

RESUMO

O presente trabalho visa estudar a teoria da complexidade e observar o andamento da aplicação da mesma pelos sistemas computacionais, com enfoque em ferramentas computacionais. De maneira geral, a teoria da complexidade é um modelo que busca tratar características estáticas e dinâmicas de um fenômeno natural em um só enfoque; para tanto, leva-se em conta tanto as relações mútuas quanto a própria auto-relação das pequenas partes possivelmente identificadas no fenômeno, podendo essas partes ser constituídas de estruturas internas adaptativas. Pretende-se chegar com esse trabalho a uma referência técnica de algumas ferramentas computacionais que funcionam como apoio científico na validação de modelos provenientes da teoria da complexidade.

ABSTRACT

This work intends to deep a study of the Complex Theory and to look into its application on computing systems, focusing on computational tools. By in large, the Complex Theory is a science model that searches statistics and dynamics features of a natural phenomena within a single focus; moreover, it considers the mutual relationships and the own self-relationships itself among many parts probaly concerned in the phenomena, being these simple parts be possible constituted of internal adaptatives structs. It intends here to reach a tecnical referece of some computing tools which works as a scientific support on the validation of models of the Complex Theory.

SUMÁRIO

1.	Introdução.....	4
2.	Revisão de Literatura.....	7
2.1.	A Teoria dos Sistemas Complexos.....	7
2.1.1.	Introdução.....	7
2.1.2.	Ordem e Caos.....	7
2.1.3.	Dinâmica.....	9
2.1.4.	Auto-organização.....	11
2.1.5.	Pré-condições da Auto-organização.....	12
2.1.6.	Estruturas de <i>Feedback</i>	14
2.1.7.	O Efeito Dominó.....	16
2.1.8.	Meta-balanceamento.....	16
2.1.9.	Sobrevivência e Uniformidade.....	17
2.1.10.	Sistemas Complexos.....	18
2.1.11.	Vórtices.....	20
2.1.12.	Ressonância.....	22
2.1.13.	Evolução.....	23
2.1.14.	Complexidade na Margem do Caos.....	25
2.1.15.	Criticalidade Auto-organizada.....	28
2.1.16.	Sistemas Complexos Adaptativos.....	29
2.1.17.	Aplicações de Sistemas Auto-organizáveis.....	30
2.2.	Adequação de Simulações Computacionais para SCs.....	31
2.2.1.	Sistema Computacional.....	31
2.2.2.	Primitivas de Computação: Partes Simples de Um SC.....	32
2.3.	Tipos de Simulações.....	32
2.4.	Ferramentas Para Simulação de Sistemas Complexos.....	34
2.4.1.	Swarm.....	34
2.4.2.	Starlogo.....	35
2.4.3.	AgentSheets.....	36
2.4.4.	Ascape.....	38
2.4.5.	MAML: (Multi-Agent Modelling Language).....	39
2.4.6.	RePast.....	40
2.4.7.	SDML: (Strict Declarative Modelling Language).....	40
3.	Material e Métodos.....	43
4.	Resultados e Discussão.....	44
4.1.	Swarm.....	44
4.2.	Starlogo.....	46
5.	Conclusão.....	48
6.	Referências Bibliográficas.....	50

1. Introdução

Que instrumento incrível é a ciência! Essa possibilitou que o ser humano pudesse fazer transformações antes ditas impossíveis na natureza. Certamente ninguém antes imaginava, por exemplo, que uma pessoa poderia voar dentro de um objeto mais pesado que o ar. E como se chegaria a transformações como essa? Poder-se-ia dizer que se chegou a essas transformações a partir da prática científica, sustentada em seus pilares, ou partes, que se resumiriam em observação, teoria e experimentação (Figura 1).

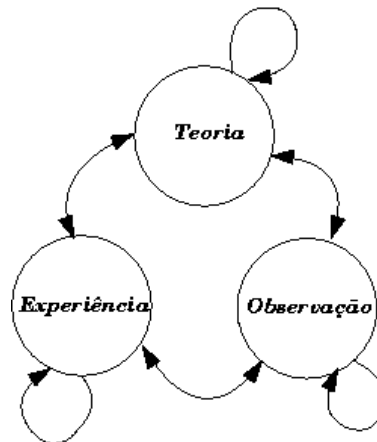


Figura 1: Um Modelo de Ciência

Da prática científica surgem inúmeros elementos que ajudam as pessoas a se relacionarem de maneira mais apropriada com a natureza. São feitos diversos enfoques com esse intuito, cada qual com a devida adequação ao que está sendo expresso a partir da realidade observada. Dentre esses enfoques, se destaca especificamente nesse trabalho, um dado modelo que busca representar, em foco único de estudo, os elementos naturais e suas transformações, chamado de Sistema Complexo (SC).

"Um Sistema Complexo pode ser livremente definido como [um fenômeno] construído por um grande número de partes simples e mutuamente

interativas, capaz de trocar estímulos com seu ambiente e capaz também de adaptar a sua estrutura interna como consequência de tal interação." [PAL02]

Esse modo de fazer ciência faz com que se possa perceber, por exemplo, que tipo de comportamento e estrutura são formados pelas partes elementares de um dado fenômeno natural. E, da mesma maneira, que tipo de partes elementares são formadas pelo mesmo fenômeno natural. Com abordagens desse tipo são formulados os modelos em SCs.

"Um modelo é uma criação cultural, um 'mentefato', destinada a representar uma realidade, ou alguns de seus aspectos, a fim de torná-los descritíveis qualitativa e quantitativamente e, algumas vezes, observáveis". [SAY01]

As formas de modelos de SCs, bem como de outros enfoques científicos, podem ser desde modelos descritivos até os representados por fórmulas matemáticas. Todos eles não deixam de estar em contínuo desenvolvimento à medida que os fenômenos se sucedem.

Mas não basta somente resgatar uma representação teórica da realidade a partir de um modelo científico: é preciso observar a influência desse modelo na natureza no intuito de aproximá-lo cada vez mais do fenômeno observado. Vários tipos de simulações podem ser feitos com intuito de auxiliar essa prática. Na teoria dos SCs especificamente, onde se busca representar de grandes quantidades de elementos, os simuladores computacionais se tornam de grande utilidade.

Programas como *Swarm*, *StarLogo*, *AgentSheets*, *Ascape*, *MAML*, *RePast*, *SDML*, dentre outros, servem como ferramentas para simulação de modelos em SCs. É possível simular modelos provenientes das Ciências Sociais por essas ferramentas, tais como modelos econômicos, por exemplo.

Esses programas se diferenciam em atributos de usabilidade, flexibilidade, compatibilidade, plataforma de máquina, suporte técnico e outras facilidades para o usuário. Além do fato de cada uma se adequar melhor a uma dada simulação. A adequação contínua dessas ferramentas aos atributos citados visa facilitar a prática científica.

A natureza se revela como um papel importante na adequação das ferramentas de simulação. Afinal, é nela que se faz as atribuições exigidas pelos modelos gerados pela ciência. Não seria isso diferente em simulação de modelos em SCs.

Nesse trabalho se apresenta inicialmente um resumo da teoria dos Sistemas Complexos, algo sobre Simulações Computacionais e uma coleta de estudos sobre as ferramentas mais freqüentemente utilizadas em simulações computadorizadas de Sistemas Complexos. A seguir, apresenta-se a proposição do trabalho, os materiais e métodos utilizados e, por fim, os resultados, discussões e conclusões deste estudo.

2. Revisão de Literatura

2.1. A Teoria dos Sistemas Complexos

2.1.1. Introdução

Sistemas Complexos (SCs), longe de serem entendidos como sistemas complicados, podem ser explicados como um paradigma científico que permite tratar aqueles fenômenos compostos de elementos mutualmente interativos e abundantes. Segundo PALAZZO, o estudo de Sistemas Complexos (SCs) cresceu muito nos últimos anos, porém o conceito de complexidade continua muito vago na literatura. E, ainda, a sobrevivência de um SC se faz pela análise sempre como um todo das partes de um fenômeno. [PAL02]

Considerando um conjunto de ferramentas computacionais um SC por exemplo, se observa que cada ferramenta pode ser considerada um elemento e, o restante do sistema computacional, em conjunto com o que está ao redor, o elemento de conexão entre as ferramentas. De modo que uma alteração em qualquer das ferramentas ou mesmo no ambiente delas provoca uma alteração na totalidade do sistema.

A imprevisibilidade do comportamento de um SC quanto a qualquer alteração em seus elementos, ou mesmo no ambiente, se torna um ponto que deve ser observado com bastante cuidado. Portando-se dessa maneira, se busca o aprofundamento do estudo de um dado fenômeno em direção à determinação de comportamentos futuros. Com essa premissa o estudo de SCs é encaminhado em direção aos seus resultados.

2.1.2. Ordem e Caos

Para se entender um SC é necessário compreender o conceito de ordem. Ordem pode ser entendida aqui como uma invariância ante às modificações

internas e externas aos elementos. A sua principal característica está na previsibilidade espacial e temporal do fenômeno ordenado. A redundância procedente da ordem faz com que o fenômeno possa ser reconstruído ou conservado sempre que necessário.

Levada ao limite esta definição, na tentativa de produzir um sistema em ordenação máxima, obtém-se um sistema caracterizado pelo fato de ser invariável sob qualquer possível transformação. A única estrutura possível para tal sistema se caracteriza por total homogeneidade: deve ser possível mapear o sistema de uma parte qualquer para outra sem que nenhuma modificação ocorra. Além disso o sistema deve possuir uma extensão infinita, porque de outro modo se poderia imaginar transformações que mapeassem de uma parte do sistema para algum elemento fora de seus limites. Em outras palavras, um sistema de ordenação máxima iria corresponder a um vácuo clássico, isto é, a uma substância estendida ao infinito na qual nenhuma parte, componente ou estrutura interna pode ser distinguida. Tal sistema é claramente o oposto do que se tomou como um sistema complexo, que se caracteriza justamente por possuir uma estrutura interna diferenciada. [PAL02]

Tendo em vista as ferramentas computacionais, a ordem estaria, por exemplo, na utilização adequada das mesmas. As ferramentas funcionam bem se manuseadas e configuradas de acordo com a especificação dada, caso houver alguma. E as ferramentas continuarão sendo executadas nos computadores enquanto for encontrada utilidade apropriada para elas.

A desordem, por outro lado, é caracterizada pela ausência de invariância, isto é, pela ausência de transformações não triviais que não teriam qualquer efeito distinguível sobre o sistema. No limite isto significa que qualquer parte do sistema, por insignificante que seja, deve ser diferente ou independente de qualquer outra parte.

Em um sistema de máxima desordem se deve ter partículas com qualquer momento físico aparecendo e desaparecendo em qualquer instante no tempo e qualquer posição no espaço. Dessa forma tanto a perfeita ordem quanto a perfeita desordem, no limite, correspondem ao vazio, isto é, à ausência de qualquer forma de complexidade.

Fica claro então que para se ter complexidade é necessário que exista algo entre a ordem e a desordem. A conexão entre as partes se torna indistinguível dentro de um SC. É como se em volta das ferramentas computacionais, por exemplo, houvesse tão somente uma “massa” de conexão.

2.1.3. Dinâmica

O passo seguinte no estudo da complexidade corresponde à análise de como um sistema complexo evolui, isto é, como ele se modifica ao longo do tempo. A invariância limitada encontrada num SC não se aplica somente a transformações geométricas ou espaciais, mas também a transformações temporais ou dinâmicas. Isto significa que certas partes ou estruturas do sistema serão conservadas durante uma certa evolução de tempo enquanto que outras irão se modificar. Até aqui esta descrição parece bastante trivial: uma parte do sistema se modifica – elementos - enquanto que outra permanece sem modificações – ambiente. É necessário entretanto algum método para identificar quais os subsistemas que irão mudar e quais os que irão permanecer inalterados.

A evolução de sistemas complexos, de acordo com esta visão, é caracterizada por um intrincado emaranhado de ordem ou invariância e desordem ou variação. Suponha-se que haja uma subestrutura relativamente invariante (isto é, não afetada por um certo conjunto de transformações) em um sistema complexo. Isto significa que enquanto os processos internos do sistema pertencerem a esta categoria de transformação a subestrutura irá permanecer invariável. Porém existe alguma forma de restrição ou seleção natural, que

mantém as estruturas mais estáveis, descartando as demais.

Os subsistemas estáveis devem ser distinguidos como indivíduos. A dinâmica de variação-restrição é o princípio sobre o qual se baseia a invariância relativa das distinções em sistemas complexos.

O princípio da variação e retenção seletiva foi elaborado a partir deste ponto de vista. [PAL02] Por definição um sistema complexo consiste em um certo número de subestruturas distintas submetidas à variação. Como a variação não é absoluta ou completa, essas subestruturas deverão apresentar alguma invariância ou estabilidade. Isto significa que as subestruturas não irão mudar todas ao mesmo tempo. Há alguma forma de inércia ou continuidade que limita a mudança. Isto permite considerar uma subestrutura como possuindo uma identidade estável durante um certo (talvez infinitesimal) espaço de tempo. O que pode então mudar durante este intervalo? Tanto as relações ou conexões entre a subestrutura considerada e as demais quanto a organização interna das partes da subestrutura mudam. O primeiro processo pode ser denominado (re)combinação e o segundo mutação.

A própria evolução das ferramentas computacionais serve de exemplo. Elas aparecem para serem utilizadas nos computadores com uma dada funcionalidade. À medida que sua utilização se torna precária, novas versões aparecem.

A utilidade desta distinção entre recombinação e mutação reside no fato de que estes processos podem em geral ser vistos em separado, ocorrem em diferentes escalas de tempo e possuem diferentes efeitos sobre níveis distintos do sistema. Se a subestrutura for intrinsecamente muito estável, então o processo de mutação será muito raro ou improvável, enquanto que o processo de recombinação pode acontecer continuamente sem afetar a sobrevivência das subestruturas envolvidas. Ambos os processos, entretanto, conduzem à variação,

que por sua vez pode conduzir a novas subestruturas que irão sendo seletivamente retidas ou descartadas pelo ambiente.

2.1.4. Auto-organização

A organização de um SC surge espontaneamente a partir da desordem e não parece ser dirigida por leis físicas conhecidas. De alguma forma a ordem surge das múltiplas interações entre as unidades componentes e as leis que podem governar este comportamento não são bem conhecidas. Percebe-se, entretanto, que o processo é não-linear e emprega simultaneamente laços de *feedback* positivo e negativo entre os componentes no nível mais baixo do sistema e entre estes e as estruturas que formam os níveis mais altos.

A perspectiva comportamental de um sistema auto-organizável poderia revelar como padrões espaciais e temporais – tais como caminhos, limites, ciclos e sucessões – que poderiam surgir em comunidades heterogêneas complexas. Os primeiros modelos de formação de padrões utilizados baseavam-se em uma abordagem *top-down* onde os parâmetros descreviam os níveis mais altos dos sistemas. Segundo este enfoque a dinâmica das populações é representada também em seus níveis mais elevados e não como resultado da atividade que ocorre ao nível mais baixo dos indivíduos.

Dizer que um sistema é auto-organizado corresponde a dizer que este não é governado por regras *top-down*, mesmo que possam existir restrições globais sobre os indivíduos componentes. Ao invés disso, ações locais e as interações entre os indivíduos são a fonte da organização dos níveis mais altos, conduzindo o sistema em direção a padrões estruturais ordenados cuja dinâmica pode ser reconhecida. Uma vez que a origem da ordem em sistemas auto-organizáveis reside nas sutis diferenças existentes entre os componentes e na interação entre eles, a dinâmica de tais sistema não pode ser entendida pela mera decomposição do sistema em partes menores.

2.1.5. Pré-condições da Auto-organização

É necessário a um sistema satisfazer diversas pré-condições e valer-se de vários mecanismos para promover a auto-organização. [PAL02] Tais mecanismos são de certa forma redundantes e pouco definidos, entretanto permitem avaliar intuitivamente o potencial de auto-organização dos sistemas. São eles:

- **Abertura Termodinâmica:** em primeiro lugar o elemento deve trocar energia e/ou massa com o seu ambiente. Um sistema deve ser termodinamicamente aberto porque de outra forma ele empregaria toda a energia disponível em si próprio. Um sistema aberto precisa portanto encontrar-se distante (às vezes não muito) do equilíbrio termodinâmico para permitir a ocorrência do fenômeno da auto-organização.
- **Comportamento Dinâmico:** se um sistema não está em equilíbrio termodinâmico, a única opção que resta para o seu comportamento é assumir algum tipo de dinâmica. Um dos tipos de comportamentos básicos verificados em sistemas auto-organizáveis é importar energia do seu ambiente, exportando em troca entropia. A idéia de exportar entropia é uma forma de dizer que o sistema não está violando a segunda lei da termodinâmica, uma vez que ele pode ser visto como uma unidade maior ambiente-sistema.
- **Interação Local:** esta condição parece ser um importante mecanismo para a auto-organização e como tal deve ser incorporada aos modelos que a representam. Nenhum componente em um sistema complexo interage com todos os outros componentes simultaneamente. Mas se diz que formam um mesmo sistema porque eles estão conectados através da estrutura global.
- **Dinâmica Não-linear:** uma alteração a partir de um local de um SC pode provocar um encadeamento de alterações que repercutem inclusive no local

de onde foi provocada a alteração. A auto-organização pode ocorrer quando existem laços de *feedback* entre as partes componentes do sistema. Ela pode ocorrer também se quando houver laços entre estes componentes e as estruturas que emergem em níveis hierárquicos mais altos.

- Grande Número de Componentes Independentes: uma vez que a origem da auto-organização recai nas conexões, interações e laços de *feedback* entre as partes dos sistemas, torna-se claro que sistemas auto-organizáveis devem possuir um grande número de componentes. Os elementos dos SCs são freqüentemente denominados agentes quando os mesmos possuem as propriedades básicas de armazenamento, transferência e processamento da informação. Modelos que consideram agentes a esse nível são denominados modelos baseados em indivíduos, enquanto que os modelos convencionais consideram a agência a nível de grupo, empregando parâmetros de acordo com o nível considerado. [HOU88]
- Comportamento Geral Independente da Estrutura Interna dos Componentes: a mesma propriedade emergente irá surgir em sistemas completamente diferentes.
- Emergência: a Teoria da Emergência diz que o todo é algo mais que a soma das partes e o todo exibe padrões e estruturas que surgem espontaneamente do comportamento das partes. A emergência indica que não há um código para uma dinâmica de nível mais alto nas partes do nível inferior. [GRE93] A emergência se encaixa bem na Teoria da Hierarquia como uma forma de descrever como cada nível hierárquico em um sistema pode seguir um conjunto discreto de regras.
- Comportamento Geral Organizado e Bem Definido: desconsiderando a estrutura interna de um sistema complexo e o observando apenas como um fenômeno emergente, se constata que seu comportamento é bastante preciso

e regular. Os fenômenos emergentes mais complicados evoluem segundo padrões bastante distintos. O comportamento geral de um elemento pode ser bastante simples e previsível, seguindo regras emergentes simples para todas as situações em que possa se encontrar.

- Efeitos em Múltiplas Escalas: a emergência também aponta para interações e efeitos entre múltiplas escalas nos sistemas auto-organizáveis. As interações em pequena escala produzem as estruturas em grande escala. E as mesmas, por sua vez, modificam a atividade na pequena escala.

Mesmo que a auto-organização ocorra em sistemas com as qualidades acima, nem todos os que as apresentam demonstram aspectos de auto-organização; e ainda não são bastante claras as razões pelas quais isto ocorre. Em outras palavras, as condições necessárias e suficientes para a ocorrência da auto-organização são ainda desconhecidas. Há entretanto estruturas muito profundas produzindo a ordem que se encontra por toda parte na natureza e que se consegue observar melhor em sistemas complexos em vários níveis de observação.

2.1.6. Estruturas de *Feedback*

Uma estrutura de *feedback* é um laço causal, uma cadeia de causas e efeitos que forma um anel. Dentre essas estruturas, a mais simples é o *feedback* de reforço, também conhecido como efeito bola-de-neve ou ciclo vicioso. A principal característica do *feedback* de reforço é o de ser auto-amplificador.

Quanto mais complexo um sistema (seres vivos, por exemplo) maior o número de estruturas de *feedback* que apresenta. Tem sido observado que sistemas que apresentam *feedback* tendem a desenvolver propriedades completamente novas. Este fenômeno, como já se viu, denomina-se emergência e as novas propriedades do sistema são ditas propriedades emergentes.

Os processos de *feedback* não são comparados a uma bola-de-neve por acaso. A metáfora da bola que rola e cresce ao mesmo tempo em que aumenta a velocidade ladeira abaixo, além de representar muito bem o fenômeno, demonstra também dois modos completamente diferentes de perceber o processo. A bola de neve apresenta dois movimentos diferentes: quando se acompanha a bola com os olhos, verifica-se que ela possui um movimento circular de rotação sobre si própria; por outro lado, quando se observa a bola rolando ladeira abaixo vê-se que sua trajetória descreve uma linha reta.

Na tradição científica ocidental se adota em geral a visão linear. O tempo é visto como passado, presente e futuro dispostos sobre uma linha. O presente é um ponto sobre esta linha, movendo-se em direção ao futuro e deixando o traço do passado atrás de si.

Na visão linear as causas estão sempre atrás dos efeitos. Isto entretanto não é o caso na visão circular, onde causa e efeito estão conectados em um ciclo. Não faz sentido falar em “na frente de” ou “atrás” em uma trajetória circular onde qualquer ponto está ao mesmo tempo na frente e atrás de outro ponto.

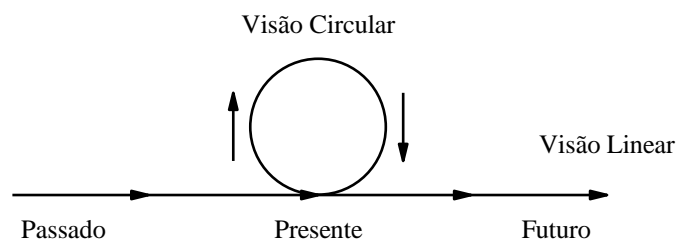


Figura 2: Duas visões complementares para estruturas de *feedback*

O efeito bola-de-neve tem a propriedade de permitir visualizar simultaneamente os dois movimentos. Infelizmente isso raramente acontece em outras estruturas de *feedback*, devido ao fato de que a visão circular focaliza a estrutura abstrata do fenômeno enquanto que a visão linear considera padrões concretos. Para ver a estrutura é necessário visualizar as relações entre os

parâmetros do sistema, razão pela qual as estruturas de *feedback* são freqüentemente invisíveis.

2.1.7. O Efeito Dominó

Causa grande impressão observar dominós caírem sucessivamente, derrubados pela queda de seus antecessores, produzindo assim uma onda. Entretanto, se este é um padrão emergente, deve haver um ciclo em algum lugar. Observando cuidadosamente o efeito dominó pode-se concluir ser o mesmo comparável ao efeito produzido por uma esfera invisível, rolando sobre os dominós e derrubando-os em seqüência.

Como se pode ver, apesar das grandes diferenças, o efeito bola de neve e o efeito dominó compartilham a mesma estrutura subjacente e resultam portanto bastante similares. Esta similaridade, entretanto, permanece invisível a menos que se adote também a visão circular do fenômeno. No caso do efeito dominó, a representação adotada pode parecer estranha, tomando o efeito (a onda) pela causa imaterial (a esfera invisível), no entanto não se deve esquecer que ambos representam apenas diferentes visões de uma mesma estrutura de *feedback*.

2.1.8. Meta-balanceamento

Os fenômenos emergentes se apresentam muitas vezes de forma inusitada. Freqüentemente, comportamentos muito organizados surgem em sistemas de extrema complexidade. Mas, ainda que os diversos fenômenos emergentes que ali ocorrem sejam muito diferentes uns dos outros, eles possuem algo em comum.

Um sistema meta-balanceado é um sistema que pode ser visto de duas diferentes perspectivas. A nível de detalhe o sistema está completamente desbalanceado, entretanto, de uma perspectiva global, o sistema parece ser estável e ordenado. O curioso aqui é que o sistema precisa estar desbalanceado

internamente para produzir ordem global.

Em termodinâmica o termo balanceamento é definido com maior precisão. Um sistema considerado balanceado ou em balanço é um sistema que não despende energia. Consequentemente um sistema está desbalanceado quando despende energia. Assim, para provocar o surgimento de fenômenos emergentes nos sistemas é necessário fazê-los despendem energia.

Além disso deve-se continuamente alimentá-los com novos componentes e energia para sustentar o meta-balanceamento. Assim o efeito dominó somente pode ser mantido enquanto houver um novo dominó em pé na frente do que está caindo. Deste modo, ao contrário dos sistemas estáveis, sistemas meta-estáveis consomem energia.

2.1.9. Sobrevivência e Uniformidade

Quando um fenômeno se auto-reproduz, avançando para o momento seguinte, ele o faz a partir de componentes diferentes. Assim, se tal fenômeno emergente for observado da perspectiva de outro qualquer, ele não será necessariamente o mesmo fenômeno. Isto é verdadeiro para todos os fenômenos emergentes.

Tem-se a sensação de sobreviver de um instante para outro na concepção dos seres humanos. No entanto o que ocorre fisicamente é um contínuo processo de substituição de componentes - onde o ciclo de substituição em nível molecular dura cerca de sete anos. Assim se refere à estrutura do sistema quando é abordado o assunto sobre sobrevivência e uniformidade.

Tabela 1
Conceitos complementares segundo diferentes visões

Visão Circular	Visão Linear
Estrutura	Padrão
Perspectiva Global	Perspectiva do Componente
Meta-balanceamento	Desbalanceamento
Esfera Invisível	Onda

2.1.10. Sistemas Complexos

A teoria da complexidade se relaciona muito de perto com a teoria dos sistemas. Ambas por sua vez estão relacionadas com a teoria do caos e com a cibernética. Esse relacionamento pode ser resumido como se apresenta, grosso modo, na Tabela 2.

Tabela 2
Objetos de diferentes teorias

	Sistemas	Comportamento
Teoria dos Sistemas	Simples	simples
Teoria da Complexidade	complexos	simples
Teoria do Caos	Simples	complexo
Cibernética	complexos	complexo

A teoria dos sistemas e a teoria da complexidade se sobrepõem e são baseadas nos mesmos princípios. Qual seria então a necessidade de duas disciplinas distintas? A razão principal parece ser o fato de que ambas pertencem a duas diferentes tradições científicas; entretanto outros motivos certamente existem.

Nem todos os sistemas são trivialmente simples. Em um sistema constituído por milhões de componentes seria quase impossível projetar uma estrutura circular descrevendo todos os possíveis laços de *feedback*. Somente é possível esquematizar o *feedback* de uma forma muito geral, como é mostrado na Figura 3.

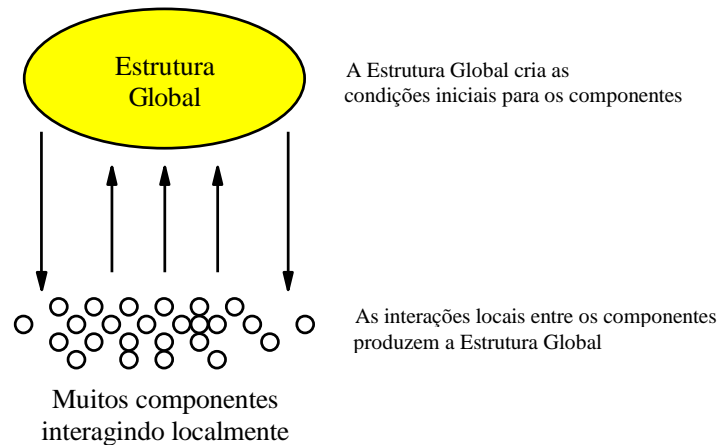


Figura 3: Feedback em sistemas complexos

Como o diagrama sugere, há um relacionamento circular entre a estrutura global do sistema e as interações locais entre os componentes. A estrutura global pode ser definida como a rede de todos os relacionamentos locais, que é produzida e mantida em um dado momento pelo total de interações que ocorrem neste momento. Cada um e todos os componentes do sistema interagem com seus vizinhos imediatos, modificando assim a estrutura global.

O comportamento de cada indivíduo é determinado pelo todo, uma vez que ele responde à estrutura global. Ao mesmo tempo, independente de todos os componentes em um dado momento, a resposta produz o todo do momento seguinte. É o que ocorre quando se acomoda uma ferramenta computacional a um dado conjunto de tarefas: ela passa a ser uma referência de montagem das simulações futuras.

Dessa maneira, um sistema complexo pode ser definido como sendo constituído por muitos componentes relativamente independentes que interagem localmente, produzindo um comportamento geral organizado e bem definido e também relativamente independente da estrutura interna dos componentes.

2.1.11. Vórtices

Os vórtices parecem ser alguma força exercida a uma grande quantidade de elementos de um fenômeno e direcionada a um lugar nesse mesmo fenômeno. Isto pode ser ainda entendido como uma simples ilusão ocasionada pelo movimento dos elementos. Se os elementos forem removidos do vórtice, resta absolutamente nada.

Entretanto, em se observando os vórtices, fica claro que existe uma força em algum lugar. Onde está ela, então? A resposta é talvez uma das mais importantes noções da ciência da complexidade: ela vem de dentro do sistema.

Ainda que na aparência uma força externa ao fenômeno esteja organizando o vórtice, são os próprios elementos em movimento que o animam. Uma das razões pela qual este conhecimento é tão importante é que ele encerrou a longa disputa entre o vitalismo e o materialismo. Os vitalistas defendiam a idéia de que a existência da vida depende de uma força vital, enquanto que os materialistas acreditavam não ser necessária nenhuma força externa para produzir vida.

O estudo dos vórtices mostra que ambas as visões estão corretas. Os vitalistas, bastante acertadamente, identificaram uma força vital que corresponde à força de sucção ilusória existente na região do vórtice. A visão materialista é também correta, uma vez que tal força vital emerge do interior do sistema.

Nada do exterior está organizando o vórtice. A força vital é real, mas não existe no sentido usual de existência, possuindo o que se denomina uma hiper-existência. Para que ocorra a hiper-existência é necessário satisfazer as seguintes condições:

1. fenômeno emergente deve estar incorporado,
2. os componentes do sistema devem estar desbalanceados, e

3. deve haver *feedback* no sistema.

Todas essas três condições são satisfeitas pelos vórtices: (1) um vórtice não pode emergir no vácuo, ele necessita estar incorporado em um meio físico. (2) Um vórtice não pode emergir a menos que os elementos do fenômeno que o compõem estejam em movimento (desbalanceados). Finalmente, (3) um vórtice é uma estrutura por si própria, possibilitando a ocorrência de *feedback*.

O que se descreveu aqui nada mais é do que o efeito de meta-balanceamento. A novidade trazida pelo estudo dos vórtices é que, uma vez que as três condições estejam satisfeitas, a meta-estabilidade se torna uma força ativa no sistema. Um sistema em estado de vórtice fica aprisionado indefinidamente a este padrão de movimento.

O conceito de vórtice é importante porque encerra uma idéia fundamental sobre o mundo. A pesquisa na área de sistemas complexos tem produzido resultados que apontam a existência de vórtices naturais no universo “impressos” na natureza. Tais vórtices permaneceriam latentes e não-existentes aguardando sua manifestação em sistemas reais.

O conceito de vórtice ainda ensina que não há apenas um nível fundamental de leis na natureza mas, na verdade, muitos. A tradição científica ocidental sempre procurou um conjunto de leis fundamentais capazes de explicar, por completo, a natureza. A ciência da complexidade eliminou esta possibilidade.

Em um mundo com um único conjunto de leis físicas sempre haveria o surgimento de novos sistemas complexos e, quando isto ocorresse, forças fundamentalmente novas acordariam de sua não-existência. Estas novas forças poderiam interagir e produzir mais forças e assim por diante. A evolução da vida é também a evolução da realidade em um universo que produz continuamente novos mundos com novas leis emergentes construídas sobre as antigas.

2.1.12. Ressonância

Além do efeito *feedback*, do efeito dominó e do vórtice, há um quarto fenômeno emergente que deve ser observado: a ressonância. A ressonância possui exatamente as mesmas propriedades das três outras estruturas. Além disso, enfatiza outras importantes propriedades dos sistemas complexos.

No sentido usual com que a palavra é empregada, ressonância é um som prolongado por um processo repetitivo. A definição técnica de ressonância é basicamente a mesma, exceto que não está restrita a ondas sonoras, generalizando a idéia para qualquer tipo de onda. O que se denominou processo repetitivo é exatamente algum tipo de *feedback*.

Uma análise do espectro deste som mostra a evolução emergente de um pico senoidal, que cresce e se torna cada vez mais alto, terminando em uma única frequência, como é mostrado na Figura 5. Este é o padrão característico de desenvolvimento de uma onda de ressonância. A novidade aqui é a redução de informação: o som original, composto de muitas frequências, termina em uma única frequência.

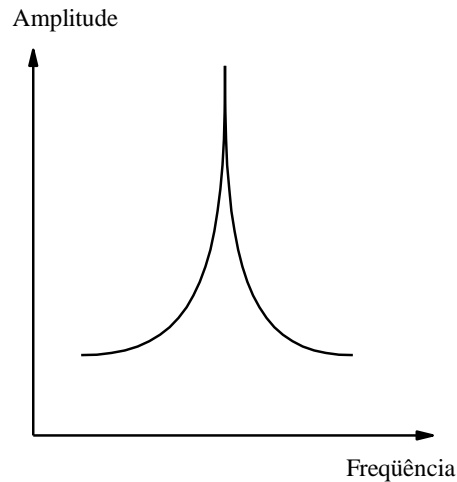


Figura 5: Ressonância como redutora de informação

Em outras palavras, a ressonância age como um *filtro emergente*. Sua estrutura de *feedback* filtra todas as freqüências deixando apenas uma. Mas, além de filtrar todas as outras freqüências, ela *amplifica* a freqüência restante.

Por esta razão a ressonância é considerada um *filtro ativo* na teoria dos filtros. Deve-se notar aqui a similaridade entre o vórtice e a ressonância. No vórtice há uma força ativa no centro que suga a matéria em sua direção. Na ressonância um sistema de freqüências é sugado e aprisionado em um único padrão.

Na definição de sistema complexo ficou estabelecido que um fenômeno emergente apresenta um comportamento bem definido. A ressonância ilustra o que isto significa. Um sistema complexo, com um grande número de dimensões, produz um comportamento emergente que pode ser traduzido por apenas umas poucas dimensões.

2.1.13. Evolução

Viu-se então que as estruturas de *feedback* podem atuar como filtros

emergentes, caracterizando processos de redução de informação. Este processo pode ser entendido como uma forma de *seleção*. Há cerca de cento e cinquenta anos atrás, o grande biólogo Charles Darwin descobriu que o mecanismo de evolução biológica correspondia exatamente a um processo de seleção, que ele denominou *seleção natural*, e que este processo nada mais era do que a sobrevivência do melhor adaptado.

Darwin viu os organismos como se fossem *máquinas perpétuas*, atravessando um processo de filtragem natural. Somente as verdadeiramente perpétuas conseguiam atravessar o filtro. O significado de uma máquina perpétua no contexto da complexidade é o de mecanismos capazes de perpetuar sua execução e de se reproduzir.

Os organismos deveriam ao menos usar a energia proveniente de seu alimento para obter mais alimento e então se reproduzir na seleção natural. Um carro está longe de ser considerado vivo por não fazer nenhum esforço próprio para obter mais gasolina para andar. Os elementos vivos que não forem capazes de obter alimento e reproduzir-se (isto é, os inadequados) serão retidos pelo filtro e serem considerados não vivos.

Recuando o olhar na história da evolução da vida sobre a terra percebe-se uma enorme diversidade de organismos. Isto ocorre porque a seleção natural não é somente um filtro, ela é também uma ressonância que amplifica os organismos adequados enquanto que os inadequados vão sendo retirados de cena. Entretanto, para ser realmente criativa a seleção natural precisa estar desbalanceada.

Levar os organismos a competir por recursos limitados também leva um sistema biológico à instabilidade. E, quando os organismos competem, eles tornam a própria adequação instável. O que hoje é adequado pode não o ser amanhã.

Um cenário de adequação dinâmica é fonte de novos fenômenos emergentes que tornam a seleção natural mais do que um mero processo de filtragem passiva. A adequação dinâmica produz *criatividade e inteligência*. Este é o fenômeno emergente mais importante de todos, porque abre o caminho para a evolução da evolução.

Sistemas complexos são *todos* constituídos de outros *todos*. Não é pois criticamente dependente de seus componentes, que por sua vez são outros *todos*. Então, para estar em meta-balanceamento, um sistema complexo precisa estar desbalanceado ao nível dos componentes.

Coloca-se um sistema complexo em desbalanceamento dando independência e liberdade a seus elementos. Sistemas complexos são meta-estáveis porque são constituídos de *todos* independentes que interagem. Quanto mais liberdades possuem os elementos, mais desbalanceado se torna o sistema e isto é fonte de mais meta-estabilidade global.

Há entretanto um preço a pagar por esta meta-estabilidade. Enquanto que por um lado ela deriva do desbalanceamento interno, a liberdade dos elementos introduz um tom de ruído no sistema. Deste modo, sistemas complexos nunca serão unidades completas.

Sempre irão ocorrer anomalias e alguns componentes se afastarão dos padrões médios dos demais. Assim, embora um organismo se apresente como uma unidade completa ele na verdade não é. Tudo isto traz claras evidências de que a natureza de alguma forma oscila entre o caos e a ordem.

2.1.14. Complexidade na Margem do Caos

Não há uma definição geral apropriada para complexidade, apesar desta ser considerada de muitas formas. Intuitivamente a complexidade se encontra em algum lugar entre a ordem e o caos. A complexidade tem sido medida

através de entropia métrica, profundidade lógica, conteúdo de informação e outras técnicas semelhantes, sendo estas medidas adequadas a aplicações específicas da química e da física mas nenhuma descrevendo completamente as características da auto-organização.

Uma forma de abordar o estudo da complexidade, considerando a ausência de uma definição satisfatória, é descrever um certo espaço, compreendido entre a ordem e o caos, denominado de “margem do caos”. [PAL02, PAC88, LAN90, KAU91 e KAU93] Em 1990 Chris Langton [LAN90] conduziu um experimento empregando autômatos celulares (AC), onde tentava descobrir sob que condições um AC simples poderia suportar primitivas computacionais, tais como transmissão, armazenamento e modificação de informações. Em seu experimento um AC unidimensional é composto por cento e vinte e oito células conectadas em círculo. Cada célula apresenta quatro possíveis estados internos e recebe todos os seus inputs das demais células em sua região, denominada sua vizinhança. A vizinhança definida por Langton era constituída por cinco células. Cada célula é considerada membro de sua própria vizinhança, juntamente com suas duas células vizinhas de cada lado. O estado interno das células no momento seguinte é determinado pelo estado de sua vizinhança e alguma função de transição que descreve qual o novo estado que a célula deve assumir para um dado estado de sua vizinhança. Assim o estado da vizinhança é associado com a transmissão, o estado interno do autômato com o armazenamento e a função de transição com a modificação da informação.

Para determinar como a ordem e o caos afetavam a computação, Langton formulou um valor λ que descrevia a probabilidade de uma dada vizinhança produzir em uma célula um determinado estado interno particular, denominado estado quiescente. Quando λ assumia o valor zero, todas as vizinhanças moviam uma célula para o estado quiescente e o sistema era

imediatamente organizado. Por outro lado, quando λ assumia o valor 1 nenhuma vizinhança se movia para o estado quiescente e o sistema mantinha-se desordenado.

O experimento de Langton mostrou a existência de um valor crítico para λ , correspondendo a pontos de transição de fase, em cuja proximidade a organização computada pelo sistema é máxima. Por outro lado, caso esse valor fosse ultrapassado, o caos surgia muito rapidamente. Segundo Langton, devido a associação da computação com tal valor crítico, um sistema auto-organizado precisaria manter-se na margem do caos para conseguir computar a própria organização.

As transições de fase nem sempre ocorrem de forma brusca como quebra entre dois estágios. Por exemplo, a passagem de um líquido do ponto de congelamento para o de ebulição se dá de forma gradual entre um estado e outro. Entretanto, a transformação do estado líquido para o gasoso nas vizinhanças de temperatura de ebulição se dá num espaço muito estreito entre os dois estados. Após um aquecimento gradual ocorre uma mudança brusca para o estado de gás, de forma que as duas fases são claramente distintas, separadas por uma estreita região que apresenta as condições de transição de fase. O estudo de tais regiões é normalmente muito útil para a previsão das propriedades do sistema ou substância em diferentes condições. As transições de fase são também locais onde interessantes fenômenos dinâmicos ocorrem, o que não se dá nas regiões de fase. Por exemplo um sólido irá normalmente absorver muito mais energia por unidade de massa e promover a quebra de ligações químicas na temperatura de fusão.

Transições de fase também ocorrem em grandes redes quando o número de conexões entre os nodos atinge um valor crítico. O grau de conectividade determina a probabilidade de um grupo de nodos conectados abranger todo o

reticulado. Quando isto ocorre diz-se que o sistema se encontra em percolação. Por exemplo, em uma rede organizada em forma de matriz a percolação ocorre com a emergência de uma propriedade conectando nodos em extremidades opostas. O limite entre redes esparsamente conectadas e redes em percolação é uma transição de fase bem conhecida. Em grandes reticulados a região de transição se torna tão estreita que é aproximada por um ponto. A percolação permite a ocorrência de relacionamentos em grande escala entre os nodos, possibilitando que nodos distantes se tornem conectados através da rede. Transições de fase e percolação ocorrem freqüentemente na natureza. O trabalho de Langton neste aspecto é particularmente importante porque aponta meios de medir e perceber as condições especiais sob as quais é possível a ocorrência de auto-organização.

2.1.15. Criticalidade Auto-organizada

Até aqui se descreveu as características e os principais mecanismos que ocorrem em sistemas auto-organizados e o experimento de Langton, que mostrou a existência de uma estreita região entre a ordem e o caos na qual a auto-organização poderia ser possível. É importante notar aqui que a maioria dos postulados apresentados se encontra ainda no terreno das hipóteses e sob intensa pesquisa. [PAL02, MIT93 e HOR95] Muitas dessas hipóteses levam a profundas conclusões.

Em um estudo pioneiro, Bak e seus colaboradores. [BAK88] investigaram o comportamento de sistemas dinâmicos espacialmente estendidos empregando simulações em computadores de um modelo que haviam desenvolvido e denominado “monte de areia”. Neste modelo grãos de areia vão sendo continuamente depositados sobre uma mesa. Em um certo momento o monte está tão alto quanto possível para sua base e então a areia escorrega pela encosta aumentando a base e permitindo ao monte continuar crescendo.

O monte de areia é muito sensível a perturbações (se a mesa for sacudida mais areia cai pelas encostas) mas os valores obtidos para a inclinação máxima do monte de areia, apesar de não se apresentarem absolutamente regulares, flutuam dependendo das condições iniciais e da perturbação exercida sobre o sistema. Devido a este equilíbrio estável, ainda que precário, Bak e sua equipe concluíram que o sistema se apresentava em estado crítico.

Além disso notaram que o sistema se auto-organizava em direção ao estado crítico sem necessidade de nenhum ajuste ou sintonia: quaisquer que fossem as condições iniciais o sistema tendia sempre ao estado crítico. Este fenômeno recebeu a denominação de criticalidade auto-organizada. Especula-se que este conceito possa ser fundamental para a escalabilidade espacial e temporal em sistemas dinâmicos.

2.1.16. Sistemas Complexos Adaptativos

Em sistemas bióticos, um elemento importante na lista de mecanismos e condições que caracterizam a auto-organização é a habilidade que os agentes apresentam de se adaptar ao meio em que se encontram. Isto significa que os agentes são capazes de alterar suas funções internas de processamento de informações. Sistemas que apresentam tal característica são denominados sistemas complexos adaptativos (SCA). [PAL02 e HOR95]

Ao se considerar a adaptação, uma série de novas questões sobre sistemas auto-organizados surgem: Quais são os mecanismos da adaptação? Sob que condições eles são possíveis? Como o sistema escolhe para onde se mover entre as possibilidades de adaptação que se apresentam? Este movimento se dá sempre na direção da criticalidade auto-organizada?

Em termos evolutivos a criticalidade auto-organizada seria obtida através das condições para a evolução (variação fenotípica individual, reprodução em excesso e herança de características genéticas). Uma população seria capaz de

adaptar-se através da herança de variações genéticas introduzidas por mutação e recombinação. Uma população poderia evoluir na direção de um estado crítico porque a seleção natural removeria as variantes do sistema que se afastassem do estado crítico.

Para melhor conduzir o sistema através do seu panorama de adequação rumo aos níveis mais altos, Kauffman empregou uma regra segundo a qual o panorama pode mover-se para estados vizinhos que oferecem níveis de adequação mais altos, rumo aos picos de adequação, mas não podem regredir a estados de adequação inferior. Segundo Ethan Decker [PAL02 e DEC97] esta regra corresponde à própria razão evolutiva que conduz à criticalidade auto-organizada e à estrutura multidimensional dos panoramas de adequação. Essa explicação evolutiva, entretanto, se torna menos convincente quando outros sistemas não dirigidos por genes são considerados, como comunidades, nichos ecológicos e sistemas abióticos (como escarpas de montanhas, linhas costeiras e climas).

Algumas outras hipóteses existem para explicar porque os sistemas auto-organizados se deslocam na direção de estados críticos, tal como a lei da produção de entropia máxima [PAL02 e SWE89] e do desequilíbrio perpétuo [PAL02 e ITO93], mas ainda se encontram no campo das conjecturas. Assim, restam como os dois mais prováveis mecanismos, responsáveis pelo surgimento de sistemas auto-organizáveis. Um é a seleção natural e as leis da física como no equilíbrio entre o atrito e, o outro, a gravidade na pilha de areia de Bak.

2.1.17. Aplicações de Sistemas Auto-organizáveis

Apesar de exigir ainda um grande esforço de pesquisa na construção de uma teoria unificada, o conceito de auto-organização apresenta excelentes perspectivas de aplicação em simulação, modelagem científica e otimização [PAL02 e DEC97]. Mecanismos encontrados em sistemas auto-organizáveis têm

sido identificados como possíveis fontes de auto-organização em diversas áreas da física, química, biologia, economia, sociologia, comunicação, informação, educação, etc.. [PAL02, BRO94 e HIE94] Diversos tipos de modelos baseados em indivíduos foram desenvolvidos e pesquisados, incluindo redes de autômatos celulares (AC) [PAL02, CAS91 e LAN94], sistemas de vida artificial (A-Life) [PAL02 e KAW94] e modelos de clareiras em florestas [PAL02 e SHU92] e o interesse em tais modelos reside na possibilidade de aplicar suas propriedades estatísticas em modelos análogos do mundo real.

2.2. Adequação de Simulações Computacionais para SCs

Exemplos de simulações de SCs seriam na área das Ciências Sociais, onde cada indivíduo poderia ser considerado uma parte "simples" do complexo, com suas características individuais inicialmente determinadas. Então se chega a uma questão: de que forma simular um modelo de SC quando a quantidade de informações é numerosa, já que em muitos fenômenos da natureza existe "um grande número de partes simples" a ser considerado? Outra questão seria em como determinar a interação mútua e as transformações internas de uma parte simples?

A última pergunta poderia ser solucionada por especialistas da área em que o elemento-parte esteja sendo cienciado, determinando suas características internas e as de relação mútua. Já, referente à primeira questão, tem-se à disposição ferramentas capazes de tratar informações abundantes e que podem auxiliar no refinamento de teorias, observações e experimentos: o sistema computacional.

2.2.1. Sistema Computacional

Sistema computacional pode ser entendido como um sistema constituído de primitivas de computação. Entende-se por primitivas de computação também

um sistema, constituído de cargas elétricas tratadas como binárias e operadas por relações lógicas de junção, intersecção e negação que se dá entre essas cargas. Tais relações se fazem através de portas lógicas que são sistemas constituídos de condutores e semicondutores.

2.2.2. Primitivas de Computação: Partes Simples de Um SC

De maneira intuitiva, a adequação da teoria da complexidade em sistemas computacionais pode se resumir na correlação das primitivas de computação à definição das partes simples dos sistemas complexos. Em outras palavras, espera-se que, se houver uma representação viável de uma parte simples de um complexo em lógica binária, se pode obter uma representação e uma simulação também viável de um SC. Um estudo exploratório pode feito nesse sentido.

A grande capacidade de processamento, ou grande capacidade de tráfego e de transformação de informações dos computadores atuais proporciona um meio cada vez mais explorado por pesquisadores para simulações de fenômenos naturais pela teoria da complexidade. Estudos de usabilidade estão sendo feitos no intuito de se facilitar tais simulações. Ferramentas, como por exemplo o *Swarm* [DUG02], estão sendo desenvolvidas nesse sentido.

2.3. Tipos de Simulações

As máquinas computacionais desprendem uma maneira bastante viável de se tratar as simulações de SCs. Vários são os tipos de simulação, cada qual adequada a cada tipo de fenômeno natural. Dentre elas, pode-se destacar as seguintes:

- Funções Matemáticas (observacionais, teóricas).
- Protótipos (experimentos).

- Maquetes (experimentos).
- Computacionais (observacionais, teóricas).
- Descritivas (teóricas).

As funções matemáticas possuem uma base mais fortemente alicerçada na descrição e na teoria. Por exemplo, as fórmulas das leis da física clássica, onde é buscado encontrar uma fórmula matemática para cada tipo de fenômeno. Essas são de apresentação mais facilmente vista em papel, ou, em plano.

Os protótipos são mais utilizados em experimentos de três dimensões. Podem ser tanto estáticos quanto dinâmicos. Esses apresentam uma visão mais aproximada do fenômeno estudado.

Maquetes têm bastante semelhança com os protótipos. A exceção é que essas são montadas principalmente em miniaturas. São utilizadas em lugares onde existe pouco espaço.

As simulações computacionais se fazem de maneira onde se busca fazer muitas contas. Tem-se os programas de computador onde, a partir de um trabalho intelectual, consegue-se a captura, tratamento e mostragem das contas que se busca realizar.

Já, as simulações descritivas, são fundamentalmente teóricas. Essas são buscadas a partir de trabalhos intelectuais. Tais trabalhos se fazem das mais diversas maneiras, nos mais diversos lugares, inclusive até na mesa de um bar.

2.4. Ferramentas Para Simulação de Sistemas Complexos

A simulação de Sistemas Complexos (SCs) exige a realização de uma enorme quantidade de operações. Isso é devida à própria definição de SCs: sistema formado de grande quantidade de elementos. Situação que os sistemas computacionais atualmente viabilizam pela capacidade de operar numerosos cálculos em tempo relativamente curto.

As ferramentas computacionais de auxílio às simulações de SCs em desenvolvimento visam aproveitar a disponibilidade de poder de cálculo dos computadores. Muitas delas estão em desenvolvimento, sendo constantemente adaptadas aos seus ambientes de utilização e de desenvolvimento. A seguir, serão descritas características de algumas dessas ferramentas: *Swarm*, *StarLogo*, *AgentSheets*, *Ascape*, *MAML*, *RePast* e *SDML*.

2.4.1. Swarm

Segundo Dugdale [DUG02], *Swarm* foi originalmente desenvolvido pelo Instituto Santa Fé especificamente para a simulação multi-agente de sistemas complexos adaptativos. Recentemente o projeto *Swarm*, baseado no Instituto Santa Fé, tem agora seu desenvolvimento e gerenciamento sob o controle do Grupo de Desenvolvimento *Swarm*.

O *Swarm* promove um conjunto de bibliotecas com as quais o desenvolvedor utiliza para formar modelos e experimentos de análise, mostragem e controle desses modelos. As bibliotecas estão escritas na linguagem *Objective C* e, até a versão mais recente, a simulação provê uma mistura de *Objective C* e *Swarm*. Entretanto, agora é possível utilizar o Java (ainda com um pouco de *Swarm*) no aproveitamento das facilidades oferecidas pelas bibliotecas. A próxima versão do *Swarm* suportará o *JavaScript* e o *Scheme* em conjunto com o *Objective C* e *Java*.

No sistema *Swarm*, o componente fundamental que organiza os agentes do modelo *Swarm* é um “enxame”. Um enxame é uma coleção de agentes com um agendamento de eventos sobre esses agentes. O enxame representa um modelo total: ele contém tanto o agente quanto uma representação do tempo. O *Swarm* suporta modelos hierárquicos onde um agente pode se composto de enxames de outros agentes em uma estrutura à parte. Neste caso, o nível mais alto das características do agente é definido pelo fenômeno emergente dos agentes dentro do seu próprio enxame. Esta aproximação de modelo multi-nível oferecido pelo *Swarm* é muito potente. Enxames múltiplos podem ser utilizados para modelar agentes que constroem modelos de seus mundos por si próprios. No *Swarm*, os agentes podem possuir seus próprios enxames, modelos que um agente monta para o próprio entendimento de seu mundo.

2.4.2. Starlogo

É um ambiente de modelagem programável especificamente voltado para explorar sistemas descentralizados através de simulação. O *Starlogo* é uma versão especializada do Logo (o qual é utilizado para ensino em algumas escolas). O *Starlogo* permite o usuário criar e controlar o ambiente de “tartarugas” (um termo mantido pelo Logo ultimamente). As tartarugas se movem numa paisagem definida pelo usuário que é montada a partir de “remendos”.

Apesar do *Starlogo* ser considerado “baseado em agente” (por exemplo, uma tartaruga é um agente), seu paradigma de programação é procedimental (um oposto do objeto-orientado, por exemplo). O que isso significa na prática? De maneira bem geral, existem vários paradigmas de programação principais (por exemplo: procedimental, objeto-orientado, declarativa, etc.). Uma linguagem de programação pode ser categorizada em um dos paradigmas. Um meio de categorizar uma linguagem é o ‘estilo’ de programação que ela adota. Assim,

Pascal, *Modula2* e *Basic* são linguagens procedimentais; *C++*, *Java* e *SmallTalk* são objeto-orientados e *Prolog* é declarativo. Existem muitas diferenças entre os diferentes paradigmas, mas o principal ponto nessa documentação é que se uma pessoa tem experiência com um paradigma de programação em particular, ele ou ela terá que aprender um novo meio de pensar sobre o problema para ser capaz de modelar e implementar usando outro paradigma. O *Starlogo* é procedimental. Ele promove um conjunto de comandos nos quais o programador utiliza para criar e controlar as tartarugas e remendos.

Na prática, o *Starlogo* é muito fácil de utilizar (em especial para pessoas que têm um pouco de experiência em programação). O *Starlogo* promove uma interface gráfica pra ajudar o desenvolvedor a implementar suas simulações. É muito fácil criar gráficos de dados de simulação e definir botões e barras de rolagem que controlam a simulação e definem a entrada de dados (ex: número de tartarugas). No entanto, apesar de ser muito fácil apresentar dados em gráficos no *Starlogo* existem alguns problemas que aborrecem com a facilidade dos gráficos (por exemplo, a pesar de muitas linhas serem impressas no mesmo gráfico, não é possível criar mais de um gráfico). Com a versão atual do *Starlogo* (ver. 1.2), é bastante fácil inserir suas simulações numa página da internet como um *applet* de visualização.

2.4.3. AgentSheets

AgentSheets é uma ferramenta, também baseada em agentes, alicerçada numa aproximação de planilha eletrônica. Ao invés das células da grade da planilha eletrônica ser ocupada de números, elas são ocupadas por agentes. As simulações então ocupam seus lugares na grade na qual os agentes vivem. O *AgentSheets* é voltado especificamente para não-programadores e, como uma consequência, é muito simples de utilizar. O *AgentSheets* utiliza o paradigma

de programação visual, significando que existe nenhuma implementação baseada em texto atualmente e todos os desenvolvimentos são realizados por interface gráfica (selecionando e arrastando elementos das caixas de ferramentas, etc.). Realmente, a facilidade de utilização do *AgentSheets* é sua grande vantagem. Os agentes são criados numa janela chamada 'galeria' e têm um ambiente associado especificado por um conjunto de regras (chamado de métodos) e eventos.

A maneira que o *AgentSheets* opera é intuitivamente fácil de entender, o que torna muito rápido o desenvolver de simulações simples. Por essa razão, ele é largamente utilizado para ensinar os princípios da simulação para estudantes. No entanto, quando os modelos de simulação são mais aprofundados, a fraqueza do *AgentSheets* se torna clara. Com respeito à simulação nas ciências sociais, duas limitações particulares do *AgentSheets* podem causar problemas: um agente não pode enviar informação para outro agente (isso seria problemático se nós precisássemos modelar a comunicação de informação entre agentes humanos numa simulação); um agente não pode mudar o atributo de outro agente (isso poderia ser problemático se nós precisássemos modelar uma situação onde um humano influenciasse outro humano). Existem meios de superar tais problemas, mas se se necessitasse modelar frequentemente essas situações nas simulações, isso iria tornar as simulações complicadas e ineficientes. Além do mais, não existe visão de "longa distância", tornando-se impossível de examinar o estado de um agente em qualquer lugar na grade. Em nível técnico, a principal desvantagem do *AgentSheets* é que ele é executado somente em Mac, embora uma versão beta de tentativa disponível para PC.

Em conjunto com a facilidade de utilização, a outra principal vantagem é que os modelos de simulação podem rodar através de um navegador de internet. O *AgentSheets* gera muito facilmente *Applets Java* e *Beans*, os quais permitem

as modelagens de simulações serem executáveis por um navegador de internet.

Inúmeros modelos de simulação são disponíveis pela página de internet da *AgentSheets* (completo, com o código fonte).

2.4.4. Ascape

O *Ascape* é um ambiente de trabalho para desenvolvimento e análise de modelos baseados em agentes o qual foi desenvolvido pelo Instituto Brookings (O Centro em Dinâmicas Sociais e Econômicas que desenvolveu a simulação bem conhecida Sugarscape).

O *Ascape* segue algumas das idéias que permeiam o *Swarm* (ex. os agentes existem com “*scape*” no qual pode por si só ser tratado como um agente). No entanto, é mais fácil de desenvolver modelos com o *Ascape* do que com *Swarm*. Sendo assim, permite a não programadores desenvolverem simulações bastante complexas, promovendo uma faixa de ferramentas de usuário final (ex.: facilidades de obter estatísticas de simulações executáveis, ferramentas de criação de gráficos, etc.).

O *Ascape* é implementado em Java e o desenvolvimento de modelos de simulação no *Ascape* poderia requerer algumas habilidades para programar em Java junto com algum conhecimento da filosofia objeto-orientada. Para o desenvolvimento prático de modelos de simulação, seria útil baixar e utilizar um Kit de Desenvolvimento Java. Tais kits são disponíveis livremente na inter-rede. A base Java do *Ascape* permite que ele seja orientado a objeto e que possa rodar em qualquer plataforma Java-habilitada. No mais, modelos desenvolvidos podem ser publicados na inter-rede e existe também uma facilidade em *Ascape* (um ‘botão de câmara’) para montar filmes de modelos dinâmicos.

Em termos de aplicabilidade do *Ascape* para simulação de ciências sociais, não haveria problemas em implementar mecanismos sociais bastante

complexos. Como o *Swarm*, a única restrição seria encontrar um programador com habilidades suficientes para implementar os mecanismos.

2.4.5. MAML: (Multi-Agent Modelling Language)

O MAML (Linguagem de Modelagem Multi-Agente) foi desenvolvido pelo Laboratório de Sistemas Adaptativos Complexos na Universidade Européia Central, na Hungria. A linguagem foi desenvolvida inicialmente para ajudar os estudantes de ciências sociais com experiência limitada em programação a montarem rapidamente modelos baseados em agente. O presente passo do projeto está em desenvolver um ambiente fácil de utilizar (completo com gráfico e interface). Entretanto, a versão atual do MAML está, como o nome sugere, numa linguagem de programação e não em um ambiente.

O MAML atualmente se situa, como linguagem de programação, acima do *Swarm* e se tem a intenção de fazer o *Swarm* fácil de ser utilizado pela promoção de chaves-de-macro que definem a estrutura do simulador e acesso às bibliotecas do *Swarm*. MAML trabalha no nível mais alto de abstração do que o *Swarm* com construções mais claras. No entanto, além de aprender MAML, o desenvolvedor necessitaria de conhecer *Objective C* e também o *Swarm*. Esse ponto limita correntemente a utilidade do MAML para programadores inexperientes. Por outro lado, programadores experientes podem preferir atualmente a funcionalidade adicional do *Swarm* e os recursos adicionais disponíveis. A programação utilizando o MAML requer ao desenvolvedor montar arquivos textos utilizando um editor de texto desde que não haja interface de desenvolvimento para MAML. Pelo fato do MAML acessar as bibliotecas *Swarm*, sua interface do modelo de simulação desenvolvido é muito similar ao que parecia ser na utilização do *Swarm*.

O código escrito utilizando *Swarm* é convertido em uma aplicação *Swarm* através de um compilador *Swarm* (chamado xmc). A aplicação

resultante é então compilada do mesmo modo que o código *Swarm* normal pelo *gcc*. Correntemente, o compilador MAML apenas executa em PC/LINUX.

2.4.6. RePast

O *RePast* foi desenvolvido no Laboratório Computacional de Pesquisa em Ciências Sociais da Universidade de Chicago especialmente para montar simulações baseadas em agentes. Ele é muito parecido com o *Swarm*, tanto em filosofia quanto em aparência e, como o *Swarm*, ele promove uma biblioteca de codificação para montagem, execução apresentação e coleta de dados das simulações.

Um dos objetivos do *RePast* é suportar a modelagem de sistemas, agentes, organizações e instituições vistas como construções sociais recursivas. Um segundo objetivo é permitir simulações estóricas localizadas para serem novamente mostradas com concepções alteradas. Ambos os objetivos fariam o *RePast* particularmente adequado para simulações em Ciências Sociais. Infelizmente essas metas são extensões futuras e não são realizadas na versão atual.

O *RePast* é baseado em Java e o desenvolvimento de uma simulação requer a habilidade de programar em Java. O *RePast* provê um pouco e bem conhecido modelo de simulação por demonstração tal como o *SugarScape*, Baratas do *Swarm* e os modelos *MouseTrap*. Infelizmente, existem muito poucas outras simulações, geralmente disponíveis na inter-rede. No entanto, existe uma lista de correio eletrônico a qual provê aos usuários suporte e discussões gerais.

2.4.7. SDML: (Strict Declarative Modelling Language)

O SDML (Linguagem de Modelagem Declarativa Restrita) não é um ambiente mas uma linguagem de programação declarativa contendo características de orientação por objeto e um princípio baseado em lógica. O

conhecimento é representado em bases de regra e base de dados é o principal mecanismo de pesquisa utilizado é a busca progressiva e retroativa. Aos agentes devem ser atribuídos regras que determinam seus comportamentos e as quais podem ser compartilhadas com outros agentes. Ao restante é possível designar às características orientadas por objeto do SDML. O fato dele ser fortemente baseado em lógica permite que provas formais de completude do modelo sejam construídas. A programação é conduzida em SDML por uma série de janelas ('Uma introdução ao SDML, um curso rápido para iniciantes' na Página do SDML promove uma boa mostragem da interface).

Simulações sofisticadas podem ser construídas utilizando SDML, envolvendo organizações interativas complexas, construções dos próximos níveis de agentes e a habilidade dos agentes possuem habilidades cognitivas limitadas. Por outro lado, como o desenvolvedor admite, a linguagem contém uma curva de aprendizagem estreita.

Mesmo que o SDML seja desenvolvido especificamente para construir simulações em Ciências Sociais, muitos dos modelos disponíveis são direcionados à economia e modelagem de marketing. Proveniente da parte da Página do SDML na Universidade Metropolitana Manchester, existem muito poucos exemplos de simulação utilizando o SDML.

O SDML no entanto promove características úteis em modelagem de agentes sociais cognitivos. Não existe nenhuma teoria herdada de cognição implementada em SDML mas qualquer cognição de um agente é representado com configurações e regras. A comunicação entre agentes é atribuída por banco de dados: o resultado de uma regra em execução é escrita em banco de dados de um agente a qual pode ser acessada por outro agente. O acesso de um banco de dados de um agente para outro banco de dados de outro agente pode ser restringida por designações de um estado de cláusula de regras (ex.: privado ou

público). Os agentes podem também se avaliar um ao outro como possíveis “colaboradores” e aceitar outros agentes como uns colaboradores reais, irreais, sucessíveis ou não sucessíveis.

O SDML é disponível para MS Windows 3.1/95/98/2000/NT, Linux, Intel, PowerMac, Unix ADUX/AIX/HPUX/SGI/Solaris.

3. Material e Métodos

A teoria da complexidade é recente e se observa uma demanda na adequação da mesma em sistemas computacionais. Isso faz surgir diversas ferramentas computacionais para observar, teorizar e experimentar os fenômenos ditos Complexos. Tendo em vista os aspectos de computabilidade e usabilidade das ferramentas computacionais para Complexos, a proposta do presente trabalho é fazer um estudo na teoria da complexidade para observar e experimentar a adequação dos sistemas computacionais, mais especificamente as ferramentas computacionais, à referida teoria da complexidade e buscar, assim, conclusões científicas. Uma avaliação preliminar das ferramentas *Swarm* e *StarLogo* serve de base deste trabalho, visto que estas se apresentam no Departamento de Ciência da Computação da UFLA como referências mais conhecidas.

O estudo será tanto qualitativo, referindo-se aos aspectos de usabilidade e adequação das ferramentas. Para isso, realizou-se o estudo teórico as bases na da teoria da complexidade, na dos sistemas computacionais aplicados, buscando assim uma adaptativa correspondência teórica.

Comprovada tal correspondência, um estudo de usabilidade das ferramentas de tratamento de sistemas complexos servirá para observar a facilidade e adequação humana na utilização delas.

4. Resultados e Discussão

Nesse trabalho foi extraído da literatura alguns quesitos sobre a teoria dos Sistemas Complexos (SCs) que dizem questão às observações de fenômenos naturais, compostos de pequenas partes. Tais partes, teoricamente, são simples e abundantemente numerosas. Baseado nisso, a teoria dos SCs está sendo delineada.

Pretende-se, a seguir, apresentar uma abordagem preliminar de ferramentas de simulação, conforme os seguintes critérios: forma de apresentação dos resultados, portabilidade, facilidade de obtenção, facilidade de modelagem dos Sistemas Complexos, facilidade de instalação.

4.1. Swarm

A forma de apresentação dos resultados do *Swarm* (Figura 6) é expressa através de interface gráfica moldada pelo próprio desenvolvedor. Através de bibliotecas o usuário pode fazer com que os resultados sejam apresentados da forma desejada. Dentre essas formas, tem-se as apresentações tanto gráficas quanto textuais.

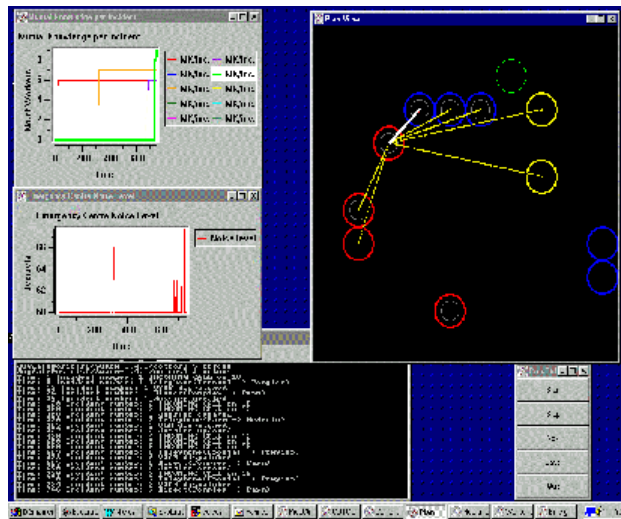


Figura 6: Uma Amostra de Tela Típica de Uma Simulação Desenvolvida Utilizando o Swarm

Quanto à portabilidade, o *Swarm* executa em qualquer máquina, ou sistema, que contenha a máquina virtual Java, ou mesmo em qualquer outra plataforma. O *Swarm* é bastante potente e flexível, com o qual poderia ser possível implementar mecanismos sociais muito intrigados e complicados. Versões para os sistemas operacionais Unix, Windows e outros são encontradas.

A obtenção do *Swarm* se faz pela inter-rede, preferivelmente nas páginas oficiais da ferramenta. Ele pode ser encontrado tanto em pacotes de programas quanto em blocos de montagem (rpm's). Existe também um respeitável suporte técnico ao redor dessa ferramenta, permitindo um melhor manuseio.

A modelagem é feita em linguagem *Objective C* ou mesmo em *Java*. Apenas a finitude da implementação faz com que a barreira seja implementada ao inverso. Como nas outras ferramentas, a população se mantém constante.

Já a instalação se torna facilitada quando se busca a mesma por pacotes completos. Os pacotes de montagem são recomendados quando se tem uma pequena parte de tempo para ser utilizado e quando se tem uma parte

considerável o bastante de pacotes já instalada. Em suma, o senso crítico irá dizer o que será melhor.

4.2. Starlogo

O *Starlogo* é apresentado de modo intrinsecamente semelhante ao *Swarm* (Figura 7). A interface busca facilitar a mostragem dos dados para as pessoas leigas. É como se houvesse um vitrô semi-transparente, transpassando o que está escrito em contos e contas.

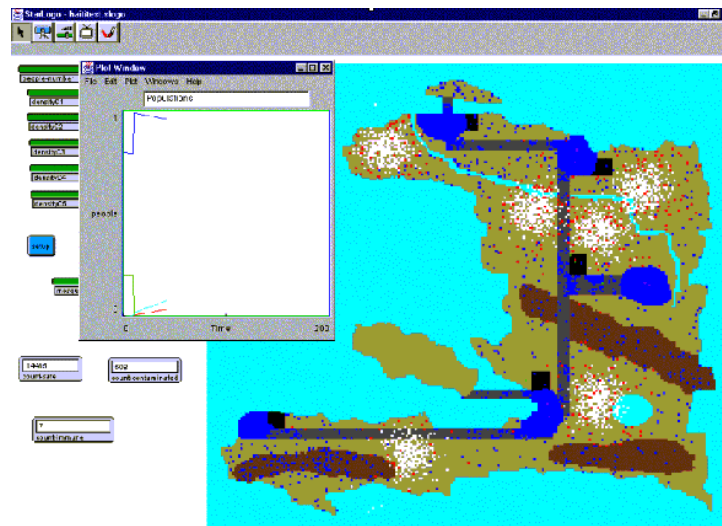


Figura 7: Exemplo de Uma Interface de Um Modelo de Simulação Desenvolvido Utilizando Starlogo

A portabilidade se dá nas máquinas Mac ou PC. No entanto, segundo Dugdale [DUG02], simulações às quais são desenvolvidas e executadas em uma plataforma irão muito, muito pouco rodar em outra. Isso faz a mostra da limitação do StarLogo.

A obtenção pode ser feita através das páginas de inter-rede, considerando-se a facilidade de apresentação das mesmas tão quanto a veracidade das mesmas. Em <http://www.media.mit.edu/starlogo/>, a página

oficial do *Starlogo*, existem várias ligações interessantes para se realizar. Vale a pena observar essa tal ligação.

A modelagem pode ser feita tanto em linguagem procedimental quanto em linguagem declarativa. Interfaces gráficas auxiliam também fornecendo quadros de inserção e absorção de regras e de apresentação dos resultados. Assim fica viável aos cientistas de SCs, leigos em computação, resolverem seus assuntos.

A instalação feita em Windows foi realizada sem problema algum nas máquinas do Departamento de Ciência da Computação. Através de um pacote de programas de computador, o *Starlogo* foi embrulhado e posto para ser distribuído na comunidade dos arredores da computação. Assim foi capturada tal ferramenta da inter-rede para o manuseio apropriado.

5. Conclusão

Com a representação a partir do modelo em SC, se pode chegar a uma maneira inter-relacional de fazer ciência, pois com a mesma se busca unir tanto características estáticas quanto características dinâmicas de um elemento posto em análise.

A observação de fenômenos complexos pode ser feita de maneira direta, ou a partir de experimentos. No entanto, a observação direta às vezes se torna inviável. Sendo assim, se torna necessário a utilização de ferramentas para esse trabalho.

Felizmente, existe a possibilidade de refinar as teorias fazendo observações e experiências através de simulações. Entende-se aqui por simulação uma experiência baseada em um modelo, realizada de maneira viável. Dessa forma que se torna possível, por outro exemplo, determinar características de um polímero antes mesmo de sintetizá-lo. Isso pode ser aplicado nos modelos de SC, caso houver um meio de simulação adequado.

As ferramentas computacionais têm o objetivo de facilitar o uso do computador para uma dada aplicação em específico. No entanto tem-se que o conceito de usabilidade ou facilidade de utilização apresenta-se um tanto quanto subjetivo, tendo em vista a diversidade de características individuais dos seres humanos. No entanto, pesquisas feitas por pesquisadores da área de Interface Homem Máquina, e áreas afins, observam tendências em comum em alguns aspectos de usabilidade que podem ser detectados nas ferramentas de simulação.

Respectivo ao SC, tais ferramentas podem se vistas como a busca de uma forma útil e abrangente de validação científica. Isso porque se consegue obter informações em abundância, o bastante para a verificação e a adequada aproximação de certos modelos à natureza. Dessa maneira, as ferramentas

computacionais se apresentam como uma das formas de se fazer simulações de maneira fácil que permitam resultados mais tangíveis.

Pelo fato da pesquisa na área da teoria da complexidade ser relativamente nova, tem-se que a usabilidade das ferramentas computacionais e adequação da teoria da complexidade aos sistemas computacionais são elementos de muito estudo e pesquisa. Acredita-se que muitas ferramentas ainda estão sendo testadas e carecem de análises estruturais, operacionais e análises de adequação. Deduz-se ainda que existem muitos materiais teóricos para análise e validação científica de um determinado assunto aonde também existir pesquisas abundantes.

6. Referências Bibliográficas

- [BRO94] BROWN, J. H. *Complex ecological systems*. Pages 419-443 in G. Cowan, D. Pines, and D. Meltzer (eds.). *Complexity : metaphors, models, and reality*. Addison-Wesley, Massachussets, 1994.
- [CAS91] CASWELL, H. & COHEN, J. E. *Communities in patchy environments: a model of disturbance, competition and heterogeneity*. Pp. 97-122 in J. Kolasa and S. T. A. Pickett (eds.): *Ecological Heterogeneity*. Springer, New York, 1991.
- [CHA75] CHAINTIN, G. J. *A theory of program size formally equivalent to information theory*. *Journal of ACM* 13(1966) *ibid* 22(1975). pp. 329. 1975.
- [CRU94] CRUTCHFIELD, J. P. *Is anything ever new?* Em G. Cowan, D. Pines e D. Melsner (eds.): *XIX Estudo SFI de Complexidade*. Adisson-Wesley, Massachussets, 1994.
- [DEC97] DECKER, N. *Self organizing systems*. University of New Mexico. *Biology 576: Landscape Ecology & Macroscopic Dynamics*. Albuquerque, 1997.
- [DUG02] DUGDALE, Julie. *An evaluation of seven software simulacion tools for use in the social sciences*. CRIC-IRIT, Toulouse, France. Antes de 2002.
- [HIE94] HIEBELER, D. *The swarm simulation system and individual-modeling based*. In *decision suport 2001: advanced tecnology for natural resource management*, 1994.
- [HEY88] HEYLIGHEN, F. *Building a science of complexity*. Anal da Conferência da Sociedade Cibernética de 1998. Loding, 1998.
- [ITO93] ITO, K. & GUNJI, Y. *Self-organisation of living systems: towards criticality at the edge of caos*. *BioSystems*, 1993. pp. 17-24.
- [KAW94] KAWATA, M. & TOQUENGA, Y. *Artificial individuals and global patterns*. *Trends in Ecology and Evolution*. 1994. pp. 417-421.

- [KAU91] KAUFFMAN, S. A. *Coevolution to the edge of chaos: coupled fitness landscapes, poised states, and coevolutionary avalanches*. Journal of Theoretical Biology, 1991. pp. 467-505.
- [KAU93] KAYFFMAN, S. A. *At home in the universe*. Oxford University Press, New York, 1995.
- [KOL65] KOLMOGOROV, A. N. *Three approaches to the quantitative definition of information*. Problems of Information Transmission, v1 n. 1, 1965. pp. 1-1.
- [PAC88] PACKARD, N. H. *Adaptation toward the edge of chaos*. Em A. J. Mandell, J. A. S. Kelson e M. F. Shlesinger (eds.) *Dynamic Patterns in Complex System*. World Scientific, Singapore, 1988. pp. 293-301.
- [PAL02] PALAZZO, Luiz A. M. & CASTILHO, José M. V. *Sistemas complexos e auto-organização*. Antes de 2002.
- [LAN90] LANGTON, C. G. *Computation at edge of chaos: phase transitions and emergent computation*. Physica D 42. 1990. pp. 12-37.
- [LAN94] LANGTON, C. G. *Artificial life iii*. Addison-Wesley, New York, 1994.
- [SAY01] SAYÃO, Luís Fernando. *Métodos teóricos da informação: abstração e método científico*. Ci. Inf. Brasília, v. 30, n. 1, jan-abr, 2001. pp. 82-91.
- [SWE89] SWENSON, R. *Emergent attractors and the law of maximum entropy production: foundations to a theory of general evolution*. Systems Research, 1989. pp. 187-197.
- [SHU92] SHUGART, H. H. & SMITH, T. M. & POST, W. M. *The potential for application of individual-based simulation models for assessing the effects of global change*. Annual Review of Ecological Systems, 1992. pp. 15-38.
- [url02] Url: <http://www.cea.uba.ar/aschu/intro.html>. Pesquisado em outubro de 2002.