

André Kenji Horie  
Paulo Muggler Moreira  
Ricardo Henrique Gracini Guiraldelli  
Virgílio Vettorazzo

*Proposta de Metodologia para Análise da  
Complexidade de Sistemas Computacionais  
em um Âmbito Global e Evolutivo*

São Paulo

2008

André Kenji Horie  
Paulo Muggler Moreira  
Ricardo Henrique Gracini Guiraldelli  
Virgílio Vettorazzo

*Proposta de Metodologia para Análise da  
Complexidade de Sistemas Computacionais  
em um Âmbito Global e Evolutivo*

Monografia para o curso PCS 2058 - Engenharia de Informação.

ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO  
DEPARTAMENTO DE ENGENHARIA DE COMPUTAÇÃO E SISTEMAS DIGITAIS

São Paulo

2008

# *Abstract*

This work proposes a model for assessing complexity of computer systems in a global and evolutive scope. This will be achieved by defining relevant metrics and the way they relate among themselves, providing a quantifiable measure for complexity. Moreover, the proposed model will be applied to available data, thus creating a valid complexity curve whose tendency will be analysed.

Keywords: Computer Systems Complexity

# *Sumário*

<b>1</b>	<b>Introdução</b>	p. 4
1.1	Objetivo . . . . .	p. 4
1.2	Conceitos Teóricos . . . . .	p. 4
<b>2</b>	<b>Definição do Modelo</b>	p. 6
2.1	Variáveis Relevantes . . . . .	p. 6
2.2	Modelo Matemático . . . . .	p. 7
<b>3</b>	<b>Aplicação do Modelo</b>	p. 8
3.1	Dados Históricos . . . . .	p. 8
3.2	Curva de Complexidade . . . . .	p. 9
<b>4</b>	<b>Conclusão</b>	p. 10
4.1	Avaliação do Modelo . . . . .	p. 10
4.2	Avaliação da Curva de Complexidade . . . . .	p. 10
	<b>Referências</b>	p. 11

# 1 *Introdução*

## 1.1 Objetivo

Este trabalho tem por objetivo analisar a complexidade em sistemas computacionais em um âmbito global e evolutivo através da definição de uma metodologia, identificando-se métricas relevantes para o cálculo da curva de complexidade. Aplicando-se o modelo proposto, espera-se verificar a tendência da complexidade em função do tempo.

## 1.2 Conceitos Teóricos

Para um estudo sobre a complexidade de sistemas computacionais, vê-se a necessidade de se definir o conceito de complexidade, para então reduzir esta definição ao escopo de sistemas computacionais e, assim, poder considerar os aspectos relevantes para a criação do modelo.

### 1.2.1 Complexidade

Warren Weaver, em seu artigo “*Science and Complexity*” (WEAVER, 1948), introduziu o conceito de complexidade na literatura científica como o grau de dificuldade de se prever as propriedades de um sistema se as propriedades de cada parte for dada. Classifica-se então a complexidade sistêmica em organizada e desorganizada. Os sistemas de complexidade desorganizada caracterizam-se pelo número elevado de variáveis e pelo seu comportamento caótico, embora as propriedades do sistema como um todo possam ser entendidas utilizando-se métodos probabilísticos e estatísticos. A complexidade organizada, por outro lado, refere-se a interações entre as partes constituintes do sistema, sendo o comportamento deste redutíveis às interações, e não às propriedades das partes elementares. A visão proposta por este artigo influenciou fortemente o pensamento contemporâneo acerca da complexidade.

Empiricamente, observa-se que uma proporção alta dos sistemas complexos encontrados na natureza possuem uma estrutura hierárquica. Em teoria, espera-se que qualquer sistema

complexo seja hierárquico, tendo como a decomposição uma propriedade de sua dinâmica (SIMON, 1962). Esta simplifica tanto o estudo do comportamento como a descrição desses sistemas.

Em 1988, Seth Lloyd afirma que a complexidade de uma propriedade física de um objeto é função processo ou conjunto de processos responsáveis por sua criação (LLOYD, 1988). Em outras palavras, a complexidade é uma propriedade da evolução de um estado, e não do estado em si. Consequentemente, uma medida da complexidade deve classificar sistemas em estados aleatórios como de baixa complexidade, e quantificar a evolução deste sistema para seu estado final.

## 1.2.2 Organização de Sistemas Computacionais

O termo “arquitetura” é amplamente utilizado para se referir à estrutura na qual um sistema é organizado. Em Tecnologia da Informação, sistemas computacionais são representados pela arquitetura de hardware, de software, de rede e de informação (GENTLEMAN, 2005). Em relação ao escopo coberto por cada uma delas, observa-se que a arquitetura de hardware é essencialmente local, sendo assim definido para apenas um nó do sistema, enquanto as arquiteturas de rede e de informação requerem necessariamente a interação entre diversos componentes. A arquitetura de software, no entanto, pode tanto indicar a organização do software em apenas um componente como também em diversos componentes distribuídos, quando aplicável. Esta decomposição vê-se necessária para melhor endereçar a complexidade de cada vertente de um sistema computacional.

## 2 *Definição do Modelo*

### 2.1 Variáveis Relevantes

A seguir serão apresentadas as variáveis relevantes para o modelo de complexidade em termos globais.

#### 2.1.1 Arquitetura de Hardware

Valores representativos para o cálculo de complexidade relacionada à arquitetura de hardware incluem métricas normalmente utilizadas para *benchmarks*, como por exemplo o número de instruções por segundo do processador, a capacidade de armazenamento das memórias de acesso aleatório, cache e do disco rígido.

Estes valores são proporcionais?

Contudo, deve-se observar que a arquitetura de hardware possui escopo local, como já mencionado anteriormente. Para transpor este valor para um âmbito global, assume-se uma proporção direta entre o aumento de complexidade e o aumento de custo de produção. Dado o custo total de produção como  $C_T(Q) = C_F + C_V = C_F + Q \cdot C_{Mg}$ , onde  $C_T$  é o custo total,  $C_F$  é o custo fixo,  $C_V$  é custo variável,  $Q$  é a quantidade produzida e  $C_{Mg}$  é o custo marginal. Portanto, a complexidade da arquitetura de hardware global é dada por  $\mathbb{C}_{HW} = f(Q) \cdot \mathbb{C}_{HW,local}$ , com  $f(Q)$  sendo o fator de aumento linear da complexidade em função da quantidade produzida.

#### 2.1.2 Arquitetura de Software

(KEARNEY et al., 1986), (RANGANATHAN; CAMPBELL, 2007).

### **2.1.3 Arquitetura de Rede**

.

### **2.1.4 Arquitetura de Informação**

.

## **2.2 Modelo Matemático**

.



## 3 *Aplicação do Modelo*

### 3.1 Dados Históricos

#### 3.1.1 Arquitetura de Hardware

Processador	MIPS	Frequência	Ano
Intel 8080	0,64	2 MHz	1974
Motorola 68000	1,0	8 MHz	1979
Intel 386DX	8,5	25 MHz	1988
Intel 486DX	54	66 MHz	1992
PowerPC 600s (G2)	35	33 MHz	1994
Intel Pentium Pro	541	200 MHz	1996
ARM 7500FE	35,9	40 MHz	1996
PowerPC G3	525	233 MHz	1997
Zilog eZ80	80	50 MHz	1999
Intel Pentium III	1354	500 MHz	1999
AMD Athlon	3561	1,2 GHz	2000
AMD XP 2400+	5935	2,0 GHz	2002
Pentium 4 Extreme Edition	9726	3,2 GHz	2003
AMD Athlon FX-57	12000	2,8 GHz	2005
AMD Athlon 64 3800+ X2 (Dual Core)	14564	2,2 GHz	2005
AMD Athlon FX-60 (Dual Core)	18938	2,6 GHz	2006
Núcleo 2 X6800 da Intel	27079	2,93 GHz	2006

Tabela 1: Evolução dos processadores (em MIPS)

### **3.1.2 Arquitetura de Software**

.

### **3.1.3 Arquitetura de Rede**

.

### **3.1.4 Arquitetura de Informação**

.

## **3.2 Curva de Complexidade**

.

## 4 *Conclusão*

### 4.1 Avaliação do Modelo

### 4.2 Avaliação da Curva de Complexidade

## *Referências*

GENTLEMAN, W. M. *Dynamic Architecture*: Structuring for change. 2005.

KEARNEY, J. P. et al. Software complexity measurement. *Commun. ACM*, ACM, New York, NY, EUA, v. 29, n. 11, p. 1044–1050, 1986. ISSN 0001-0782.

LLOYD, S. *Black Holes, Demons and the Loss of Coherence*: How complex systems get information, and what they do with it. Tese (Doutorado em Física Teórica) — The Rockefeller University, Nova York, NY, EUA, 1988.

RANGANATHAN, A.; CAMPBELL, R. H. What is the complexity of a distributed computing system? *Complex.*, John Wiley & Sons, Inc., New York, NY, USA, v. 12, n. 6, p. 37–45, 2007. ISSN 1076-2787.

SIMON, H. A. The architecture of complexity. *Proceedings of the American Philosophical Society*, v. 106, n. 6, p. 467–482, 1962.

WEAVER, W. Science and complexity. *American Scientist*, v. 36, p. 536–544, 1948.