

[a4paper,12pt]article [ascii]inputenc [T1]fontenc [english,portuges]babel amsmath,amssymb,amsfonts,textcomp
color calc hyperref colorlinks=true, linkcolor=blue, filecolor=blue, pagecolor=blue, urlcolor=blue graphicx
[1]_1@Standardoddheadevenheadoddfootevenfoot21.001cm29.7cm-1in-1in3cm12pt0cm12pt+0cm29.7cm-3cm-2.49
*-0.018cmblack0.250.018cm*0.101cm Standard document Standard englishSIMULAÇÃO NUMÉRICA
DE SISTEMAS DINÂMICOS

englishGuilherme de Azevedo Silveira
portuges Instituto de Matemática e Estatística – Universidade de São PauloR. do Matão, 1010 – Cidade
Universitária – CEP 05508-090 – São Paulo
english mailto:gas@linux.ime.usp.br gas@linux.ime.usp.br
englishEduardo Colli
portuges Instituto de Matemática e Estatística – Universidade de São PauloR. do Matão, 1010 – Cidade
Universitária – CEP 05508-090 – São Paulo
portuges mailto:colli@ime.usp.br englishcolli@ime.usp.br

englishAbstract. Este artigo descreve o processo de idealização e realização de um software para uso
de alunos na área de matemática aplicada com ênfase em sistemas iterados. Também será tratada a idéia
de ajudar pesquisas que estudam bacias de atração. O mais importante é que o estudante de matemática,
professor ou pesquisador não deve ser obrigado a aprender detalhes de uma linguagem de programação para
executar simulações numéricas clássicas podendo, então, focar em suas habilidades matemáticas. dynamical
systems, java and mathematics, numerical simulations.

englishAbstract. This paper describes the process of brainstorming, developing and using an open
source software in order to help mathematics students dealing with iterated systems. It also aims at helping
researchs which are based on watching those systems iterations and attractors basins, giving the feeling of
what is going on. The main idea is that the math student, teacher or researcher should not need to learn
advanced topics of a programming language in order to make some numeric simulations, so he can focus on
his mathematical analysis and skills. sistemas dinâmicos, java e matemática, simulação numérica.

english1. A necessidade de uma ferramenta

english A medida que crescemos e aprendemos matemática na escola ficamos cada vez mais encantados
com a mágica de analisar equações complexas e de vivenciar as mais belas propriedades numéricas. Tudo isso
muda uma vez que nos tornamos intelectualmente mais adultos e entramos na faculdade, onde encontramos
algumas dificuldades.

english A maior parte dos alunos não lida tão bem com programação, ferramenta que já se provou
extramente útil em diversas áreas da matemática aplicada. Mais grave ainda, isso não acontece somente com
alunos mas também com diversos professores e pesquisadores que têm o mesmo medo de programar.

english A sala de aula seria um ambiente mais descontraído e vivo se os alunos fossem capazes de focar
em suas técnicas matemáticas em vez de ter que aprender os detalhes de shift de bits em C ou qualquer outra
linguagem de programação: não importa se na pesquisa ou na aula, muitas vezes o foco principal é na parte
matemática mas os alunos e professores precisam gastar muito tempo extra para aprender uma linguagem
que talvez não seja o foco do mesmo naquele momento.

english Dessas complicações nasce a necessidade de uma ferramenta que torne opcional o conhecimento
aprofundado de alguma linguagem de programação, sendo que ela ainda seria capaz de simular alguns
sistemas durante a aula, deixando a mesma mais interessante e os exemplos mais práticos, possivelmente
diminuindo o número de alunos que dormem na aula.

english O Iterador (Pulga) é um projeto escrito em Java e desenvolvido com o foco em tornar possível
tudo isso que fora mencionado até esse instante.

english2. Simulação de sistemas dinâmicos

english2.1. O problema

english Um exemplo prático de uso desse software escrito em Java é a simulação numérica de sistemas
dinâmicos contínuos ou discretos através de resultados gráficos.

english Um estudante que deseja ver o que acontece para determinada condição inicial do sistema com
determinados parâmetros pode fazê-lo facilmente sem se preocupar com detalhes de como um gráfico deve
ser mostrado em uma tela como por exemplo técnicas de double buffering etc.

english Outro exemplo é o professor enriquecer uma aula mostrando para alunos não somente na lousa os resultados numéricos de uma iteração mas também graficamente a órbita percorrida, facilitando aos alunos digerir tudo aquilo que fora explicado na aula.

english2.2. A solução

english Com a simples entrada da fórmula de iteração e a escolha do espaço a ser plotado na tela podemos simular a iteração de um sistema.

english Por exemplo, começamos com o atrator de Hénon (com $a = 1.4$ e $b = 0.3$) e configurando $x1$ para $1 - 1.4 * x1 * x1 - 0.3 * x2$ e $x2$ para $x1$. Escolhendo o espaço $-1.5 \leq x1 \leq 1.5$ e $-1.5 \leq x2 \leq 1.5$ temos o resultado da esquerda da Figura 1.

english Figura 1. Attrator de Henon ($a = 1.4$, $b = 0.3$)

center [Warning: Image ignored]