DBGen - Gerador de Dados Sintéticos com Distribuição Fractal

Mônica Ribeiro Porto Ferreira[1], Renato Bueno[1], Caetano Traina Júnior[1]

[1]Instituto de Ciências Matemáticas e de Computação (ICMC) – USP

Av. Trab. Sãocarlense, 400, Centro, Cx. Postal 668, São Carlos, SP 13560-970

monika@grad.icmc.usp.br, {renato | caetano}@icmc.usp.br

Abstract. *When new algorithms are developed to store, retrieve, analyze, mine and present complex data for database applications, their debugging and tuning often require datasets that follows pre-defined properties in order to evaluate the correctness of the implemented programs. It is also important to have these specially prepared data to effectively compare distinct algorithms, so their behavior over different workload can be evaluated. These datasets can be obtained in the real world, or they can be synthetically generated, following rules that enforce desired properties. In this paper we present a new tool to synthesize multi-dimensional data in spaces of any dimensionality, following geometric, statistical or fractal distribution, aiming to provide a rich palette of options for the analyst to drill their algorithms and analysis techniques.*

Resumo. *As tarefas de implementação e ajuste fino para desenvolvimento e configuração de algoritmos, técnicas e aplicativos para dar suporte à armazenagem, recuperação, análise, mineração e visualização de dados complexos em bases de dados freqüentemente requerem a disponibilidade de arquivos e bases de dados de teste, que permitam a avaliação e comparação dos novos algoritmos e aplicativos. Esses arquivos e bases de teste compreendem dados reais e dados sintéticos que são gerados segundo regras que permitam garantir determinadas propriedades nos dados. Neste artigo, é apresentada uma ferramenta capaz de gerar conjuntos de dados sintéticos com qualquer dimensionalidade e com formas de distribuição estatísticas e/ou fractais definidas pelo usuário, para que os algoritmos e aplicativos implementados possam ser aferidos e para que se valide sua resposta em situações de teste controladas.*

1. Introdução

No desenvolvimento de aplicações e em pesquisas na área de Bases de Dados é necessário, muitas vezes, saber como se comportam os dados de uma determinada base, para que se possa avaliar melhor ou até mesmo validar os resultados da aplicação de alguma técnica: o conhecimento prévio da distribuição dos dados em uma base é importante na concepção e nos testes de algoritmos desenvolvidos em trabalhos de pesquisa.

As tarefas de implementação e ajuste fino (*fine tunning*) associadas ao desenvolvimento de coleções de algoritmos, técnicas e aplicativos para dar suporte à armazenagem, recuperação, análise, mineração e visualização de dados complexos em bases de dados (BARIONI, 2002, p.264-269) (TRAINA, 2001, p.184-193) (TRAINA JUNIOR, 2002, p.244-260), freqüentemente, requerem a disponibilidade de arquivos e bases de dados de teste, que permitam a avaliação e comparação dos algoritmos e

aplicativos. Esses arquivos e bases de teste compreendem dados reais, obtidos pela operação dos sistemas em situações reais, e dados sintéticos. Estes últimos são gerados segundo regras que permitam garantir determinadas propriedades nos dados, para que os algoritmos e aplicativos possam ser aferidos e para que se valide sua resposta em situações de teste controladas. Portanto, são necessários bases de dados sintéticos, que obedeçam a distribuições conhecidas, e que possam ter características específicas escolhidas pelo desenvolvedor, de acordo com o propósito de sua utilização.

Para suprir essa necessidade de dados previamente conhecidos (dados sintéticos), foi desenvolvido um aplicativo gerador de dados sintéticos, chamado DBGen. A geração de casos sintéticos é freqüentemente necessária, e as propriedades que se espera serem atendidas pelos dados gerados são muito variadas. Assim, existem inúmeros geradores de dados sintéticos desenvolvidos disponíveis. Uma busca pelo termo "dbgen" no portal google no início de julho de 2005 encontrou 91.200 páginas com essa referência, a grande maioria referindo-se diretamente a ferramentas com o propósito de gerar dados sintéticos.

Este artigo apresenta mais uma ferramenta também chamada DBGen (*Database Generator*), porém com uma funcionalidade não encontrada em nenhuma outra ferramenta de sua classe: enquanto as demais ferramentas se concentram na definição das propriedades dos dados gerados controlando cada atributo (ou coluna) individualmente, a ferramenta aqui descrita inclui a capacidade de gerar dados multidimensionais com distribuição fractal. Esta funcionalidade é importante na avaliação de algoritmos em Bases de Dados, pois muitos conjuntos de dados reais comportam-se como fractais e, com isto, torna-se natural a idéia de aplicar conceitos da Teoria dos Fractais para a análise desses conjuntos (Faloutsos and Kamel, 1994). A grande diferença entre distribuições estatísticas e distribuições fractais é que as primeiras seguem uma regra que se aplica globalmente ao conjunto, enquanto as segundas seguem uma regra que é reaplicada em diferentes escalas dos dados. Casos reais são decorrentes de uma regra que também vale em diferentes escalas, portanto dados gerados segundo regras de distribuição fractal tendem a apresentar um comportamento mais próximo da realidade do que aqueles gerados segundo regras puramente geométricas ou estatísticas.

O DBGen permite ao usuário gerar coleções de dados com qualquer dimensionalidade com formas de distribuição definidas, como em forma de figuras geométricas, estatísticas e fractais.

2. A Ferramenta DBGen

O DBGen tem a finalidade de gerar bases de dados sintéticos, obedecendo a um modelo geométrico, estatístico ou fractal definido para distribuição de tais dados. Os dados gerados, que chamamos de objetos, constituem coleções, primária ou secundária, e são armazenados em memória. O aplicativo permite que duas coleções de objetos estejam simultaneamente armazenadas em memória. São permitidas também operações que atuam nas duas coleções, como a troca e a concatenação, para criação de coleções de dados mais complexas, acumulando os resultados de várias sínteses.

Os objetos armazenados inicialmente em memória podem ser gravados em arquivos de texto em formato csv (comma separated values) ou armazenados diretamente em bases de dados originando as bases de dados sintéticas. Também é possível visualizar graficamente as coleções geradas. A visualização pode ser feita com a própria ferramenta, com o auxílio da ferramenta GNUPLOT ou da ferramenta FastMapDB (TRAINA JUNIOR, 2001, p.95-109).

O usuário pode interagir com o aplicativo por meio de linhas de comandos ou por interface mediante menus. Todos os comandos disponíveis no modo linha-de-comando também estão disponíveis nos menus do aplicativo. Qualquer comando acionado pelo menu tem sua linha-de-comando equivalente escrita no *prompt* do aplicativo, familiarizando o usuário com os comandos. A interface de linhas de comando facilita a utilização de *scripts*, isto é, a utilização de arquivos de texto com as seqüências de comandos que podem ser carregados pelo aplicativo.

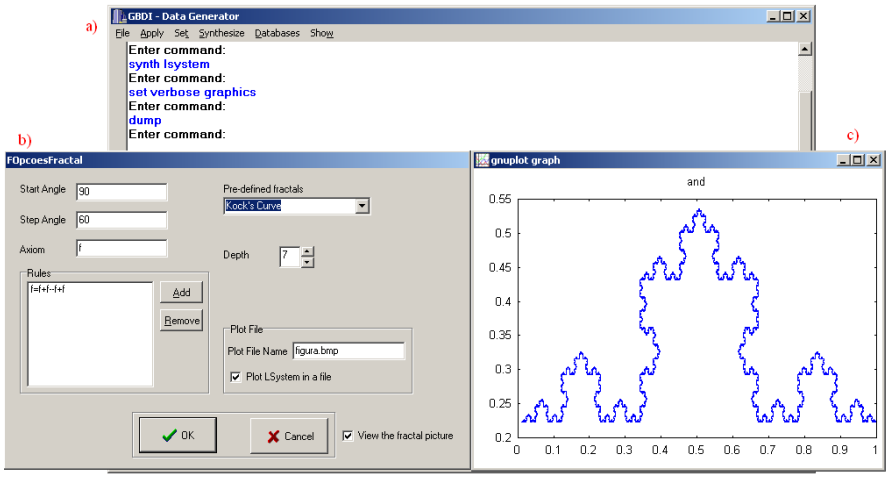

Figura 1: a) Tela principal do DBGen; b) Janela para configuração de fractais e c) Fractal gerado pela ferramenta DBGen.

2.1 "Linguagem Montadora de Conjunto de Dados"

A ferramenta DBGen dispõe de uma linguagem de comando que permite definir valores e regras para geração de valores, transformações lineares sobre o conjunto de dados, e a geração de dados segundo regras pré-definidas. Essa linguagem considera que um conjunto de dados é uma tabela de *N* linhas e *D* colunas e dispõe de comandos para gerar novas tuplas ou definir valores para as colunas, considerando cada coluna como uma dimensão de um espaço *D*-dimensional. Sua filosofia básica é semelhante à linguagem de máquina, tendo sido desenvolvida considerando-se que podem existir simultaneamente até três tabelas, sendo que uma delas, chamada tabela A, atua como um acumulador, e é onde sempre são executados os comandos. Nas outras duas tabelas, que atuam como registradores auxiliares, podem ser executados alguns comandos específicos, como a concatenação com a tabela A. Dessa maneira, novos comandos são facilmente acrescentados à ferramenta, implementados como novos comandos que atuam na tabela A ou em combinação com as outras duas tabelas. Esta filosofia torna a ferramenta poderosa, pois novos comandos podem ser combinados com os já existentes de qualquer maneira que o usuário necessite, aumentando o poder destes comandos.

2.2 Geração dos Dados

A cada comando solicitando síntese de dados (*Synthesize*), uma nova coleção de dados é criada e passa a ser a coleção primária de dados. Se já existia uma coleção primária de dados, esta passa a ser chamada de coleção secundária e a nova coleção, gerada pelo último comando de síntese, passa a ser a coleção primária de dados. Os dados gerados por este comando podem ter qualquer dimensionalidade, podendo gerar bases de dados complexos.
As características das coleções de dados geradas pelo comando de síntese são definidas

pelo usuário mediante a escolha de alguma distribuição das coleções como, por exemplo, em forma de planos, linhas, circunferências, fractais (*L-System*) (FLAKE, 1999) ou nuvens de agrupamentos (Cluster). Na geração das nuvens de agrupamentos (cluster), foi utilizado o algoritmo proposto por CIACCIA (1997, p.426-435). A finalidade deste comando é a geração de dados que apresentem múltiplos agrupamentos, pois isto tem se tornado necessário para a geração de casos de testes para avaliação de algoritmos de análise e indexação de dados em diversos agrupamentos, o que é freqüente em conjunto de dados reais.

Após a criação das coleções de dados, a ferramenta possibilita que o usuário execute operações com as duas coleções (comandos de manipulação de dados) que inclui: definir uma taxa de amostragem a ser aplicada, configurar transformações geométricas (rotação, translação e escala) a ser aplicada nos objetos, realizar a troca ou a concatenação entre as coleções criadas para a criação de coleções de dados mais complexas, etc. O usuário também pode visualizar as coleções presentes em memória, assim como informações sobre tais coleções. Os dados podem ser gravados em arquivos texto ou em bases de dados. Também é possível gerar *scripts* com os comandos utilizados, os quais que podem ser carregados e re-executados posteriormente para a geração de outros dados que seguem as mesmas regras de formação.

2.3 Geração de Fractais

O aplicativo DBGen utiliza a interpretação gráfica de um *L-System* (FLAKE, 1999), conhecida como "gráficos tartaruga" que é um sistema para traduzir uma seqüência de símbolos de entrada em movimentos de um autômato, para gerar um gráfico. O autômato é a "tartaruga". Imagina-se que uma tartaruga está localizada em um plano bidimensional e, dependendo do símbolo de entrada, a tartaruga move-se ou muda de direção. O DBGen armazena os pontos dos vértices do gráfico gerado. Tais pontos são armazenados nos eixos (x, y), escolhidos com o comando set axis.

O axioma, os símbolos de entrada e outros parâmetros podem ser configurados com o comando LSET. As definições dos *L-Systems* de vários fractais conhecidos estão pré-definidas, como as curvas de Koch, Hilbert e Peano. No entanto, a ferramenta permite que o usuário crie sua própria definição de comandos. Adicionalmente, o conceito do *L-System* foi estendido para trabalhar não apenas com duas dimensões, mas com qualquer quantidade de dimensões, gerando "*fractais multidimensionais*". Isso foi feito estendendo a linguagem de comando do *L-System* de maneira que os comandos de alteração de ângulo possam se referir aos ângulos espaciais em qualquer dimensão do conjunto que está sendo gerado.

Usualmente, ângulo é um conceito aplicável a conjuntos de dimensão dois ou três. Para o desenvolvimento desta ferramenta, esse conceito foi generalizado para permitir a definição de ângulo em qualquer número de dimensões, usando a representação matricial de ângulo. Um ângulo em um espaço bi-dimensional (também chamado ângulo plano) é representado por uma matriz da seguinte maneira:

$$\mathrm{R}(\theta) = \begin{bmatrix} \cos\theta & sen\theta \\ -sen\theta & \cos\theta \end{bmatrix}$$

Figura 2: Representação Matricial do Ângulo Plano

Um ângulo em um espaço tri-dimensional (também chamado ângulo sólido) pode ser descrito por dois ângulos planos sendo que, em cada ângulo plano, uma das três coordenadas do espaço fica invariante. Assim, em um espaço onde as coordenadas são

chamadas *x*, *y*, e *z*, um ângulo sólido pode ser descrito, por exemplo, pelo ângulo plano entre as coordenadas *x* e *y*, chamado azimute ou longitude, mais o ângulo plano entre as coordenadas *y* e *z*, chamado ângulo polar ou latitude. Um ângulo em um espaço tri-dimensional é representado por uma matriz da seguinte maneira:

$$R_x(\theta)=\begin{bmatrix}1 & 0 & 0\\ 0 & \cos\theta & sen\theta\\ 0 & -sen\theta & \cos\theta\end{bmatrix}, R_y(\theta)=\begin{bmatrix}\cos\theta & 0 & -sen\theta\\ 0 & 1 & 0\\ sen\theta & 0 & \cos\theta\end{bmatrix} \text{ e } R_z(\theta)=\begin{bmatrix}\cos\theta & sen\theta & 0\\ -sen\theta & \cos\theta & 0\\ 0 & 0 & 1\end{bmatrix}$$

Figura 3: Representação Matricial do Ângulo Sólido

Não existe uma definição para ângulos em espaços de dimensão maior do que três. Assim, generalizamos a definição de ângulos plano e sólido da seguinte maneira: um ângulo em um espaço *N*-dimensional, chamado hiper-ângulo *N*-dimensional, pode ser descrito por *N*-1 ângulos planos, sendo que em cada plano *N*-2 coordenadas do espaço ficam invariantes. Ou seja, cada ângulo plano que descreve um hiper-ângulo *N*-dimensional faz com que apenas duas coordenadas do espaço sejam rotacionadas. Essa definição faz com que um hiper-ângulo *N*-dimensional seja representado por uma matriz da seguinte maneira (considerando neste exemplo que a rotação será feita nas duas últimas coordenadas):

$$R_N(\theta)=\begin{bmatrix}1 & 0 & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & 0 & 0 & 0\\ 0 & 0 & 1 & 0 & 0 & 0\\ \dots & \dots & \dots & \dots & \dots & \dots\\ 0 & 0 & 0 & 0 & \cos\theta & sen\theta\\ 0 & 0 & 0 & 0 & -sen\theta & \cos\theta\end{bmatrix}$$

Figura 4: Representação Matricial do Hiper-Ângulo N-Dimensional

A definição de Hiper-ângulo *N*-dimensional aplicada ao sistema de definição de fractais *L-System* permite a geração de fractais multidimensionais. Esses fractais têm se mostrado de grande utilidade para o teste e validação de algoritmos e técnicas de indexação e mineração de dados. Como a maioria dos dados colhidos em aplicações reais tem comportamento fractal, dados gerados segundo esta regra tendem a apresentar um comportamento mais próximo da realidade do que os dados gerados segundo regras algorítmicas simples ou estatísticas. Um bom exemplo de aplicação de dados sintéticos é a avaliação de algoritmos de indexação para dados complexos, ou algoritmos para mineração de dados. Essas classes de algoritmos são comumente avaliadas de maneira empírica e, portanto dependem da existência de dados adequados para teste. Tais algoritmos também tendem a ser bastante sensíveis as propriedades dos dados. Normalmente, quando se usam dados sintéticos, é porque se pretende explorar facetas dos algoritmos cuja análise dos resultados depende de um conhecimento prévio do comportamento dos dados. Assim, é necessário que seja possível gerar dados de comportamento ao mesmo tempo conhecido e que sejam representativos de situações reais. Dados com distribuição fractal estão entre os mais representativos da realidade, ao mesmo tempo em que a ampla riqueza de definição de regras de geração permite atender a uma grande quantidade de requisitos para teste.

2.4 Dimensão Intrínseca

A dimensão de imersão E de um conjunto de dados é a dimensão do espaço onde o

conjunto está representado, ou seja, o numero de atributos do conjunto de dados. A dimensão intrínseca de um conjunto de dados é a dimensão espacial do objeto representado pelo conjunto de dados, indiferentemente do espaço no qual ele está imerso. Por meio da dimensão intrínseca do conjunto de dados é possível decidir quantos atributos são necessários para caracterizá-lo. Um plano definido em um espaço 20-dimensional continua tendo dimensão intrínseca 2. A dimensão intrínseca pode ser representada por um número fracionário, e também é chamada de dimensão fractal (RAZENTE, 2004, p. 24).

O DBGen possibilita o cálculo aproximado da dimensão fractal dos conjuntos de dados pelo método Box-Counting: para um conjunto de dados imerso em um espaço E-dimensional, o espaço é subdividido em pequenas células, e a dimensão fractal é definida baseando-se na contagem do número de objetos por célula (TRAINA JUNIOR, 2005). O aplicativo também exibe o gráfico relativo ao Box-Counting. Com isso a ferramenta apresentada integra um conjunto poderoso de recursos para a geração de dados sintéticos que emula uma distribuição fractal, que freqüentemente aproxima a distribuição de dados reais mais adequadamente que distribuições estatísticas baseadas nos atributos componentes dos conjuntos de dados.

3. Referências

BARIONI, M. C. et al. Data Visualization in RDBMS. In: Information Systems and Databases, 2002, Tóquio. In proc ISDB, Acta Press, 2002. p.264-269.

CIACCIA, P.; PATELLA, M.; ZEZULA, P.: M-tree"An efficient access method for similarity search in metric spaces. In proc. VLDB, 23, 1997, Athens (GR), 1997.p.426-435.

FALOUTSOS, C.; KAMEL, I.: Beyond Uniformity and Independence: Analysis of R-Trees using the Concept of Fractal Dimension. In proc. 13th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database System. Minneapolis: ACM, 1994. Minneapolis (MN): 1994. p.4-13.

FLAKE, G.W. The Computational Beauty of Nature – Computer Explorations of Fractals, Chaos, Complex Systems and Adaptations. London: The MIT Press, 1999.

RAZENTE, H. Análise visual em processos de redução de dimensionalidade para mineração em sistemas de bases de dados. Dissertação de Mestrado, Maio de 2004, ICMC, USP.

TRAINA, A. J. M. et al. Tri-Plots: Scalable Tools for Multidimensional Data Mining. In proc. ACM SIGKDD-2001. New York: ACM, 2001. San Francisco (CA): 2001. p.184-193.

TRAINA JUNIOR, C.; TRAINA, A. J. M; BARIONI, M. C. N.; BOTELHO, E.; BUENO, R.: Visualização de Dados em Sistemas de Bancos de Dados Relacionais. XVI SBBD, Rio de Janeiro, 2001, p.95-109.

TRAINA JUNIOR, C. et al. Fast indexing and visualization of metric data sets using Slim-trees. IEEE TKDE, Los Alamitos (CA), v.14, n.2, p.244-260, 2002.

TRAINA JUNIOR, C.; SOUZA, E. P. M.; TRAINA, A. J. M; "Using Fractals in Data Mining". in Next Generation of Data Mining Applications, Vol. 1, M. Kantardzic and J. Zurada, Eds. Wiley/IEEE Press, 2005, p. 30p.