XII Workshop de Computação Aplicada -WORCAP 2012

Visualização Analítica de Padrões Espaços-Temporais em Representações Bi-dimensionais

Juliana Paschoal Bueno[1], **Reinaldo Roberto Rosa**[2]

[1]Programa de Mestrado em Computação Aplicada – CAP
Instituto Nacional de Pesquisas Espaciais – INPE

[2]Laboratório Associado de Computação e Matemática Aplicada – LAC
Instituto Nacional de Pesquisas Espaciais – INPE

julianapb@gmail.br, reinaldo@lac.inpe.br

Abstract. *This paper proposes the development of methods and computational tools for analysis of spatial information in the form of matrices that vary over time. The study of spatio-temporal structures is important for the understanding of physical phenomena such as the behavior of the ionosphere, the solar flares or the distribution of galaxies in the universe. Methods and tools such as spatial correlation and GPA (Gradient Pattern Analysis) are applied to matrices of measures of real systems and compared with canonical systems of statistical physics, such as envelopes of sine waves and coupled map lattices with the goal of identifying patterns of systems and nonlinear processes from actual data observed in nature. In this paper, we show the computational implementation of the spatial correlation as well as its application in spatio-temporal data generated from a two-dimensional harmonic model.*

Resumo. *Este trabalho propõe o desenvolvimento de métodos e ferramentas computacionais para análise de informações espaciais, na forma de matrizes, que variam com o tempo. O estudo de estruturas espaço-temporais é importante para o entendimento de fenômenos físicos como o comportamento da ionosfera, de explosões solares ou a distribuição de galáxias no universo. Métodos e ferramentas tais como correlação espacial e GPA (Gradient Pattern Analysis) serão aplicados em matrizes medidas em sistemas reais e comparadas com sistemas canônicos da física estatística, como envelopes de senos e mapas acoplados, com o objetivo de identificar padrões de regimes e processos não-lineares a partir de dados reais observados na natureza. Nesse trabalho, mostramos a implementação computacional da correlação espacial , bem como a sua aplicação em dados espaço-temporais gerados a partir de um modelo harmônico bidimensional.*

Palavras-chave: *envelope de senos, mapa acoplado, correlação espacial*

1. Introdução

Dados em forma de matrizes estão presentes em vários ramos de pesquisa em física espacial, tais como, o estudo da distribuição de galáxias no universo, a formação e evolução de regiões ativas solares e a distribuição de conteúdo eletrônico da ionosfera. Apesar disso, a área de ferramentas computacionais e métodos para análise dessas informações ainda é pouco explorada [Stoyan 2000]. O objetivo do trabalho é explorar essa lacuna através da avaliação de métodos para análise de informações espaciais, na forma de matrizes, que variam com o tempo, combinado-os em uma única metodologia analítica diretamente associada ao processo de visualização. Essa metodologia incorpora de forma inédita o cálculo de coeficientes de correlação espacial no tempo com a análise de padrões de gradiente. Para validação desta metodologia analítica visual, sua aplicação é estudada em sistemas canônicos gerados através de ondas harmônicas, mapas acoplados e equações de amplitude. Finalmente são discutidos os resultados considerando a aplicação desta metodologia na validação de modelos dinâmicos construídos para estudar fenômenos reais observados principalmente nas áreas da física espacial.

2. Geração de Dados Canônicos

Para que seja possível avaliar a qualidade da metodologia para visualização analítica de dados é necessário aplicar os métodos a conjuntos de dados cujo comportamento é bem conhecido. Nas próximas seções serão apresentados dois conjuntos distintos de dados utilizados para verificar o resultado da aplicação da metodologia proposta. Tais conjuntos são modelos simplificados de sistemas físicos.

2.1. Padrões Regulares Harmônicos

O primeiro conjunto de dados apresentado tem como objetivo analisar o uso da metodologia proposta em dados que apresentam padrões de periodicidade. Dessa maneira, foi utilizado neste estudo o modelo de envelope de senos [Freitas 2012], representando ondas periódicas. A função envelope de um sinal é uma curva que tem seus extremos limitados pela amplitude do sinal. Ela pode variar em função do tempo, do espaço, de um ângulo ou de qualquer outra variável que descreva tal sinal.

Envelopes de senos 1-D podem ser representados por:

$$X_t = As_j * \sin(\omega_j * t + \phi_j) + \epsilon_{j,t}$$

Para a equação acima t é o valor do tempo ou do espaço unidimensional variando de 1 a N. O valor de j é número da medida analisada, variando de 1 a n medidas, As_j é o vetor de amplitudes em cada componente j, $\omega = 2\pi f$ é a frequência, e ϕ é o ângulo de fase, n o número de componentes periódicas com j-ésimos parâmetros e $\epsilon_{j,t}$ é o ruído gaussiano de média 0, variância σ^2 e dimensão $1 \times N$.

Envelopes de senos 2-D podem ser representados por:

$$Ms = \sin(B * X_k) + \sin(B * X_k)' + \epsilon_{m,k}$$

Para a equação acima X é uma série multi-periódica definida pela equação anterior de dimensão $1 \times N$, B é um vetor unitário de dimensão $1 \times N$. O valor de k é número da série analisada, variando de 1 a m medidas e m é o número de matrizes periódicas com os k-ésimos elementos. $\epsilon_{m,k}$ é uma matriz de ruídos gaussiano com média μ, variância σ^2 e dimensão $N \times N$.

As figuras abaixo representam envelopes de senos 1-D (Figura 1) e 2-D (Figura 2). Estas figuras foram geradas através da implementação do algoritmo do apêndice A que é uma implementação dos modelos apresentados anteriormente nas equações.

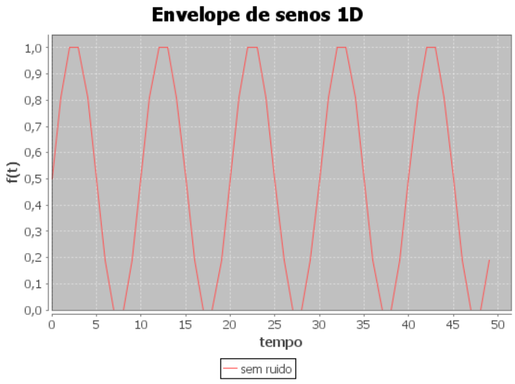

Figura 1. Envelopes de Senos 1-D

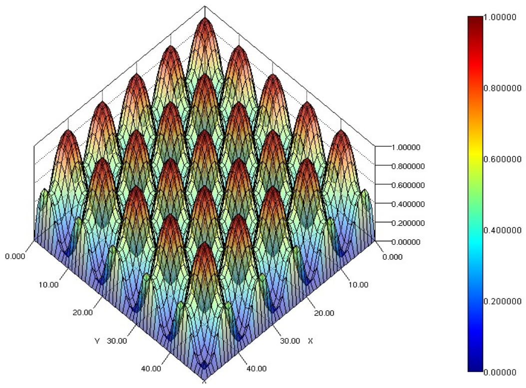

Figura 2. Envelopes de Senos 2-D

2.2. Padrões Irregulares Caóticos

Muitos sistemas físicos não seguem modelos periódicos. Assim, é necessário também avaliar a aplicação da metodologia proposta a modelos que representem tais sistemas. Um desses modelos que será apresentado é a grade de mapas acoplados [Kaneko 1992, Ramos et al. 2000]. Ele é um sistema dinâmico que representa as características de sistemas não-lineares. As grades de mapas acoplados são usadas principalmente no estudo de dinâmica caótica em sistemas multidimensionais. Esse modelo representa dados de forma discreta tanto em tempo quanto espaço. O uso desses modelos ocorre em diversas áreas, desde o estudo do crescimento de cristais [Kessler et al. 1990], o estudo da convecção [Yanagita and Kaneko 1993], ou até mesmo para a modelagem do fenômeno da ebulição [Yanagita 1992].

Mapas acoplados podem ser representados por [Ramos et al. 2000]:

$$x_{n+1} = (1-\epsilon)f(x_n(k)) + \frac{\epsilon}{N}\sum_{t}^{M} f(x_n(l))$$

Para a equação acima n é um passo discreto no tempo e $M = k = l$ é o tamanho da estrutura. Geralmente a função $f(x)$ escolhida é o mapa logístico ($f(x) = 1 - \rho x^2$) por representar um sistema dinâmico simples, mas de comportamento complexo.

3. Análise Computacional de Matrizes Dinâmicas

3.1. Cálculo da Correlação Espacial

As métricas Moran (I) [Moran 1950] e Geary (C) [Geary 1954] são medidas de auto correlação espacial. A primeira foi desenvolvida por Patrick A.P. Moran e a segunda foi desenvolvida pelo Dr. Robert Charles Geary. A auto correlação espacial é caracterizada pela correlação de um sinal entre locais próximos no espaço. A auto correlação espacial é um estudo mais complexo do que o estudo da auto correlação unidimensional, pois a correlação espacial refere-se à analise das relações existentes em duas ou mais dimensões e direções.

A auto correlação espacial de Moran é representada pela equação:

$$I = \frac{N}{\sum_i \sum_j w_{ij}} \frac{\sum_i \sum_j w_{ij}(X_i - \bar{X})(X_j - \bar{X})}{\sum_i (X_i - \bar{X})^2}$$

Para a equação acima N é o número de unidades espaciais indexadas por i e j, X é a variável de interesse, $\bar{X}$ é a média de X e W_{ij} é um elemento da matriz de pesos espaciais.

O valor de I varia de -1 a 1. Nessa métrica, valores negativos indicam auto correlação espacial negativa, enquanto valores positivos indicam o oposto. Nesse caso, o valor mínimo (-1) indica dispersão perfeita, enquanto o valor máximo $(+1)$ equivale à correlação perfeita. O valor 0 representa uma distribuição espacial aleatória.

O valor da métrica de Geary (C) pode variar entre 0 e 2, sendo que o valor 1 indica que há autocorrelação. A auto correlação espacial de Geary é representada pela equação:

$$C = \frac{(N-1) \sum_i \sum_j w_{ij}(X_i - X_j)^2}{2W \sum_i (X_i - \bar{X})^2}$$

Para a equação acima N é o número de unidades espaciais indexadas por i e j, X é a variável de interesse, $\bar{X}$ é a média de X, W_{ij} é um elemento da matriz de pesos espaciais e W é a soma de todos w_{ij}

O resultado C de Geary varia entre 0 e 2. Nesse intervalo, o valor 1 indica que não existe auto correlação espacial. Por outro lado, resultados menores que 1 indicam auto correlação espacial positiva, e valores maiores que 1 referem-se à auto correlação espacial negativa.

4. Resultados

Na figura Figura 3, mostramos a caracterização de envelopes de seno a partir das medidas estatísticas apresentadas na Seção 3.1. Trata-se de um resultado preliminar que será conjugado a medida dos coeficientes de assimetria a serem obtidos a partir da análise dos mesmos dados. Esta parte do trabalho está em desenvolvimento.

5. Conclusões

Nesse trabalho mostramos parte dos objetivos do projeto de mestrado, a qual consistiu no estudo, implementação, teste e aplicação de dois operadores computacionais para o

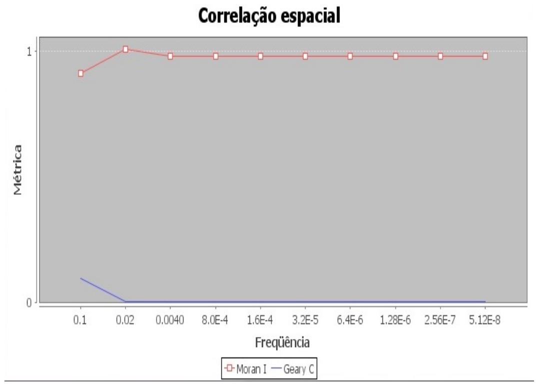

Figura 3. Caracterização de Envelopes de Seno

cálculo da correlação espacial de matrizes canônicas geradas pelo modelo harmônico. Os resultados obtidos confirmam a importância desta pesquisa preliminar como metodologia para análise de dados espaço-temporais. Futuramente, a medida do coeficiente de assimetria de uma matriz será conjugada à correlação espacial resultando em uma nova ferramenta de analítica visual para dados espaço-temporais. Além de outro conjunto de dados canônicos baseado no modelo do mapa acoplado, estudos de caso na área espacial serão definidos.

Referências

Freitas, R. M. (2012). Estudo de métodos comutacionais para visualização e caracterização do uso e cobertura da terra utilizando imagens de sensoriamento remoto. pages 159–162. Instituto Nacional de Pesquisas Espaciais - INPE.

Geary, R. C. (1954). The contiguity ratio and statistical mapping. *The Incorporated Statistician*, (5):115–145.

Kaneko, K. (1992). Overview of coupled map lattices. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 2(3):279–282.

Kessler, D. A., Levine, H., and Reynolds, W. N. (1990). Coupled-map lattice model for crystal growth. *Phys. Rev. A*, 42:6125–6128.

Moran, P. A. P. (1950). Notes on continuous stochastic phenomena. *Biometrika*, 37(1-2):17–23.

Ramos, F. M., Rosa, R. R., Neto, C. R., and Zanandrea, A. (2000). Generalized complex entropic form for gradient pattern analysis of spatio-temporal dynamics. *Physica A: Statistical Mechanics and its Applications*, 283(1-2):171–174.

Stoyan, D. (2000). Preface. In Mecke, K. R. and Stoyan, D., editors, *Statistical Physics and Spatial Statistics. The Art of Analyzing and Modeling Spatial Structures and Pattern Formation*, volume 554 of *Lecture Notes in Physics, Berlin Springer Verlag*, page 3.

Yanagita, T. (1992). Phenomenology of boiling: A coupled map lattice model. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 2(3):343–350.

Yanagita, T. and Kaneko, K. (1993). Coupled map lattice model for convection. *Physics Letters A*, 175(6):415 – 420.

Apêndice

Algoritmo para Cálculo da Correlação Espacial

O algoritmo abaixo é uma implementação da equação de correlação espacial de Moran mostrada na seção 3.1. O código foi desenvolvido através da linguagem de programação JAVA e do paradigma de programação orientado a objetos.

```
public class MoranTechnique {
    public double calculateMetric(){
        //calculando a media dos valores de X
        DoubleMatrix dm = new DoubleMatrix(this.X);
        averageX = dm.calculateAverageMatrix();

        //transforma a matriz X em um arraylist para
        //facilitar o trabalho
        Xarray = dm.transformMatrixInArrayList();

        //monta matriz de adjacentes
        adjacent = dm.generateAdjacentMatrix();

        //calcula cada elemento da matriz menos a media ao
        //quadrado e armazena em uma lista para depois
        //realizar o somatorio
        for(Iterator<Double> it = Xarray.iterator(); it.hasNext();){
            auxiliary.add(Math.pow((it.next() - averageX),2));
        }

        //calcula somatorio
        for(Iterator<Double> it = auxiliary.iterator();
         it.hasNext();){
            sum = sum + it.next();
        }

        //calcula desvio padrão
        standardDeviation = Math.sqrt((sum/(Xarray.size()-1)));

        //calcula a lista de Z's
        for(Iterator<Double> it = Xarray.iterator();
         it.hasNext();){
            Z.add((it.next() - averageX)/standardDeviation);
        }

        //percorrer as linhas da matriz de adjacentes
        //verificando as adjacencias de cada area
        //variavel auxiliar
        Double storedValueZ = 0.0;
        for(int line = 0; line < adjacent.length; line++){
            //percorrer cada linha identificando as colunas
            //que tem valor igual a 1 e recuperar o Z
            //correspondente a coluna
```

```
            for(int column = 0;
             column < adjacent.length;
             column++){

                if(adjacent[line][column] == 1){
                    //somo os valores de Z da linha
                    storedValueZ = Z.get(column) + storedValueZ;
                    sumWij++;
                }
            }
            //multiplico pelo valor de Z da coluna
            WijZiZj.add(Z.get(line) * storedValueZ);
            storedValueZ = 0.0;
        }

        //Calculo o somatorio de todos WijZiZj do arraylist
        for(
        Iterator<Double> it = WijZiZj.iterator();
        it.hasNext();){
            WijZiZjSum = WijZiZjSum + it.next();
        }

        //calculo a metrica final
        I = (Xarray.size() * WijZiZjSum) /
         ((Xarray.size() - 1) * sumWij);

        //retornando a metrica
        return I;
    }
}
```

O algoritmo abaixo apresenta a implementação da equação de correlação espacial de Geary mostrada na seção 3.1. Novamente, o código foi desenvolvido utilizando a linguagem de programação JAVA e o paradigma de programação orientado a objetos.

```
public class GearyTechinique {
    public double calculateMetric(){
        //calcular a media dos elementos de X
        DoubleMatrix dm = new DoubleMatrix(X);
        averageX = dm.calculateAverageMatrix();

        //transforma a matriz X em um arraylist para
        //facilitar o trabalho
        Xarray = dm.transformMatrixInArrayList();

        //monta matriz de adjacentes
        adjacent = dm.generateAdjacentMatrix();

        //calcula cada elemento da matriz menos a media ao
        //quadrado e armazena em uma lista para depois
```

```
        //realizar o somatorio
        for(Iterator<Double> it = Xarray.iterator();
         it.hasNext();){
            auxiliary.add(Math.pow((it.next() - averageX),2));
        }

        //calcula somatorio
        for(
         Iterator<Double> it = auxiliary.iterator();
         it.hasNext();){
            sum = sum + it.next();
        }

        //percorrer as linhas da matriz de adjacentes verificando as
        //adjacencias de cada area
        Double storedValue = 0.0;
        for(int line = 0; line < adjacent.length; line++){
            //percorrer cada linha identificando as colunas
            //que tem valor igual a 1 e recuperar o Xarray
            //correspondente a coluna
            for(int column = 0;
             column < adjacent.length; column++){
                if(adjacent[line][column] == 1){
                    //Calculo Xarray da linha menos o
                    //Xarray da coluna da matriz de adjacentes
                    storedValue =
                     Math.pow(
                     (Xarray.get(line) -
                     Xarray.get(column)), 2) +
                     storedValue;
                    sumWij++;
                }
            }
            WijZiZj.add(storedValue);
            storedValue = 0.0;
        }

        //calcula somatorio de todos WijZiZj
        for(Iterator<Double> it = WijZiZj.iterator();
         it.hasNext();){
            sumWijZiZj = sumWijZiZj + it.next();
        }

        //calcula a metrica C
        C = ((Xarray.size() - 1) * sumWijZiZj) / (2 * sumWij * sum);
        return C;
    }
}
```