Algoritmos de Cross-Matching de Dados Astronômicos

Vinícius P. Freire[1], **Fábio Porto**[2,3],
Ana Maria de C. Moura[2], **José A. F. de Macêdo** [1]

[1]Mestrado e Doutorado em Ciência da Computação (MDCC)
Universidade Federal do Ceará (UFC) – 60455-760 – Fortaleza – CE – Brasil

[2]Extreme Data Laboratory (DEXL Lab)
Laboratório Nacional de Computação Científica (LNCC)
Petrópolis-RJ, Brasil

[3]Laboratório Interinstitucional de e-Astronomia (LIneA)
Rua Gal. José Cristino 77, Rio de Janeiro, RJ - 20921-400, Brasil

{vinipires,jose.macedo}@lia.ufc.br, {fabio,anamoura}@lncc.br

Abstract. *This summary describes the main indexing structures to represent the celestial sphere in databases and a series of cross-matching algorithms used in matching catalogues. At the end of the summary, we present a experiment with one matching algorithm used by the community.*

Resumo. *Este resumo descreve as principais estruturas de indexação para representar a esfera celeste em banco de dados e uma série de algoritmos de cross-matching usados em catálogos correspontes. Ao final do resumo, apresentamos um experimento com um dos algoritmos de matching utilizados pela comunidade.*

1. Introdução

Na área de integração de banco de dados, identificação de entidades (*entity identification*) é o problema de identificar instâncias de objetos de diferentes banco de dados que correspondem à mesma entidade no mundo real. A astronomia possui um problema importante a ser tratado na área de identificação de entidades em grande volume de dados, no qual a posição espacial dos objetos é muito importante: o *cross-matching* de catálogos. Um catálogo de astronomia é um *dataset* que lista uma coleção de objetos e suas características, tais como posição, magnitude e cor. Esses conjuntos de dados astronômicos estão espalhados por todo o mundo, e caracterizam-se por diferentes formatos, esquemas, estruturas de dados, entre outros, uma vez que são tratados por projetos distintos e independentes. Considerando ainda que, ao se capturar dados, existe um pequeno grau de incerteza na posição dos astros, devido aos erros de astrometria. Todas essas características tornam o *cross-matching* de catálogos um grande desafio, sendo algumas delas explanadas ao longo deste artigo.

2. Materiais e Métodos

O primeiro passo para o *cross-matching* de grandes catálogos é ter uma estrutura de dados que represente o céu de forma a facilitar a posição dos astros e seus vizinhos no espaço. Em banco de dados, ela é conhecida como estrutura de indexação e é útil para aumentar a

velocidade de consultas espaciais. Essa estrutura, após implementada, é acessada por um algoritmo de *cross-matching*, que tem por objetivo comparar os objetos presentes em dois catálogos, identificando aqueles objetos que se correspondem. Esta seção trata de uma síntese do que existe na literatura sobre estruturas de indexação e algoritmos de *matching* de catálogos.

2.1. Estruturas de Indexação

A literatura aponta 4 estruturas importantes de indexação (Figura 1): HTM (*Hierarchical Triangular Mesh*)[Kunszt et al. 2001], HEALPix (*Hierarchical Equal Area isoLatitude Pixelisation*) [Gorski], Q3C (*Quad Tree Cube*) [Koposov and Bartunov 2006] e ZONES [Gray et al. 2004]. Na estrutura HTM, a esfera celeste é inicialmente representada por um octaedro. Cada triângulo é então dividido em quatro e o processo é repetido de forma recursiva a uma profundidade pré-determinada. Os triângulos são numerados com o objetivo de preservar a proximidade espacial. Na HEALPix a superfície esférica é subdividida em quadriláteros curvilíneos, de tal modo que cada pixel possua a mesma área. A partição de resolução mais baixa é formada por 12 pixels. A cada subdivisão, todos os pixels são subdivididos em quatro novos pixels de áreas iguais. A estrutura Q3C projeta a esfera celeste em um cubo. Cada posição de um astro é mapeada para um valor binário de 64 bits através do mapeamento curva ordem-Z [Ramakrishnan 2011], acrescentando nos três primeiros bits o número da face que varia de 0 a 5. Já a estrutura ZONES divide a esfera celeste em zonas (anéis de DECs constantes). Em geral são milhares de zonas e cada uma delas tem um número. Os objetos são agrupados então em suas respectivas zonas, de acordo com a sua posição.

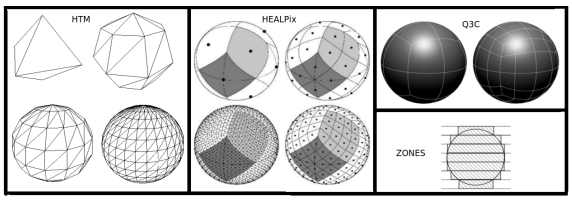

Figura 1. Representação da esfera celeste por estrutura de indexação.

2.2. Algoritmos

A maioria dos algoritmos de *cross-matching* faz a correspondência de dois objetos utilizando somente suas posições, através das coordenadas *ra (right ascension)* e *dec (declination)*. Em [Fu et al. 2012], por exemplo, para tratar o problema de *matching* entre um catálogo com objetos conhecidos e uma nova imagem do céu obtida, os autores assumem que as arestas da imagem são paralelas às direções de *ra* e *dec*. Logo, necessitam encontrar as correspondências para todos os objetos da imagem. Especificamente, para cada objeto p da imagem, procura-se por um objeto q no catálogo tal que: entre todos os objetos na imagem, p é o mais próximo de q; entre todos os objetos do catálogo, q é o mais próximo de p; e a distância entre p e q é no máximo de 1 arcseg (1/3600 grau). Inicialmente, os objetos do catálogo são indexados por faixas, recuperando-se em memória todos os objetos do catálogo que estão na área da imagem estendida em 1 arcseg nos quatro lados. As faixas recuperadas são divididas em subfaixas de 1 arcseg de largura, e os objetos que se encontram dentro das subfaixas são ordenados por *ra*. Para cada objeto da

imagem, o seu objeto correspondente só pode estar distante no máximo de 1 arcseg, e para tanto considera-se apenas os objetos da sua subfaixa e das duas adjacentes. Calcula-se as distancias entre cada um dos objetos das três subfaixas ao objeto da imagem. A condição final para o *match* é: para cada objeto *p* da imagem, assume-se que o seu objeto do catálogo mais próximo é *q*; se o objeto mais próximo de *q* também é *p*, então *q* é o *match* para *p*.

A implementação do Q3C para o postgreSQL possui uma função q3c para realizar o *matching* de dois catálogos. Ao executar a consulta *select * from tabela1, tabela2 where q3c_join*$(tabela1.ra, tabela1.dec, tabela2.ra, tabela2.dec, 0.001)$, determina-se todos os atributos dos objetos equivalentes nas tabelas 1 e 2, correspondentes ao raio de busca 0,001. Assumindo que o *q3c index* foi criado na tabela2, para cada tupla da tabela1, a função q3c_join() é convertida em quatro subconsultas de intervalo correspondentes aos quadrados da pixelização Q3C, projetados para aproximar ao círculo do *cross-matching*. Além disso, uma vez que esses quadrados ocupam uma área maior que o raio do *cross-matching*, deve ser feita uma filtragem adicional com base na distância deste raio. Se objeto da tabela1 estiver dentro desses limites, então o *matching* é feito. Uma validação desse algoritmo de *join* é apresentado na seção 3.

Em [Zhao et al. 2011], os autores apresentam uma função de *matching* paralelizado em grande escala usando a estrutura de indexação HEALPix. O objetivo é diminuir o tempo consumido por operações de I/O. Para isso, o *cross-matching* é dividido em um certo número de sub-tarefas, sendo cada uma responsável por uma pequena região do céu. No entanto, devido a erros de astrometria, se um objeto cai numa região de fronteira na base de dados A, é possível que seu objeto correspondente na base de dados B seja categorizada em uma região vizinha. Este é um problema comum tratado nos algoritmos de *matching*.

Existem algoritmos de *cross-matching* que consideram, além da posição dos objetos, propriedades físicas como classificação estrela-galáxia, magnitude, cor, *redshift* e movimento próprio. Em [Rohde et al. 2005] por exemplo, os autores apresentam os resultados da aplicação de técnicas de *machine learning* automatizada para o problema de *matching* de objetos de diferentes catálogos. O estudo apresenta o *matching* posicional entre dois catálogos, cuja operação atingiu a 44% dos objetos. Porém, ao usar os algoritmos de *machine learning* propostos, considerando ainda outros atributos do objeto, conseguiram um bom desempenho na classificação (99,12% correto).

Com relação ao *matching*, existe ainda o problema da falta de transitividade. Por exemplo, dados três objetos O1, O2 e O3 de catálogos diferentes, o objeto O2 faz *matching* com O1 e O3, mas nem sempre o objeto O1 faz *matching* com O3. Adicionalmente, existe o problema de *matching n-way*, que ocorre quando se deseja fazer o *matching* entre três ou mais catálogos. O fato de se escolher uma ordem diferente de correspondência entre os pares de catálogos pode gerar catálogos resultantes diferentes. Pensando neste problema, os autores em [Budavári and Szalay 2008] formalizaram um algoritmo probabilístico para a identificação de fontes astronômicas correspondentes. O algoritmo é baseado em testes de hipóteses Bayesiana para decidir se uma série de observações realmente pertence a um único objeto astronômico. Classificação morfológica ou medidas do *redshift* aumentam a precisão dos resultados.

3. Resultados

Foram realizados experimentos de *matching* no Q3C para postgreSQL entre o catálogo 2MASS [Skrutskie et al. 2006] (com 470.992.970 de objetos) e o BCC v.05 [Wechsler and et al. (the BCC team) 2013] (com 1.376.582.713), utilizando um raio de busca de 0,001 grau. O processamento da consulta levou 142 segundos e resultou no *matching* de 17.701.306 objetos. No entanto, a falta de conhecimento prévio dos elementos correspondentes entre esses dois catálogos dificultou a avaliação desse *matching*. Para avaliar a confiabilidade do algoritmo de *cross-matching* do Q3C, foi feito um teste com o catálogo 2MASS. Ao executar a função q3c_join() da tabela 2MASS com ela mesma, esperava-se que ela retornasse seus 470.992.970 elementos. No entanto, ao utilizar a função de *join* com raio de 0,001 (valor proposto em [Koposov and Bartunov 2006]), retornaram 483.197.616 *matchings*. Os 12.204.646 (2,6% do total) de *matchings* a mais representam objetos próximos com localizações diferentes, logo são falsos positivos. Porém o número real de *matchings* indicando falsos positivos foi de 6.102.323 (1,3% do total), pois os matchings saíram duplicados ao utilizar duas tabelas iguais. Para explicar esse resultado, assume-se que o índice q3c foi criado na tabela 2, então para cada objeto da tabela 1, a função seleciona os objetos da tabela 2 que estão dentro da área coberta pelo raio de busca. Se, nessa área, existir mais de um objeto da tabela 2, a função interpreta que esses objetos correspondem ao mesmo que está sendo tomado como referência, produzindo resultados a mais.

4. Conclusões Perspectivas

Neste resumo, foi apresentado o estado da arte dos algoritmos de *cross-matching* de objetos astronômicos. Uma das implementações de *matching* mais utilizadas no meio da astronomia, o Q3C, foi escolhida para a realização de uma validação. Como resultado, obteve-se que o algoritmo pode dar como resposta *matchings* equivocados. Apesar desses equívocos representarem apenas 1.3% do total no teste feito na seção 3, a quantidade de *matchings* com ambiguidade foi muito grande, na ordem de milhões de objetos. A questão da ambiguidade dos *matchings* é um dos problemas em aberto na área e que precisa ser explorado.

Referências

Budavári, T. and Szalay, A. S. (2008). Probabilistic cross-identification of astronomical sources. *The Astrophysical Journal*, (1):301.

Fu, B., Fink, E., Gibson, G., and et al. (2012). Fast approximate matching of astronomical objects. *2012 IEEE International Conference on Cluster Computing Workshops*.

Gorski, K. M. Healpix. `http://healpix.jpl.nasa.gov/`. Em 09-04-2013.

Gray, J., Szalay, A. S., Thakar, A. R., and et al. (2004). There goes the neighborhood: Relational algebra for spatial data search. *CoRR*, cs.DB/0408031.

Koposov, S. and Bartunov, O. (2006). Q3c , quad tree cube â the new sky-indexing concept for huge astronomical catalogues and its realization for main astronomical queries (cone search and xmatch) in open source database postgresql. *The Astronomical Data Analysis Software and Systems (ADASS) conference*, 351:735–738.

Kunszt, P. Z., Szalay, A. S., and Thakar, A. R. (2001). The hierarchical triangular mesh. In Banday, A. J., Zaroubi, S., and Bartelmann, M., editors, *Mining the Sky*, page 631.

Ramakrishnan, R. (2011). Indexação baseada em curvas de preenchimento de espaço. In *Sistemas de Gerenciamento de Banco de Dados*, pages 809–811. McGraw-Hill.

Rohde, D. J., Drinkwater, M. J., Gallagher, M. R., and et al. (2005). Applying machine learning to catalogue matching in astrophysics. *Monthly Notices of the Royal Astronomical Society*, 360(1):69–75.

Skrutskie, M., Cutri, R. M., Stiening, R., Weinberg, M. D., and et al. (2006). The two micron all sky survey (2mass). *AJ*, 131(2):1163–1183.

Wechsler, R. H. and et al. (the BCC team) (2013). Catalog simulation provided by des collaboration "blind cosmology chanlenge". `http://www.slac.stanford.edu/~risa/des_bcc/`.

Zhao, Q., Sun, J., Yu, C., and et al. (2011). Improved parallel processing function for high-performance large-scale astronomical cross-matching. *Transactions of Tianjin University*, 17(1):62–67.