5.1 Base Teórica

Visando oferecer suporte ao desenvolvimento dos algoritmos de fragmentação horizontal propostos nesta tese, esta seção revisa, inicialmente, dois conceitos já discutidos no capítulo 2. O primeiro conceito, descrito sucintamente na seção 5.1.1, é o conceito de hierarquia de relacionamento de atributos. Em especial, esse conceito foi introduzido detalhadamente na seção 2.4.1.1, sendo sua revisão indispensável a um melhor entendimento dos algoritmos. Já o segundo conceito revisado neste capítulo, abordado formalmente na seção 5.1.2, é o conceito de grafo de derivação. Como discutido resumidamente na seção 2.5.1.1, um grafo de derivação consiste em uma representação apropriada para um *lattice* de visões.

5.1.1 Hierarquia de Relacionamento de Atributos

Os atributos de uma dimensão podem possuir um relacionamento com outros atributos da mesma dimensão através de hierarquias de relacionamento de atributos. Essas hierarquias de relacionamento especificam níveis de agregação e, conseqüentemente, granularidade dos itens de dados. Um exemplo de hierarquia de relacionamento de atributos para a dimensão *filial* é:

- filial → cidade → estado → região → país.

De acordo com essa hierarquia, *filial* possui menor granularidade do que *cidade*, que por sua vez possui menor granularidade do que *estado*, e assim por diante. Nesse exemplo, o atributo *país* possui a maior granularidade. A existência de mais do que uma hierarquia de relacionamento de atributos para uma mesma dimensão é totalmente plausível. Como exemplo, pode-se citar as seguintes hierarquias para a dimensão *tempo*:

- dia → mês → trimestre → semestre → ano; e
- dia → mês → quadrimestre → ano.

5.1.2 Grafo de Derivação

A seção 5.1.2.1 define formalmente o grafo de derivação, enquanto que a seção 5.1.2.2 apresenta exemplos que ilustram os conceitos discutidos.

5.1.2.1 Conceito de Grafo de Derivação

Um grafo orientado é um par (V,E) de conjuntos disjuntos de vértices V e arestas E, juntamente com dois mapeamentos: inic: E → V e term: E → V. Estes mapeamentos associam a cada aresta e um vértice inicial inic(e) e um vértice terminal term(e), de forma que e é direcionada de inic(e) para term(e). Um grafo orientado não possui ciclos, ou seja, inic(e) ≠ term(e). Adicionalmente, um grafo orientado não possui arestas múltiplas. Isto significa que existe somente uma única aresta entre os mesmos dois vértices do grafo [Die97].

Um *lattice* de visões (ou *lattice* de visões agregadas ou, resumidamente, *lattice* de agregações) é definido pelas seguintes propriedades [HRU96, BPT97, GHRU97, SDN98]:

- existe uma ordenação parcial $\preceq$ entre as agregações no *lattice*. Para as visões agregadas u e v, v $\preceq$ u se e somente se v pode ser determinada usando somente os resultados de u. Em outras palavras, a agregação v é dependente da agregação u;

- existe uma visão topo, da qual cada outra visão agregada é dependente. Mais especificamente, esta visão é utilizada para derivar qualquer outra visão no *lattice*. Nesta tese, a visão topo é denominada agregação derivante total; e
- existe uma visão completamente agregada vazio que pode ser calculada a partir de qualquer outra visão no *lattice*. A visão vazio representa a visão completamente agregada.

Adicionalmente, algumas funções podem ser definidas para os elementos do *lattice*. Dado três agregações v, w e u, ancestrais, descendentes, ancestrais diretos e descendentes diretos de v são definidos como:

$$\text{ancestrais}(v) = \{\, w \mid v \preceq w \,\}\ . \tag{5.1}$$

$$\text{descendentes}(v) = \{\, w \mid w \preceq v \,\}\ . \tag{5.2}$$

$$\text{ancestrais\_diretos}(v) = \text{pais}(v) = \{\, w \mid v \prec w,\ \nexists\, u,\ v \prec u,\ u \prec w \,\}\ . \tag{5.3}$$

$$\text{descendentes\_diretos}(v) = \text{filhos}(v) = \{\, w \mid w \prec v,\ \nexists\, u,\ w \prec u,\ u \prec v \,\}\ . \tag{5.4}$$

Nas Equações 5.3 e 5.4, a relação existente entre os símbolos $\prec$ e $\preceq$ é representada por:

$$v \prec w \;\Rightarrow\; v \preceq w \;\wedge\; v \neq w\ . \tag{5.5}$$

Uma vez definidos os conceitos de grafo orientado e de *lattice* de visões, o conceito de grafo de derivação pode ser apresentado. Um grafo de derivação G consiste em um grafo orientado no qual V(G) representa um conjunto de visões agregadas (medidas numéricas agregadas), ao passo que E(G) representa um conjunto de relações de dependência $\preceq$ entre essas visões agregadas. Assim, se a aresta $e \in G$ representa uma relação de dependência entre dois vértices v e $u \in G$ de forma que $v \preceq u$, então inic(e) = u e term(e) = v.

5.1.2.2 Exemplos Relacionados ao Grafo de Derivação

Visando exemplificar os conceitos discutidos na seção 5.1.2.1, a Figura 5.1 ilustra dois grafos de derivação. O primeiro deles, exibido na Figura 5.1a , é o mesmo grafo de derivação representado na Figura 2.9, para as dimensões produto (p), filial (f) e tempo (t). Já a Figura 5.1b exibe o grafo de derivação em termos das dimensões e das suas respectivas hierarquias de relacionamento de atributos. Nessa figura, são consideradas: (i) as dimensões produto (p) e filial (f); e (ii) as hierarquias de relacionamento de atributos produto (p) → marca (m); filial (f) → cidade (c) → estado (e).

Como discutido na seção 2.5.1.1, cada agregação representada por cada vértice do grafo de derivação agrega medidas numéricas sobre as dimensões presentes naquele vértice, e é nomeada de acordo com essas dimensões. Quando são considerados os atributos das dimensões, então os vértices também são nomeados de acordo com essas dimensões, considerando no entanto a granularidade de seus atributos. Para uma aplicação de *data warehousing* de uma cadeia de supermercados, a qual analisa a medida numérica *vendas* de acordo com as dimensões presentes em cada uma das Figuras 5.1a e 5.1b, o vértice pf em ambas as figuras representa a visão multidimensional vendas por produto por filial. O vértice me na Figura 5.1b, por sua vez, representa a visão multidimensional vendas por marca por estado.

Já as dependências existentes entre agregações adjacentes são representadas através de arestas direcionadas. O grafo de derivação apenas ilustra as dependências entre as agregações adjacentes, e não as suas cláusulas transitivas. Na Figura 5.1a, o vértice p pode ser computado a partir do vértice pf, ou seja, inic(pf_p) = pf e term(pf_p) = p. Mais detalhadamente, para o vértice p, as funções definidas nas Equações 5.1 a 5.4 podem ser instanciadas respectivamente por:

- ancestrais(p) = {p, pf, pt, pft};
- descendentes(p) = {p, vazio};
- ancestrais_diretos(p) = {pf, pt}; e
- descendentes_diretos(p) = {vazio}.

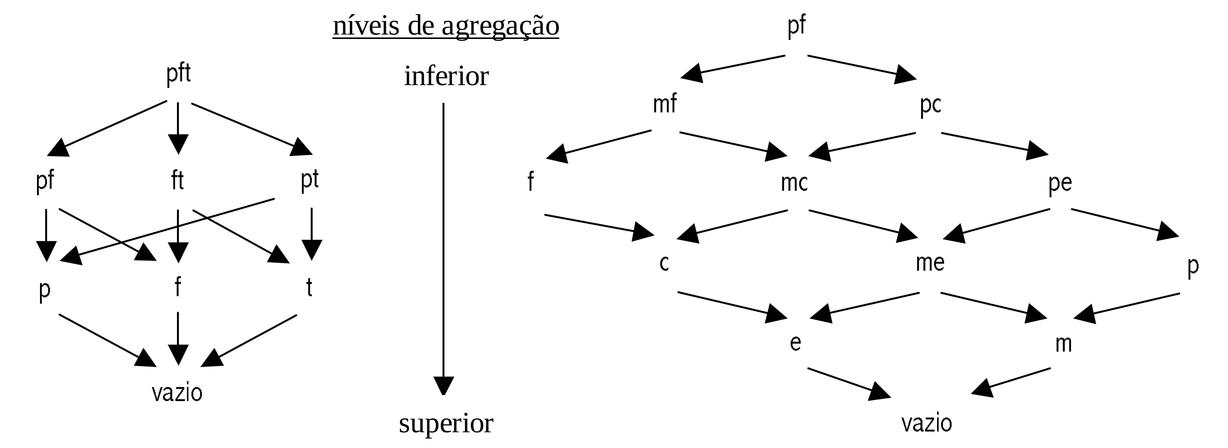

Figura 5.1a Dimensões produto (p), filial (f) e tempo (t).

Figura 5.1b Dimensões produto (p) e filial (f), e p → marca (m); f → cidade (c) → estado (e)

Figura 5.1 Exemplos de grafos de derivação.

Por fim, na Figura 5.1 também é exemplificada a relação existente entre visões agregadas e níveis de agregação. Dentro deste contexto, a visão pft na Figura 5.1a corresponde ao nível inferior da hierarquia de agregação. A mesma observação é válida para a visão pf da Figura 5.1b. Por outro lado, a visão completamente agregada vazio corresponde ao nível mais superior da hierarquia de agregação.

5.2 Fragmentação Horizontal Unidimensional para Grafos de Derivação que Representam apenas Dimensões (FHU–D)

Esta seção apresenta o algoritmo FHU–D. Este algoritmo é caracterizado por ser o algoritmo básico proposto nesta tese, uma vez que é voltado a grafos de derivação que representam apenas dimensões e que realiza a fragmentação horizontal dos dados do *data warehouse* em termos de apenas uma única dimensão. Entretanto, o algoritmo FHU–D permite que várias restrições sejam aplicadas a esta dimensão.

5.2.1 Entradas para o Algoritmo FHU–D

As entradas requeridas pelo algoritmo FHU–D são:

- o esquema do *data warehouse*, representado através de um grafo de derivação G;