

2 Modelos de Multi-resolução

Os objetivos deste capítulo são (a) apresentar o problema de multi-resolução (**MR**) de entidades geométricas representadas por modelos poligonais/poliedrais, tais como as malhas de simulação em RNPs, e (b) caracterizar dois tipos de modelos de MR (**MMRs**): os hierárquicos e os reticulares.

Os MMRs tratados aqui são denominados por Floriani [Flo97] e Puppo [Pup97] **MMR Geométricos**, em contraposição a outras abordagens existentes para a MR como, por exemplo, a de espaço de escala [Lin94], baseada na teoria dos sinais. Além disso, o enfoque principal é dado à MR sob o ponto de vista de uma técnica de aceleração.

É necessário explicitar que este trabalho faz uso de conceitos relacionados à representação paramétrica de objetos, definida por uma função $\mathbf{o}: U \subset \mathbb{R}^n \rightarrow \mathbb{R}^m$, onde U é chamado de **domínio do objeto** ou **espaço paramétrico**, e \mathbb{R}^m é chamado de **espaço geométrico do objeto**. Por exemplo, algumas superfícies imersas no \mathbb{R}^3 podem ser representadas pela função $\mathbf{o}: U \subset \mathbb{R}^2 \rightarrow \mathbb{R}^3$ (Figura 2.1). Dessa forma, os termos **domínio** e **modelo** são utilizados para representar, respectivamente, o objeto no espaço paramétrico e no espaço geométrico.

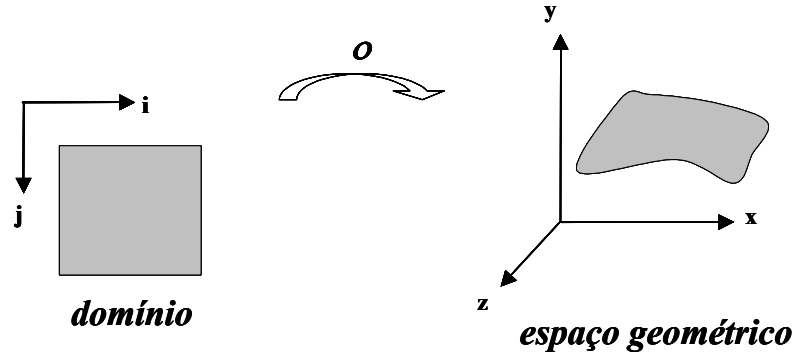


Figura 2.1 – Superfície paramétrica.

2.1 Multi-resolução

Um MMR é um mecanismo que suporta a representação e o processamento de entidades geométricas em diferentes níveis de detalhe [Pup97]. Garland [Gar99] formalizou o conceito de MMR através da família de modelos

$$\mathbf{M}: \mathbf{C} \rightarrow \mu$$

onde \mathbf{C} representa os contextos de visualização (parâmetros de câmera e descrição de cena, por exemplo) e μ é o espaço não-enumerável de todos os modelos existentes que representam um mesmo objeto. Assim, $\mathbf{M}(\mathbf{v}) \in \mu$ representa o modelo do objeto que é adequado ao contexto $\mathbf{v} \in \mathbf{C}$.

Devido à grande afinidade entre os problemas de MR e os de simplificação poligonal³ e com base nos resultados de Suri e Agarwal [Aga94], que provaram que o problema de simplificação poligonal baseado na norma de erro L_∞ para campos escalares é NP-Hard, estima-se que achar $\mathbf{M}(\mathbf{v})$ de forma ótima seja também um problema NP-Hard [Gar99].

Para contornar este problema, geralmente utilizam-se heurísticas que sejam capazes de selecionar um subconjunto finito $\mu' \subset \mu$, tal que μ' seja satisfatório em todos os

³ Dado um espaço μ de modelos que representam um mesmo objeto e um modelo de referência $\Sigma_0 \in \mu$, o problema de simplificação poligonal pode ser assim definido: achar o modelo $\Sigma \in \mu$ com o menor número possível de células, tal que $\varepsilon(\Sigma_0, \Sigma) < \Delta$, onde ε é a função que define o erro entre dois modelos e Δ é o limite máximo permitido para o erro [Math].

contextos de visualização. Esse processo será denominado, neste trabalho, **construção do espaço discreto de modelos** e é comumente feito em pré-processamento.

A construção, a armazenagem e a codificação de μ' devem ser tais que permitam, preferencialmente, dentre outras coisas:

- uma construção eficiente do espaço discreto de modelos;
- uma representação compacta da estrutura de dados – geralmente, considera-se razoável que a quantidade de memória utilizada seja um pouco maior (em torno do dobro) do que a quantidade de memória necessária para armazenar unicamente o modelo de maior resolução [Gar99]; e
- um mecanismo eficiente de extração, em tempo de exibição, de modelos adaptativos dependentes do contexto de visualização.

A principal idéia proposta na literatura para caracterizar um MMR é definir o espaço discreto de modelos μ' a partir de um esquema de decomposição do domínio no qual a resolução de cada modelo é controlada pelo grau de refinamento da decomposição [Flo97] (Figura 2.2). Ou seja, o número de células da decomposição de cada modelo, de forma geral, diz qual é a sua resolução.

Esse esquema é apropriado para modelos dados por decomposições do espaço e, particularmente, para outros tipos de decomposições não contínuas (como as malhas de simulação em reservatórios), nas quais cada célula está associada a uma porção do domínio do objeto, formando a unidade de fragmentação da decomposição.

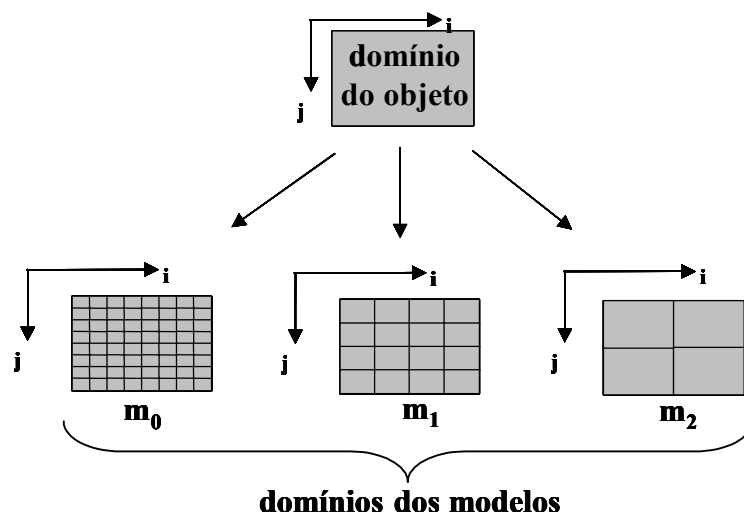


Figura 2.2 – Esquema de decomposição do domínio.

É importante para o MMR dispor de uma quantidade de fragmentos que lhe ofereça o **poder de expressão**⁴ necessário. Dado o modelo inicial de um objeto, com a quantidade de fragmentos suficiente para gerar uma única representação desse objeto, é tarefa do MMR gerar novos fragmentos e estruturá-los adequadamente. Este processo corresponde exatamente à construção do espaço discreto de modelos.

Os diversos algoritmos existentes para resolver este problema são baseados, de uma forma ou de outra, em dois tipos de alterações [Pup97, Gar98]: **alterações locais**, que atuam sobre apenas uma parte do domínio (pequeno grupo de células), e **alterações globais**, que atuam sobre todo o domínio (todas, ou quase todas, as células do modelo).

A utilização extrema da idéia de alterações globais cria MMRs caracterizados pela existência de um número fixo, normalmente pequeno, de representações distintas de um mesmo objeto, sendo classificados como **discretos**. Por outro lado, o uso de alterações locais proporciona a criação de MMRs caracterizados pela existência de um número elevado, virtualmente infinito, de representações distintas, sendo classificados como **contínuos**.

⁴ O poder de expressão de um MMR é igual à cardinalidade de μ' [Velho00].

Neste contexto, existem dois MMRs que abrangem a maioria dos algoritmos de MR propostos na literatura, encerrando em si importantes problemas e conceitos da área: **modelos hierárquicos** e **modelos reticulares**⁵ [Flo97].

2.2 Modelos de Multi-resolução Hierárquicos

Consideremos um **modelo base** Σ_0 que seja a aproximação inicial de um objeto. Os modelos de multi-resolução hierárquicos (**MMRHs**) são construídos a partir da decomposição gerada pelo modelo base através de **regras de modificação** que definem padrões para alterar cada célula da decomposição, modificando sua conectividade e inferindo a nova geometria. Neste sentido, cada célula $c_i \in \Sigma_0$ é vista como uma entidade independente que abrange um domínio $dom(c_i)$.

As regras de modificação predizem a forma com que as células são modificadas através de **padrões de modificação**. Um padrão pode ser de simplificação (cuja aplicação, na prática, só é possível em alguns casos bem definidos, como o do algoritmo *mipmapping* [Wil83]) ou o de refinamento. Sem perda de generalidade, consideremos apenas o caso do refinamento, no qual Σ_0 é o modelo menos refinado. Um padrão de refinamento é aplicado à célula c_i , substituindo-a por um novo conjunto de células que formam o modelo Σ_j , que abrange exatamente o domínio $dom(c_i)$ (Figura 2.3). Este padrão é aplicado novamente a cada célula de Σ_j e o processo é repetido recursivamente até que o domínio $dom(c_i)$ esteja representado na resolução máxima necessária para a aplicação, estando próximo o suficiente do objeto segundo a norma de erro utilizada.

⁵ Originalmente, Floriani [Flo97] referencia o modelo reticular através do termo *modelo piramidal* (*pyramidal model*). O termo *reticular* é, aqui, utilizado por motivos de clareza e preferência pessoal. Outras denominações existentes para os modelos hierárquicos e reticulares são, respectivamente, *modelos representados por árvores* e *modelos baseados em históricos* [Pup97].

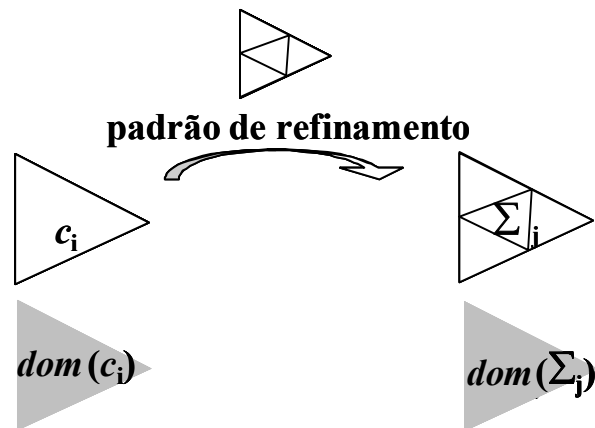


Figura 2.3 - Padrão de refinamento conhecido como triangulação quaternária.

A estrutura natural para representar o processo de refinamento é uma árvore. Os nós da árvore são as células de entrada ou saída do padrão de modificação e cada arco liga uma célula à decomposição que a substituiu (Figura 2.4). A Figura 2.5 ilustra uma representação alternativa do processo conhecida como árvore de segmentação. Sua construção pode ser feita da forma *top-down* ou *bottom-up*, dependendo se o padrão é de refinamento ou de simplificação, respectivamente. Percorrendo a árvore, pode-se escolher o grau de refinamento desejado para o modelo através de um corte transversal.

Cada nível da árvore de segmentação gera um modelo que abrange todo o domínio com a mesma resolução. Cortes que possuam nós em resoluções diferentes abrangerão o domínio adaptativamente, ou seja, em resoluções diferentes ao longo do domínio.

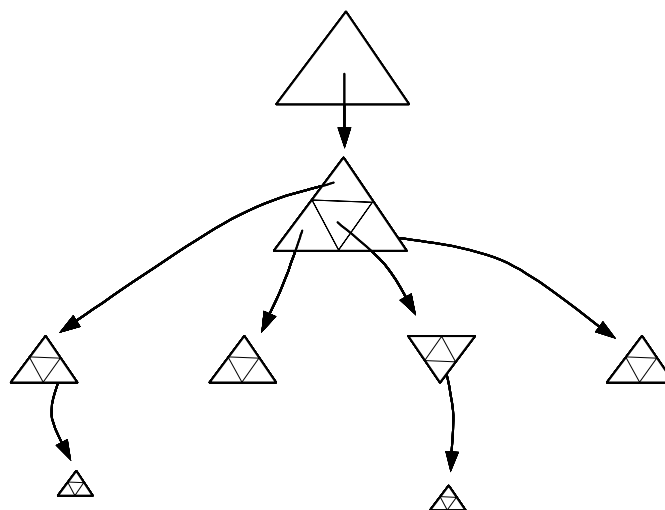


Figura 2.4 - Árvore que representa o processo de refinamento.

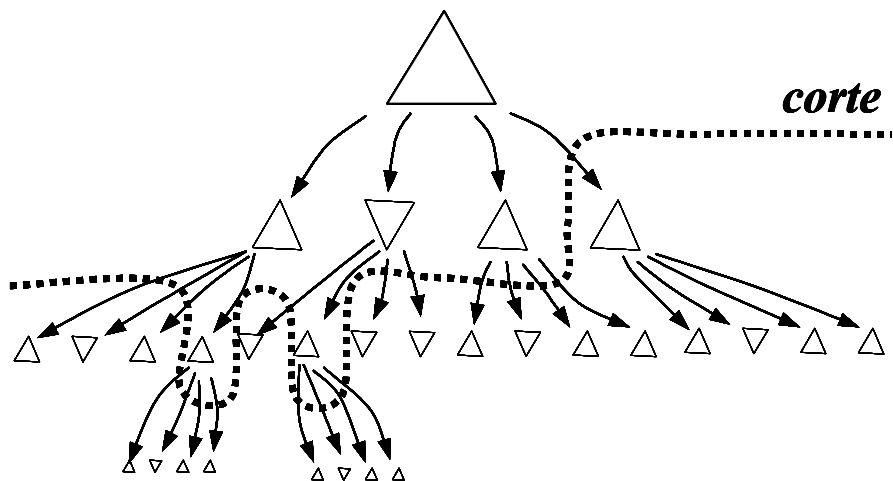


Figura 2.5 – Corte transversal na árvore.

O padrão de modificação é fundamental para a qualidade dos modelos gerados, principalmente quando a aplicação exige que os modelos sejam conformes. Neste caso, um padrão qualquer pode gerar modelos com descontinuidades entre as células, causadas pela formação de vértices T (Figura 2.6-a), principalmente entre células vizinhas de um mesmo corte que estão em níveis diferentes. Em superfícies imersas no \mathfrak{R}^3 , as descontinuidades são evidenciadas pela presença de aberturas (*cracks*) (Figura 2.6-b).

Os modelos que possuem descontinuidades geradas por vértices T são não-conformes. A maioria das aplicações deve evitar a existência dessas descontinuidades. Para tanto, é necessário utilizar padrões de modificação adequados e criar regras que conduzam a escolha dos cortes a serem feitos durante o caminhamento na árvore. Neste sentido, as principais soluções propostas são árvores quaternárias com restrições [Her87], triangulações adaptativas [Flo95, Sca92], hierarquia de triângulos direitos [Heb95] e hierarquia de triângulos do tipo 4-8 [Vel00]. Elas variam quanto ao tipo de célula utilizada, ao padrão de modificação empregado, ao número de padrões disponíveis e às regras de caminhamento na hierarquia para gerar cortes.

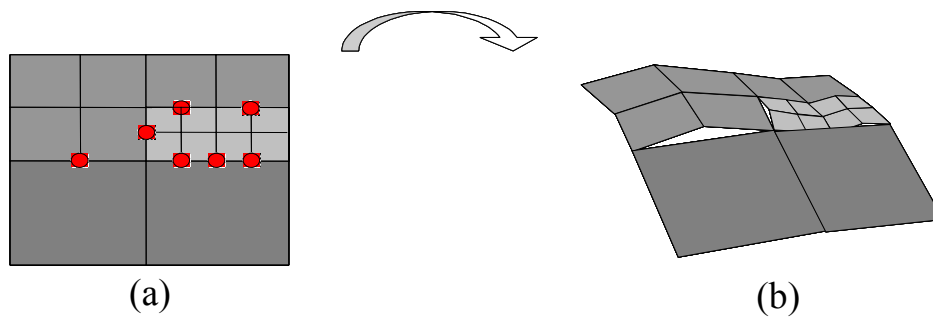


Figura 2.6 - Aberturas geradas por vértices T, em destaque.

Em geral, os MMRHs possuem boas propriedades, como indexação espacial [Pup97], agrupamento com coerência espacial (*clustering*) [Gar98] e simplicidade de implementação e entendimento.

Por outro lado, a geração do modelo base pode não ser trivial no caso de refinamento. Além disso, a presença de aberturas é um problema sério, cujas soluções normalmente possuem algum tipo de efeito colateral indesejado, como:

- a geração de modelos com células com uma péssima razão de aspecto, fato ressaltado por Floriani [Flo84] que se deve à tentativa de manter intactas as arestas das células ao longo de diferentes níveis;
- a redução do poder de expressão da hierarquia (ex. [Her87]), uma vez que os cortes válidos são restringidos; e
- o aumento do tempo de extração de um modelo, devido a várias passadas na hierarquia para tornar o modelo conforme (ex. [Her87]).

As soluções mais promissoras parecem estar relacionadas à hierarquia de triângulos do tipo 4-8 tratada por Velho [Vel00, Vel01, Vel01b]. Entretanto, ainda não é claro como esta solução pode ser aplicada a malhas não-conformes.

2.3 Modelos de Multi-resolução Reticulares

Ao contrário dos modelos hierárquicos, os modelos de multi-resolução reticulares (**MMRRs**) não são dados por padrões de modificação fixos aplicados a cada célula individualmente. Dado um modelo base Σ_0 , qualquer sub-modelo $\kappa_i \subseteq \Sigma_0$ pode sofrer uma transformação T gerando um novo sub-modelo κ_j , mais ou menos refinado, (Figura 2.7) que abrange o mesmo domínio de κ_i ($dom(\kappa_j) = dom(\kappa_i)$):

$$T(\kappa_i) = \kappa_j.$$

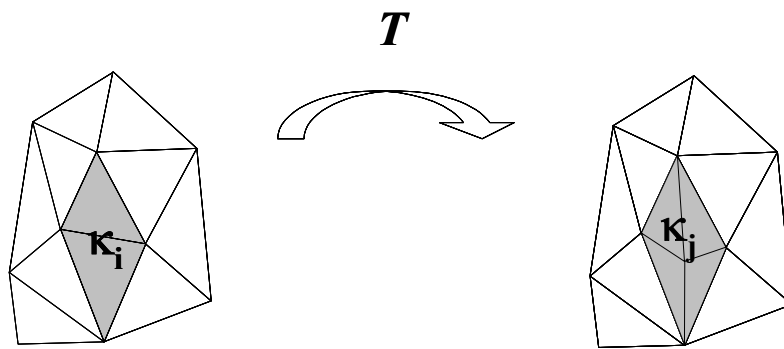


Figura 2.7 - Transformação de decomposição aplicada a um domínio.

A idéia é que uma série dessas transformações $\{T_1(\kappa_1), T_2(\kappa_2), \dots, T_n(\kappa_n)\}$, cuja união dos domínios pode potencialmente abranger todo o domínio $dom(\Sigma_0)$, pode ocorrer em paralelo desde que os domínios sejam disjuntos entre si (ou seja, os domínios não interfiram uns nos outros). Após a aplicação de todas as transformações, o modelo troca globalmente de resolução, gerando um novo modelo Σ_1 . O processo é repetido para o modelo Σ_1 e assim sucessivamente, até que o modelo com a resolução desejada seja alcançado.

Pode-se ilustrar o processo de modificação através da Figura 2.8. Para cada célula do modelo Σ_i , existem arcos direcionados ligando-a às células do complexo Σ_{i+1} cujos domínios interceptam o domínio da célula em questão. Percebe-se que as células que permanecem inalteradas nas duas malhas formam arcos ditos **triviais**. Além disso, uma célula $c_k \in \Sigma_{i+1}$ pode ter interseção com mais de uma célula de Σ_i , recebendo

mais de um arco. Este é o motivo que, ao contrário dos MMRHs, impede que este modelo seja representado por uma árvore.

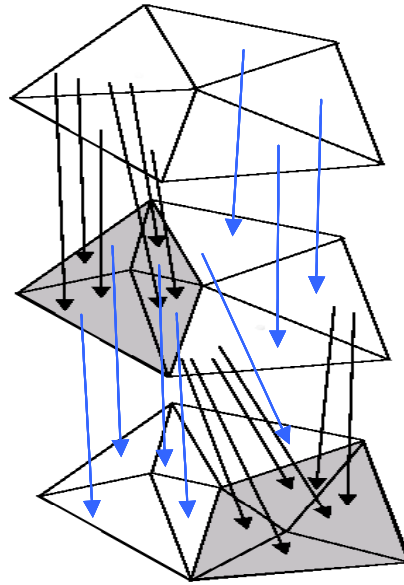


Figura 2.8 - Dependências entre domínios. As setas azuis representam dependências óbvias e são denominadas arcos triviais.

Uma condição importante é que as transformações T devem ser **operações locais**, ou seja, que alteram a conectividade e a geometria do modelo apenas localmente. Normalmente, tais modificações são baseadas em **operadores** que determinam como a modificação será realizada. Um operador aumenta ou diminui a complexidade do modelo gradualmente, podendo ser um operador de refinamento ou de simplificação, respectivamente. Os operadores mais comuns são: contração de arestas, contração de vértices, expansão de vértices, remoção de vértices, inserção de vértices, contração de células e expansão de células (Figura 2.9). Observando a Figura 2.9, percebe-se que a conectividade da borda da operação não é alterada. A exceção a esta regra são as bordas da operação que coincidem com as bordas do objeto, que podem ser alteradas.

É necessário enfatizar que a maioria dos padrões de modificação utilizados em MMRHs não são operadores e não são locais, de modo que eles normalmente alteram a conectividade e a topologia do modelo além do domínio no qual são aplicados.

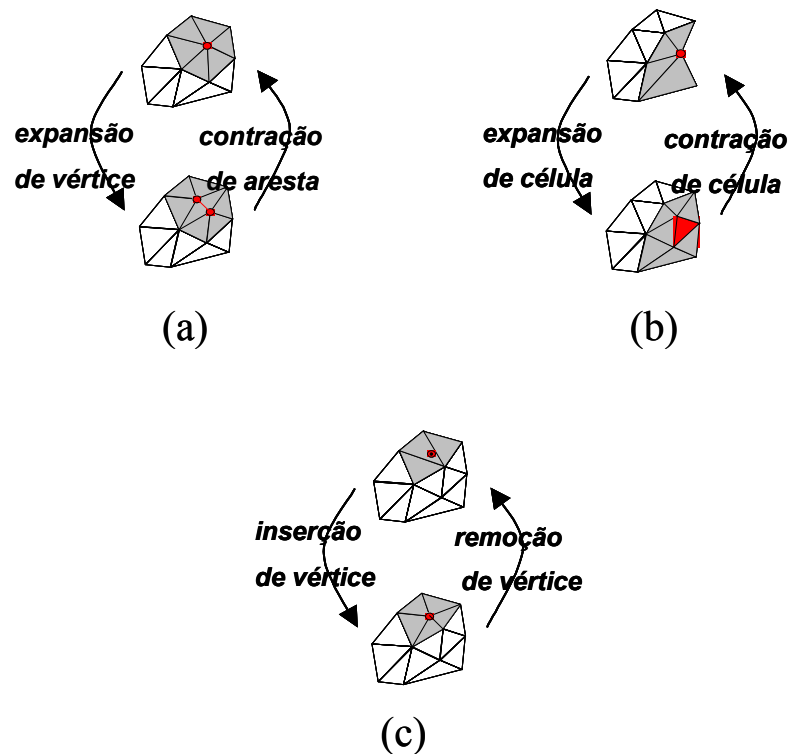


Figura 2.9 – Operações locais.

Os algoritmos que implementam operadores também são chamados de **incrementais** porque são capazes de criar um **histórico** cronológico de operações locais que alteram monotonicamente e gradualmente a complexidade do modelo. Como as operações podem ser tanto de refinamento quanto de simplificação, os históricos podem levar do modelo mais refinado ao mais simplificado ou o contrário.

Para se obter maior compactação e ter um tempo de extração proporcional ao tamanho do modelo extraído, é necessário utilizar históricos que contenham operações de refinamento. Um bom exemplo desta estratégia é o trabalho de Hoppe, que apresenta o conceito de malhas progressivas [Hop96].

Entretanto, apenas o uso de históricos não é suficiente para representar todo o modelo reticular. Para representar inteiramente um MMRR são necessários o histórico e as relações de interferência entre as operações. O trabalho seguinte de Hoppe, que trata da extração de malhas dependentes da câmera [Hop97], é um bom exemplo do uso de históricos juntamente com um estudo de análise de interferência.

Outro trabalho importante que, dentre outras contribuições, possibilita uma visão abrangente dos modelos reticulares foi o de Floriani, Puppo e Magillo [Flo97]. Nele, um MMRR é representado por um **grafo acíclico direcionado** (DAG). Os nós do grafo representam as operações do histórico e os arcos representam as relações de dependência diretas entre as operações efetuadas. Uma operação a depende diretamente de outra b , se a eliminar alguma célula criada por b (Figura 2.10), ou seja, $dom(a) \cap dom(b) \neq \emptyset$. Por outro lado, a depende indiretamente de b se existir um caminho no grafo que leve de b até a . A extração de modelos pode ser feita através de um caminhamento pelo grafo, desde que sejam respeitadas as relações de dependência. Com este novo caminhamento, o poder de expressão do MMR é maior do que o simples uso do histórico de operações, possibilitando a geração de malhas com diferentes resoluções ao longo do domínio. Uma vez que os operadores não alteram a conectividade na borda da operação, as malhas produzidas pelo caminhamento mantêm naturalmente a topologia das regiões *manifold*.

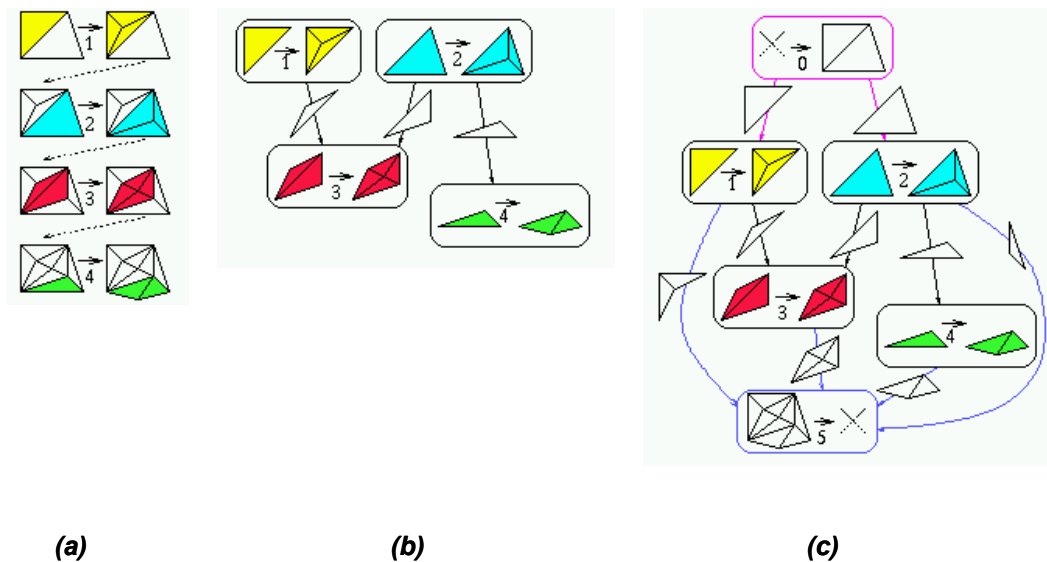


Figura 2.10 – Modelo Reticular. Em (a), histórico (H) de operações de refinamento. Em (b), o DAG que representa o histórico e suas relações de dependência. Em (c), a inserção de dois *dummy nodes* que representam a malha mais refinada e a menos refinada.