

Traçado automático de hipergrafos direcionados para gerência de projetos

Directed hypergraph drawing for project management

Zeno Stivanin¹
André Luiz Pires Guedes²

¹Copel – Companhia Paranaense de Energia
zeno.stivanin@copel.com

²Universidade Federal do Paraná
andre@inf.ufpr.br

Abstract: To promote an automatic distribution of vertices and edges of a graph following a set of established aesthetic characteristics defines the objective of a graph drawing algorithm. The objective of this work is, from a preliminary study, verify the most adequate algorithm for the automatic directed hypergraph drawing, especially focused in the representation of relationships of anteriority in activities of project management. For this purpose the GDE and VGJ software tools were used. Modifications in the VGJ software tool have been implemented in order to allow for the automatic tracing of directed hypergraphs, which is a task not yet supported by software tools of this nature.

Keywords: Directed hypergraphs, directed graph, project management, automatic drawing, graph drawing.

Resumo: Promover uma distribuição automática dos vértices e arestas de um grafo de maneira automática seguindo um conjunto de características estéticas preestabelecidas define o objetivo de um algoritmo de desenho de grafos. O objetivo deste trabalho é, a partir de um estudo preliminar, verificar o algoritmo mais adequado para o traçado automático de hipergrafos direcionados, especialmente focados na representação de relacionamentos de precedência em atividades de gerência de projetos. Para tanto foram utilizadas as ferramentas de software GDE e VGJ. Foram efetuadas modificações na ferramenta VGJ para permitir o traçado automático de hipergrafos direcionados, função inédita em ferramentas de software dessa natureza.

Palavras-Chave: Hipergrafos direcionados, grafos direcionados, gerência de projetos, traçado automático, desenho de grafos.

1 Introdução

O estudo de técnicas de traçado automático de grafos é uma área de pesquisa em constante crescimento devido à sua importância e abrangência. Há diversas aplicações para desenho automático de grafos, tais como:

- Representação gráfica de navegação na internet;
- Representação de atividades em gerência de projetos;
- Desenhos de circuitos integrados e placas de circuito impresso;
- Atividades de processamento paralelo.

Desenhar grafos de forma automática não é tarefa trivial. Os critérios que definem se um desenho de grafo é de boa qualidade são bastante subjetivos e sua caracterização pode ser influenciada até mesmo por questões culturais. Para avaliar um desenho de grafo, é necessário levar em conta as características desejadas para a representação formada. Por exemplo, para desenhar placas de circuitos é necessário que o desenho seja formado somente por linhas retas e preferencialmente formando ângulos retos. Já a representação de uma árvore genealógica não ficaria adequada com essa mesma configuração. Essa diversidade de aplicações que utilizam o traçado de grafos requer uma grande variedade de padrões de desenho, tais como padrão ortogonal, padrão árvore e circular.

Outro ponto a ser observado é que, devido à grande variedade de tipos de grafos (direcionados, não-direcionados, planares, não planares), o desenvolvimento de um algoritmo de traçado que seja capaz de gerar desenhos com qualidade para qualquer tipo de grafo de entrada é uma tarefa muito difícil e não muito utilizada. Por isso prioriza-se o desenvolvimento de algoritmos especialistas, focados em aplicações específicas de modo a possibilitar uma melhor exploração das características agregadas aumentando a qualidade do desenho.

A área de gerência de projetos é uma especialização que pode se beneficiar do uso de ferramentas de traçado automático, no que diz respeito à gerência de atividades e seus relacionamentos. Existem poucas ferramentas de software capazes de satisfazer essa necessidade, motivo pelo qual esse foco foi escolhido.

O hipergrafo direcionado, objeto de estudo deste trabalho, é uma generalização do conceito de grafo com características particulares [1] [5] [6]. O hipergrafo direcionado difere do grafo direcionado pela sua capacidade de representar fluxos e

paralelismo com maior qualidade. Existe pouca bibliografia tratando desse tipo especial de grafo, sendo que nenhum autor abordou a questão de traçado automático (excetuando-se [12], que trata do traçado de hipergrafos, porém não de hipergrafos direcionados), sendo esta, portanto, uma contribuição inédita deste trabalho.

Neste trabalho são utilizados algoritmos consagrados para o estudo sobre o traçado de hipergrafos direcionados. O que torna essa abordagem possível é a utilização da transformação do hipergrafo direcionado em grafo direcionado segundo Guedes [6] [7].

São comparados os traçados dos algoritmos por simetria, ortogonalidade e hierarquia, aplicados a hipergrafos direcionados especialmente focados na aplicação prática de representação das dependências entre as atividades de projetos.

Para o estudo prático foi utilizado o software de traçado automático de grafos GDE [9]. Finalmente o software VGJ[11] foi modificado para permitir o traçado de hipergrafos direcionados.

Dessa forma, foi possível ampliar o estudo realizado por Guedes adotando uma abordagem prática, comprovando a teoria mediante estudos de casos reais. A pesquisa original [6] [7] fornece maiores detalhes a respeito.

O resultado dos testes mostrou que o algoritmo de traçado hierárquico é o mais indicado para o traçado de hipergrafos gerenciais devido às suas características estéticas particulares e especialmente à capacidade de representação de fluxos.

Na seção 2 são apresentados os conceitos teóricos para o desenvolvimento do trabalho e na seção 3 é feita uma exemplificação sobre o uso de hipergrafos direcionados em gerência de projetos. Na seção 4 é descrito o estudo sobre os algoritmos utilizados. Na seção 5 são descritas as implementações efetuadas sobre a ferramenta de traçado para adaptá-la ao traçado de hipergrafos direcionados. Finalmente, na seção 6 são apresentadas as conclusões.

2 Considerações teóricas

Nesta seção serão apresentadas as definições teóricas básicas para auxiliar na compreensão do trabalho.

2.1 Padrão de desenho de grafo

Define-se padrão de desenho como um conjunto de características que determinam como os vértices e as arestas são dispostos no espaço.

Usualmente, independente do padrão em questão, os vértices são representados por pontos ou círculos, e cada aresta (u,v) é representada por uma simples linha ligando os pontos associados aos vértices u e v .

2.2 Hipergrafos direcionados

Esse tipo de grafo tem importância significativa em áreas tais como:

- Arquitetura paralela, onde é utilizado na representação de processamento paralelo;
- No gerenciamento de projetos, sendo utilizado para representar o relacionamento entre os serviços e atividades a serem realizadas em seu decorrer;
- Na arquitetura de software, na representação gráfica de diagramas, tais como o diagrama UML.

O hipergrafo é uma generalização de grafos. Em um hipergrafo, a cardinalidade das arestas pode ser diferente de dois. Um hipergrafo $H = (V,A)$ onde V é um conjunto finito de vértices e A é um conjunto finito de hiperarestas, onde uma hiperaresta pertencente a A é um subconjunto não-vazio de V .

Tendo que um hipergrafo direcionado é uma generalização do conceito de grafos direcionados, e uma especialização de hipergrafos, pode-se visualizar um hipergrafo direcionado como um conjunto de vértices e um conjunto de hiperarcos, onde os hiperarcos são pares de conjuntos disjuntos de vértices (X,Y) .

2.3 Representações gráficas de atividades em gerência de projetos

De acordo com [8], o método mais utilizado na maioria das ferramentas de software disponíveis para gestão de projetos é o diagrama de precedência (*Preceding Diagramming Method* – PDM), que representa as atividades nos nós (*activity-on-node* – AON), por retângulos, e as relações de precedências são estabelecidas nas setas. Esta forma de representação também é conhecida como *Método Francês*. Esse diagrama é utilizado no *MsProject* [2], (Figura 1, letra c).

As precedências, que nesse modelo são representadas nas setas, podem ser de vários tipos:

- Término/Início: o início do trabalho da sucessora depende do término da predecessora.
- Término/Término: o término do trabalho da sucessora depende do término da predecessora.
- Início/Início: o início do trabalho da sucessora depende do início da predecessora.

- Início/Término: o término do trabalho da sucessora depende da predecessora.

A precedência término/início é o tipo mais usado. Os demais são raramente utilizados e ainda não foram implementados de forma consistente. Observe-se, entretanto, que estes podem ser feitos, com pequenas adaptações, com a utilização do primeiro tipo.

Devido a característica de representação de precedência de atividades temporais, esse diagrama não permite a utilização de ciclos.

O segundo método mais utilizado é o Método do diagrama de flecha/arco (*Arrow Diagramming Method – ADM*) (Figura 1, letra b). Embora a maioria das ferramentas de software não utilize esse tipo de representação, ele é de grande importância quando soluções de personalização forem desenvolvidas, pois auxilia na programação, uma vez que utiliza as regras de programação em grafos. O método do diagrama de flecha ou arco que representa, como o próprio nome sugere, as atividades nas flechas/arcos (*activity-on-arrow – AOA*) e as relações de precedência são definidas nos eventos, sendo apenas do tipo término/início.

Para a elaboração desse tipo de representação não devem existir laços, desvios condicionais ou ciclos fechados.

Outro método utilizado é o Método do diagrama condicional (CDM) (Figura 1, letra a). Esse método de representação assemelha-se ao diagrama de precedência (PDM). No entanto, nesta forma de representação são permitidos ciclos e desvios condicionados, que tanto no PDM como no ADM não são permitidos.

3 Exemplo de uso de hipergrafos direcionados para representação de atividades de um projeto

A título de contextualização, a Companhia Paranaense de Energia – COPEL [3] possui como ferramenta de gerência de projetos de tecnologia de informação o software *Girassol*, desenvolvido internamente.

O *Girassol* foi desenvolvido com o objetivo de possibilitar o cadastro e manutenção de demandas, sua posterior classificação em projetos ou serviços e seu gerenciamento. Nesse gerenciamento de demandas, é possível ao responsável, caso necessário, designar atividades para o auxílio do desenvolvimento e melhor gerenciamento da demanda. O responsável por essa atividade pode, por sua vez, criar outras atividades relacionadas e designar responsáveis para sua execução e assim sucessivamente, sem limite predeterminado, criando dessa forma uma árvore de atividades abaixo da demanda

gerenciada. Com isso, todas as atividades necessárias para o desenvolvimento da demanda ficam registradas e são gerenciadas por meio da ferramenta.

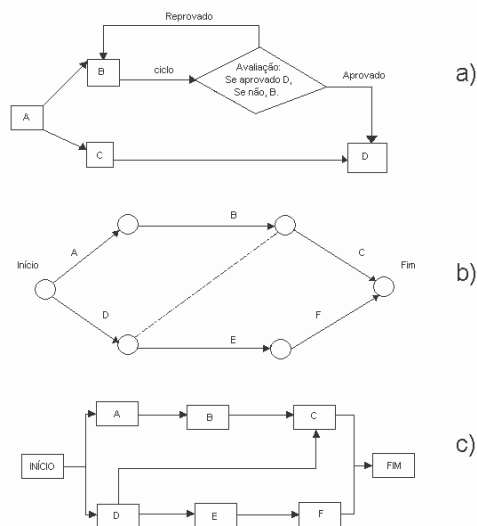


Figura 1: Representações gráficas de atividades em projetos.

Essas relações identificadas entre as atividades e suas subatividades representadas internamente no software *Girassol*, são utilizadas para indicar uma dependência direta entre a atividade pai e suas respectivas filhas, mas não possuem relação de dependência entre as atividades e seu fluxo de ocorrência.

Não há nenhuma representação do fluxo de ocorrência das atividades, o que torna muito difícil para o administrador do projeto identificar os gargalos existentes.

Para tornar mais fácil a visualização do hipergrafo extraído do projeto, foram atribuídos nomes para cada objeto conforme a tabela representada na Figura 2.

Dados os nomes para cada objeto do projeto, em conjunto com o líder do projeto, representante da COPEL, modelou-se a relação entre os objetos determinando o fluxo. Modelado o relacionamento, foram descritas as relações que determinam o hipergrafo direcionado com seus hiperarcos correspondentes (Figura 3).

A Figura 4 representa graficamente o hipergrafo direcionado que modela o fluxo das atividades do projeto utilizado. Como exemplo, foi utilizado o método do diagrama de precedência como forma de representação. Observa-se facilmente no gráfico onde ocorrem os gargalos do sistema.

Nome da Atividade Gerenciada	Apelido da Atividade Gerenciada
Marco Zero do Projeto	M0
Proposta de Estrutura Básica de Armazena	P1
Elaborar Proposta de Estrutu	AP11
Análise de Requisitos	P2
Apresentação do Domino.Dc	AP21
Elaborar documento de anal	AP22
Levantar requ	AP221
Confeccionar	AP222
Ata de Reunião para fechamento de requis	P3
efetuar reunião de fechamen	AP31
elaborar ata de reunião	AP32

Figura 2: Tabela com os nomes dos objetos.

```

M0->P4, P14, P1
P1->AP11
P4->AP41, AP42
P14->AP141
AP11->P2
P2->AP21, AP22
AP22->AP221, AP222
AP21, AP221, AP222->P3
P3->AP31, AP32
AP32, AP31->P5, P6, P7, P8, P9
P5->AP51, AP52, AP53
P6->AP61, AP62
P7->AP71
P8->AP81
P9->AP91, AP92
AP51, AP52, AP53, AP61, AP62, AP71, AP81, AP91, AP92->P10, P12
P10->AP101, AP102
P12->AP121, AP122
AP101, AP102->P11
P11->AP111
AP111, AP121, AP122->P13
P13->AP131, AP132
AP141, AP131, AP132->FIM

```

Figura 3: Exemplo de arquivo texto com representação de um hipergrafo direcionado.

Essa informação auxilia no gerenciamento das atividades de maneira a contornar os entraves que possam ocorrer.

O objetivo deste trabalho é determinar um padrão de desenho de hipergrafos direcionados tendo como premissa atender às características predominantes na especificidade dos fluxos inerentes a objetos gerenciados em projetos.

Os hipergrafos direcionados produzidos devem contemplar as representações para o método do diagrama de precedência e para o método do diagrama de flecha/arco. O método do diagrama condicional não será contemplado neste trabalho já que não é utilizado na representação do fluxo de atividades e sim na representação de

processos. Desse modo, não haverá a preocupação na representação de ciclos.

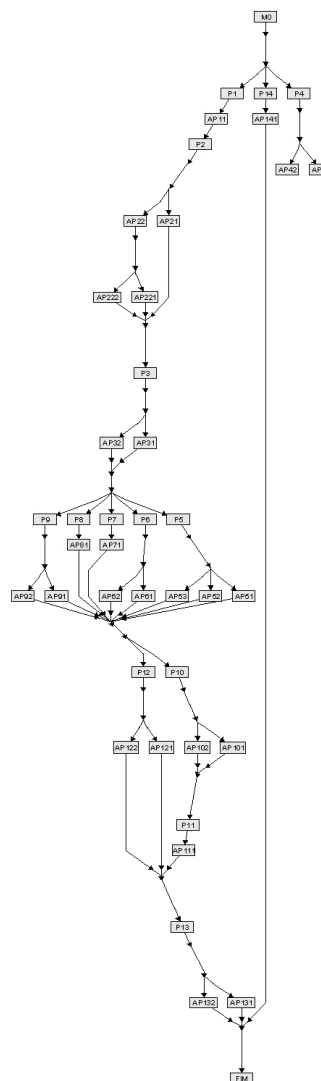


Figura 4 Hipergrafo Direcionado traçado a partir de exemplo de projeto do software Girassol.

Os exemplos de projetos modelados e estudados em forma de hipergrafos direcionados foram fornecidos pela COPEL.

4 Estudo do padrão de desenho de hipergrafos direcionados para gerência de projetos

Nesta seção serão abordados os passos efetuados para a escolha do algoritmo e padrão de traçado que mais se adapta ao hipergrafo direcionado.

4.1 Escolha do algoritmo de traçado e padrão de desenho

Observando os hipergrafos direcionados voltados para modelagem de fluxo de atividades gerenciais, nota-se as características de uniformidade, minimização de cruzamentos, uniformização dos tamanhos das arestas, simetria e fluidez da representação hierárquica.

Dessa forma, o padrão de desenho ortogonal não é o mais indicado pela falta de cuidado em relação a representabilidade de fluxos, fator de forte importância no tipo de grafo trabalhado.

Da mesma forma, o uso do algoritmo Simetria, que gera um traçado baseado no padrão linha reta, também não se aplica à solução do problema já que preza pela simetria, deixando de lado, como no caso do traçado ortogonal, a representabilidade do fluxo.

A forma mais adequada de desenhar o hipergrafo em questão é utilizar o algoritmo de traçado hierárquico gerando o padrão de desenho poligonal. As características de algoritmo adaptam-se às características requeridas no desenho dos hipergrafos. A representação do fluxo é bem abordada, juntamente com as outras características.

A abordagem de representação adotada neste trabalho será o uso do algoritmo de traçado de grafos de forma hierárquica aplicado ao traçado de hipergrafos direcionados.

De modo a comprovar na prática a eficiência dessa representação, foi desenvolvido um software de traçado de hipergrafos direcionados. Nas seções a seguir serão descritas as etapas realizadas de modo a atingir esse objetivo.

4.2 Transformação de hipergrafos direcionados em grafos direcionados

Para testar o traçado de hipergrafos direcionados com alguns padrões de desenho, torna-se extremamente útil a transformação de hipergrafos direcionados em grafos direcionados.

O objetivo é obter o produto da transformação e utilizar ferramentas de software especialistas em traçado de grafos para estudar as características de cada padrão.

A técnica de transformação utilizada neste trabalho é descrita em [6] e [7]. Esta técnica transforma hipergrafos direcionados em grafos direcionados mantendo as características de planaridade.

Dado um hipergrafo direcionado qualquer, para cada hiperarco promove-se a transformação da seguinte forma:

- Para cada hiperarco, criam-se, no lugar do fluxo, dois vértices interligados por uma aresta.
- Cada um dos vértices criados será ligado via arestas com os vértices que compõem o grupo de origem e de destino do hiperarco. Ou seja, para cada hiperarco $h = (X, Y)$, são criados os arcos $(X, C1)$, $(C1, C2)$ e $(C2, Y)$. A Figura 5 exemplifica a transformação.

Tomando como base a técnica de transformação descrita, foi desenvolvida uma ferramenta de transformação. Ela foi construída em linguagem *Java* e tem como característica receber dois parâmetros de entrada que são respectivamente: (i) nome do arquivo de entrada e, (ii) nome do arquivo de saída, em formato *CSV*, que é utilizado pelo software *GDE* [9].

Observando os parâmetros de entrada, a chamada do programa de transformação criado ficou da seguinte forma:

Hiper2Graph entrada saida.csv

Quanto ao formato do arquivo de entrada, ele deve ser composto de uma ou mais linhas. Cada linha deverá conter um par $O \rightarrow D$, onde O é um conjunto não vazio de vértices representando origens separadas por vírgula quando necessário e D é outro conjunto não vazio de vértices também separados por vírgula, quando necessário. O conjunto de caracteres “->” representa a orientação do hipergrafo. Dessa maneira foi possível representar um hipergrafo direcionado de maneira simples e clara. A Figura 6 exemplifica alguns casos de transformação de hipergrafos direcionados em grafos direcionados.

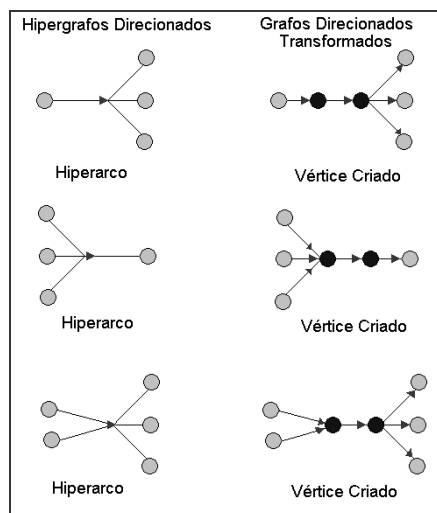


Figura 5: Exemplo de transformação de hipergrafo direcionado para grafo direcionado.

entrada1.txt	entrada2.txt	entrada3.txt	entrada4.txt
1->2,3,4	1,2,3->4	1,2,3->4,5	1->4 2->3 1->3 2->4 1,2->3,4
saida1.csv	saida2.csv	saida3.csv	saida4.csv
1;C1 C1;C2 C2;2 C2;3 C2;4	1;C1 2;C1 3;C1 C1;C2 C2;4	1;C1 2;C1 3;C1 C1;C2 C2;4 C2;5	1;4 2;3 1;3 2;4 1;C1 2;C1 C1;C2 C2;3 C2;4

Figura 6: Exemplos de transformação de hipergrafo direcionado para grafo direcionado.

Note-se nos exemplos que os vértices criados pelo algoritmo são nomeados iniciando com a letra C de modo a tornar fácil a distinção entre os demais vértices já existentes.

4.3 Promovendo testes de traçado de hipergrafos direcionados

Com a implementação do transformador de hipergrafos direcionados em grafos direcionados foi possível a realização de testes de traçado nos padrões disponíveis na ferramenta GDE.

Essa ferramenta foi escolhida devido à qualidade do seu traçado e à variedade de padrões de traçado programados.

Foram efetuados testes preliminares com hiperarcos simples, sem levar em conta os exemplos reais, de modo a verificar as características estéticas mais gerais.

A Figura 7 mostra os traçados automáticos nos padrões Hierárquico, Ortogonal e por Simetria produzidos automaticamente através da ferramenta GDE após a importação dos arquivos chamados *saida1.csv*, *saida2.csv*, *saida3.csv*, *saida4.csv*, que são o resultado da transformação feita através da ferramenta de transformação desenvolvida aplicados aos arquivos *entrada1.txt*, *entrada2.txt*, *entrada3.txt* e *entrada4.txt*, descritos anteriormente.

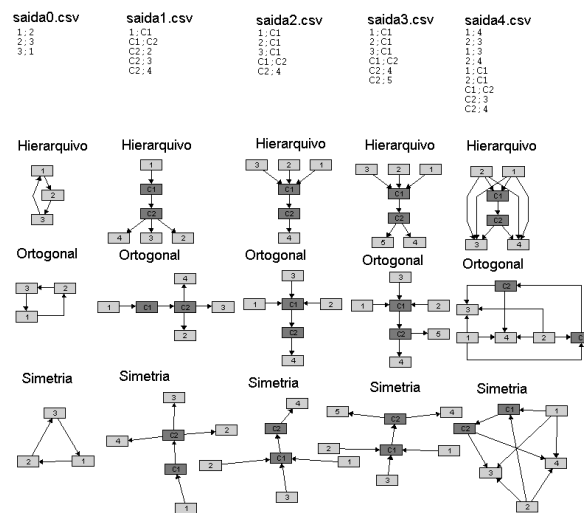


Figura 7: Exemplo do estudo preliminar de traçado.

Observando a figura, nota-se que mesmo para exemplos pequenos como os utilizados nos primeiros experimentos já é possível notar a capacidade de representar fluxo utilizando o algoritmo hierárquico. Com o aumento da complexidade dos desenhos que ocorre da esquerda para a direita o poder de representação de fluxos torna-se mais explícito.

Analisando o resultado do uso do algoritmo de traçado ortogonal observa-se que para exemplos simples, como o encontrado no arquivo *saida1.csv*, a diferença na qualidade de representação dos fluxos não fica tão evidente. Com o aumento da complexidade do fluxo, mais precisamente nos dois últimos exemplos, a diferença na qualidade dessa característica é mais evidente.

O algoritmo de simetria, do primeiro ao quarto hipergrafo direcionado, apresentou uma boa aderência à característica aqui comparada. Houve perda de qualidade devido à característica de prezar pela distribuição uniforme dos vértices, buscando a padronização das distâncias entre eles, o que levou a um mascaramento no fluxo, fato que não ocorre no algoritmo hierárquico. No último exemplo, a representação do fluxo ficou bastante prejudicada devido às características prezadas pelo algoritmo.

Após esses testes preliminares, foram efetuados testes com hipergrafos direcionados voltados ao foco de estudo, a saber, ao fluxo de atividades em projetos. A seguir apresenta-se a análise efetuada no hipergrafo modelado na Seção 3 do exemplo real retirado do software Girassol.

Observando o resultado do uso do algoritmo de traçado automático por simetria sobre o hipergrafo

objeto de estudo mostrado na Figura 8, tem-se com o caso real a comprovação de que com esse algoritmo não é possível uma boa representação visual do fluxo inerente à modelagem, tornando inviável o uso desse algoritmo para o traçado dos hipergrafos objeto deste estudo.

O mesmo acontece no traçado do hipergrafo quando se utiliza o algoritmo de traçado ortogonal mostrado na Figura 9. A capacidade de visualização do fluxo é um pouco melhor que no caso anterior, mas ainda insuficiente para uma interpretação clara do fluxo.

Observando o resultado do traçado automático do hipergrafo utilizando o algoritmo hierárquico, (Figura 10) é visualmente clara a qualidade da representação do fluxo, atestando que esta é a melhor escolha tratando-se do tipo de hipergrafo especial em que o estudo é focado.

Comparando os resultados dos três algoritmos, comprovou-se de forma prática que a maneira mais adequada de traçar os hipergrafos direcionados para gerenciamento de projetos é por meio da utilização do algoritmo hierárquico. Sendo assim, este será o algoritmo a ser adaptado.

5 Ferramenta de traçado automático de hipergrafos direcionados

Esta seção destina-se a descrever as etapas empregadas na adaptação da ferramenta VGJ de modo a prepará-la para traçar automaticamente hipergrafos direcionados.

5.1 Desenvolvimento da ferramenta

Foi escolhida a ferramenta VGJ para adaptação e uso neste trabalho por ser implementada em Java, com política de código aberto [4] e preparada para funcionar por meio da *Web* sobre tecnologia

applet. Com isso, a distribuição da ferramenta após as modificações é facilitada. Outra característica que contribuiu para a escolha da ferramenta foi o fato de implementar um algoritmo de traçado de grafos direcionados hierarquicamente, dispensando a implementação completa de outro algoritmo.

A seguir são descritas as implementações efetuadas na ferramenta.

5.2 Implementação da ferramenta

A primeira implementação efetuada no programa original foi permitir a leitura de um arquivo contendo uma representação de hipergrafo direcionado. Dessa forma, o programa é capaz de interpretar o hipergrafo como um grafo direcionado.

Inicialmente foi padronizado o tipo de arquivo de entrada da mesma forma como feito no experimento anterior sobre o transformador de hipergrafos direcionados em grafos direcionados.

Nesse ponto o programa está preparado para ler arquivos contendo a representação padronizada de hipergrafos direcionados. Com isso, desenvolveu-se o algoritmo de transformação de hipergrafos direcionados em grafos direcionados. O algoritmo utilizado foi baseado na teoria de transformação apresentada anteriormente [6].

O padrão de arquivos suportado pelo programa é o formato GML [10]. Dessa forma, foi necessário desenvolver uma transformação da representação do grafo direcionado, já em memória, para o formato GML.

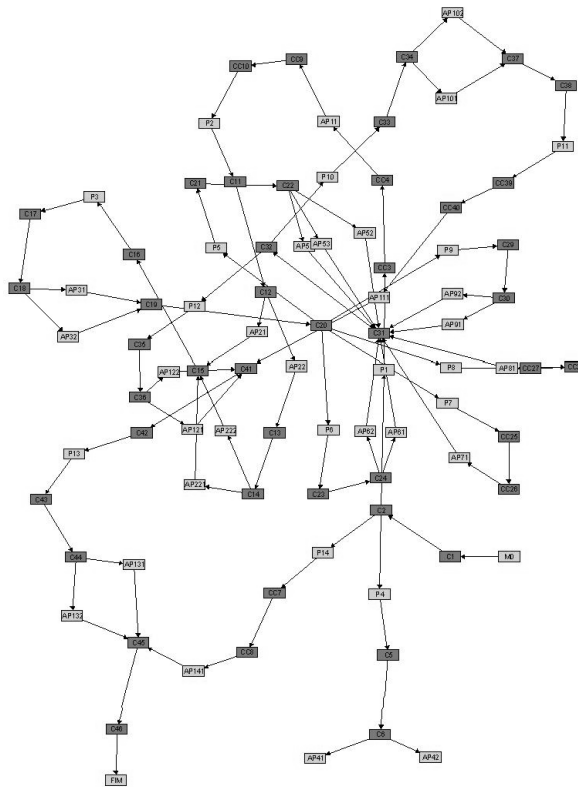


Figura 8: Traçado automático de hipergrafo do Girassol - Algoritmo Simetria.

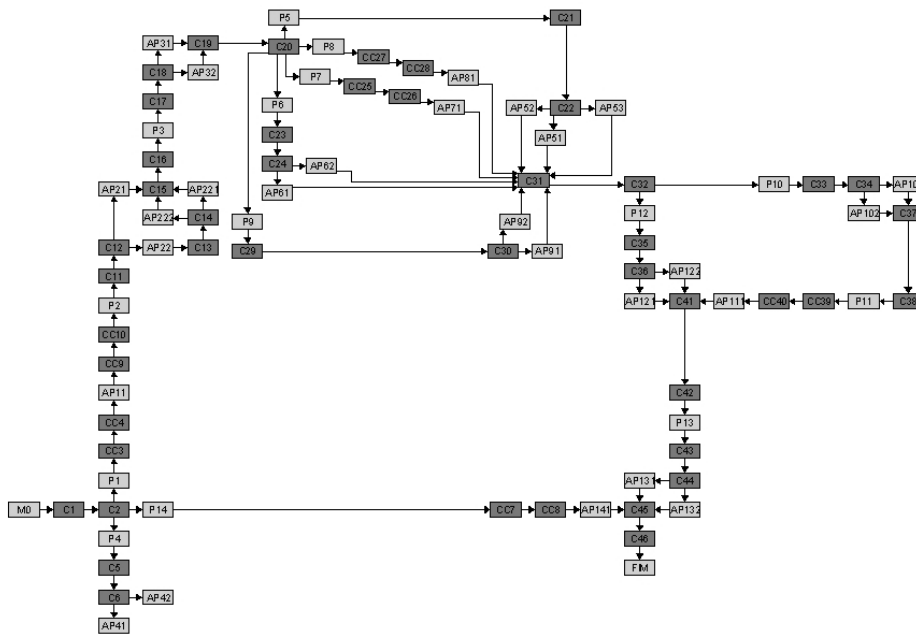


Figura 9: Traçado automático de hipergrafo do Girassol - Algoritmo Ortogonal.

uma representação especial para indicação de fluxo (Figura 12, letra d).

5.3 Resultados

A Figura 13 traz as três representações gráficas do mesmo hipergrafo direcionado. Esse hipergrafo foi modelado na Seção 3 a partir de exemplo prático extraído do software Girassol da Copel. A letra "a" mostra uma representação esperada, que traz embutida as características desejáveis para a boa interpretação do problema de fluxos em atividades de projetos. A letra "b" é o resultado prático do traçado do mesmo hipergrafo direcionado utilizando a ferramenta GDE para o estudo da abordagem mais adequada. Deve-se notar que esse traçado foi feito com a representação de um grafo direcionado resultado da transformação do hipergrafo objeto de origem. Finalmente a letra "c" é o resultado gráfico obtido com o uso da ferramenta VGJ após a aplicação das implementações aqui descritas.

O resultado obtido possui uma ótima qualidade de representação de fluxo, cumprindo assim os objetivos preestabelecidos anteriormente. Nota-se que a distribuição dos vértices contém a característica de que a direção do arco do ancestral sempre é anterior aos seus descendentes, que o balanceamento dos vértices ocorre da direita para a esquerda nos níveis, e que há uma minimização de cruzamentos entre as arestas e poucos cotovelos.

6 Conclusão

Tendo em vista que o objetivo do trabalho era, a partir de um estudo prático teórico, identificar, entre os algoritmos por simetria, hierárquico e ortogonal, qual o mais apropriado para o traçado de hipergrafos direcionados focados na representação de atividades inerentes a projetos, além da adaptação de uma ferramenta de traçado de grafos direcionados, permitindo com ela o traçado de hipergrafos direcionados, conclui-se que o objetivo foi alcançado.

O estudo prático teórico efetuado com o auxílio da ferramenta de traçado de grafos direcionados GDE, com o reaproveitamento do algoritmo de transformação de hipergrafos direcionados em grafos direcionados, comprovou a eficácia do algoritmo hierárquico devido às suas características que se adaptam para solucionar o problema, além da qualidade visual evidenciada no estudo prático.

O software desenvolvido sobre o aplicativo VGJ permitiu o aproveitamento do algoritmo de traçado hierárquico pré-implementado para traçar automaticamente hipergrafos direcionados. Por ter sido desenvolvido sobre tecnologia Java e com a possibilidade de ser aplicado sobre uma página *web*, há um grande poder de adaptação a ser explorado. Por exemplo, qualquer software de gerência de projetos que deseje fazer uso da ferramenta desenvolvida para agregar visualização do fluxo de atividades dos projetos pode facilmente acoplá-lo mediante a construção de uma pequena interface com tal finalidade.

Outros pesquisadores que desejarem estudar o desenho de hipergrafos direcionados podem utilizar a mesma abordagem de aproveitamento de representação sobre um grafo direcionado e assim aproveitar toda a variedade de algoritmos desenvolvidos para tal fim, promover os testes, comparações e análises mais facilmente.

Depois da integração, pode-se ainda construir uma interface para a visualização, configuração e cadastro de características inerentes às atividades, aumentando assim o poder da ferramenta.

6.1 Sugestões de trabalhos futuros

Devido à importância da representação de ciclos em outras aplicações de grafos, um possível trabalho futuro é promover a alteração na ferramenta desenvolvida com o objetivo de permitir o traçado de hipergrafos direcionados que contenham ciclos. Dessa forma, a ferramenta torna-se mais genérica e poderá ser utilizada com objetivos diferentes daquele abordado neste trabalho.

Outro ponto que pode contribuir para o aumento da qualidade do traçado gerado é o estudo e melhoria no algoritmo de traçado, de maneira que a minimização dos cruzamentos entre as arestas seja melhorada principalmente para os casos dos grafos planares.

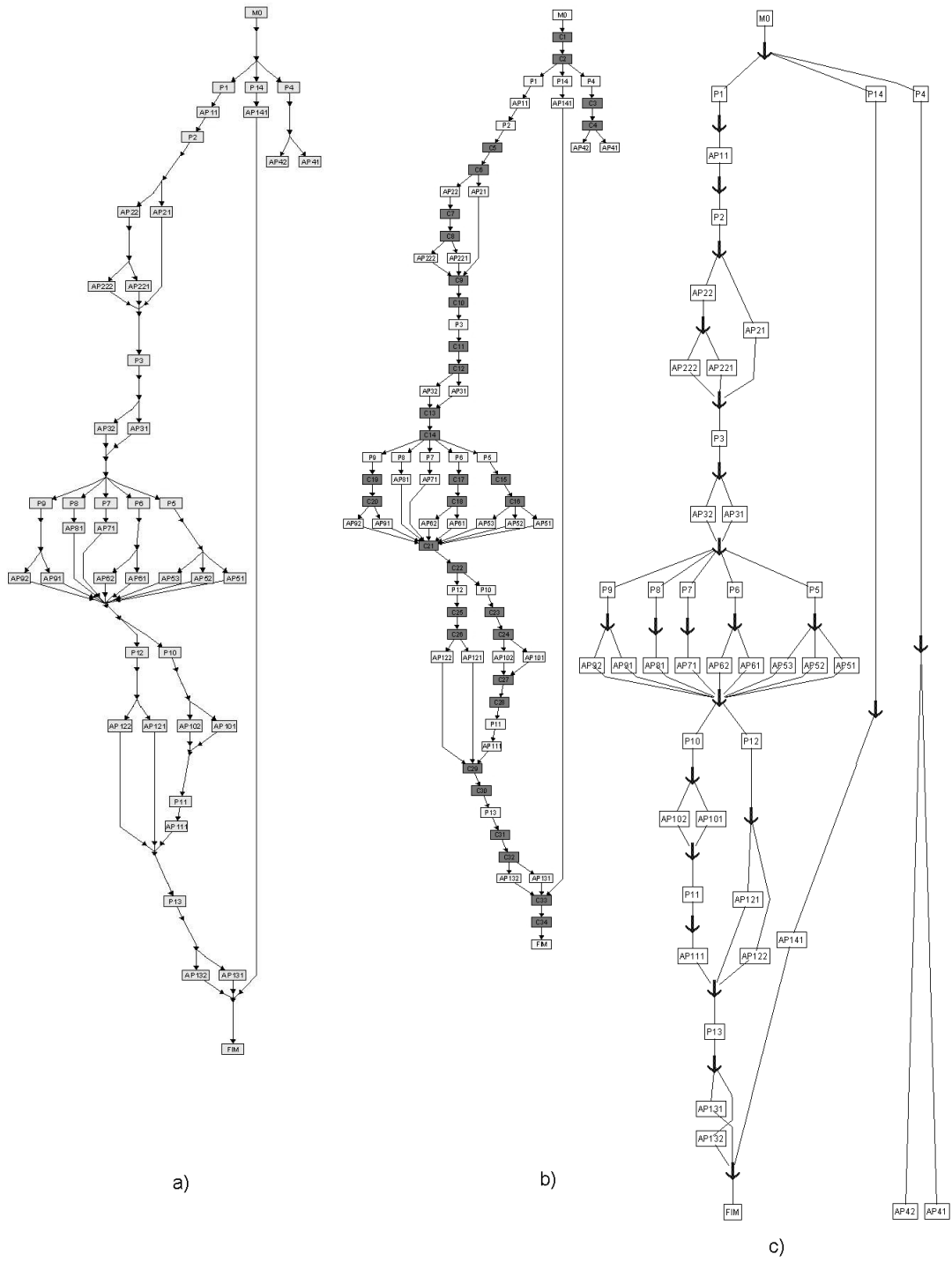


Figura 13: Resultado obtido com as alterações do VGJ.

7 Referências

- [1] Giorgio Ausiello, Alessandro D'Atri, e Domenico Sacc'a. Graph algorithms for functional dependency manipulations. *Journal of the ACM*, 30(4):752–766, outubro de 1983.
- [2] Microsoft Corporation. Microsoft Project 2003. <http://www.microsoft.com/brasil/office/projec/default.asp>, acessado em 14/04/2005.
- [3] Companhia Paranaense de Energia. Companhia Paranaense de Energia. <http://www.copel.com/>, acessado em 20/10/2005.
- [4] Free Software Foundation. GNU GENERAL PUBLIC LICENSE. http://www.eng.auburn.edu/department/cse/research/graph_drawing/COPYING, acessado em 04/04/2005.
- [5] Giorgio Gallo, Giustino Longo, Sang Nguyen, e Stefano Pallotino. Directed hypergraphs and applications. *Discrete Applied Mathematics*, 42(177-201):3–60, 1993.
- [6] André Luiz Pires Guedes. Hipergrafos Direcionados. Thesis (D.Sc.), Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brasil, 2001.
- [7] André Luiz Pires Guedes e Lilian Markenzon. Directed hypergraph planarity. *Pesquisa Operacional*, 25(3):383–390, setembro/dezembro de 2005.
- [8] Marly Monteiro de Carvalho e Roque Rabechini Jr. Construindo competências para gerenciar projetos: teoria e casos. Atlas, 1th edition, 2005.
- [9] Oreas Automatic Diagram Layouts. Oreas - Automatic Diagram Layouts. <http://www.oreas.com/>, acessado em 04/04/2005.
- [10] Marcus Raitner. The GML File Format. <http://infosun.fmi.uni-passau.de/Graphlet/GML/>, acessado em 20/10/2004.
- [11] Computer Science e Software Engineering. Drawing Graphs with VGJ. http://www.eng.auburn.edu/department/cse/research/graph_drawing/vgj.html, acessado em 30/03/2005.
- [12] Laura Tarantino and Tiziana Catarci. A Hypergraph -based Framework for Visual Interaction with Databases. *Journal of Visual Languages & Computing* Volume 6, Issue 2 , June 1995.