

UNIVERSIDADE FEDERAL FLUMINENSE

RIAN GABRIEL SANTOS PINHEIRO

**Método Exato para Biclusterização por Edição de
Arestas e Aplicação em Formação de Células de
Manufatura**

NITERÓI

2012

UNIVERSIDADE FEDERAL FLUMINENSE

RIAN GABRIEL SANTOS PINHEIRO

Método Exato para Biclusterização por Edição de Arestas e Aplicação em Formação de Células de Manufatura

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para obtenção do Grau de Mestre em Computação. Área de concentração: Algoritmos e Otimização.

Orientador:
Luiz Satoru Ochi

Coorientador:
Fábio Protti

NITERÓI
2012

Método Exato para Biclusterização por Edição de Arestas e Aplicação em Formação de Células de Manufatura

Rian Gabriel Santos Pinheiro

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para obtenção do Grau de Mestre em Computação. Área de concentração: Algoritmos e Otimização.

Aprovada por:

Prof. Luiz Satoru Ochi / IC-UFF (Presidente)

Prof. Fábio Protti / IC-UFF

Prof. Luidi Gelabert Simonetti / IC-UFF

Prof. Luerbio Faria / UERJ

Niterói, 31 de agosto de 2012 .

Ficha Catalográfica elaborada pela Biblioteca da Escola de Engenharia e Instituto de Computação da UFF

P654 Pinheiro, Rian Gabriel Santos
Método exato para biclusterização por edição de arestas e aplicação
em formação de células de manufatura / Rian Gabriel Santos
Pinheiro. – Niterói, RJ : [s.n.], 2012.
90 f.

Dissertação (Mestrado em Computação) - Universidade Federal
Fluminense, 2012.

Orientador: Luiz Satoru Ochi, Fábio Protti.

1. Computação. 2. Clusterização. 3. Algoritmo. 4. Edição de
arestas. I. Título.

CDD 004

Agradecimentos

Agradeço inicialmente a Rosane, minha mãe e amiga, pela contribuição educacional, pelas palavras de incentivo e pelas orientações.

Agradeço à minha família: Ronaldo, Marlene, Diego, Luiza e Juan.

A Sandra e Helena, madrinha e tia que sempre estiveram presentes me dando força e orientação nas minhas escolhas.

Agradeço aos meus orientadores, Prof. Satoru e Prof. Fábio, pela orientação, pelos conselhos e por ajudar na minha formação acadêmica.

Ao Prof. Luidi pelas contribuições neste trabalho.

Aos meus amigos: Ivan, Paulo, Phillipe, Manu, Alisson e David.

Aos amigos e colegas da UFF e da UFAL, em especial ao Tamer, Uérverton, Igor e Eyder.

Ao Prof. Alejandro e a Prof. Eliana pela inspiração e apoio ao mestrado.

Aos amigos de Maceió.

A CAPES e a FAPERJ pelo apoio financeiro.

Rian Gabriel

Resumo

O PROBLEMA DE BICLUSTERIZAÇÃO POR EDIÇÃO DE ARESTAS (PBEA) é um problema $\mathcal{NP}$ -difícil em que, dado um grafo bipartido $G = (V, U, E)$ e um inteiro $k \geq 0$, deseja-se adicionar ou remover no máximo k arestas de forma a tornar G uma união disjunta de subgrafos bipartidos completos (bicliques). Neste trabalho, um método exato para o PBEA é proposto. Este método é um *Branch and Cut* com algoritmo de separação, heurística primal e pré-processamento. Ele foi testado em instâncias representadas por grafos bipartidos aleatórios, que foram gerados utilizando o modelo $\mathbb{G}(m, n, p)$, também conhecido como modelo binomial. É provada a relação do PBEA com o PROBLEMA DE FORMAÇÃO DE CÉLULAS DE MANUFATURA (PFCM). Com isso, foi desenvolvido um algoritmo exato para o PFCM que resolveu 27 das 35 instâncias da literatura, e proposto um novo modelo matemático linear fracionário para o problema. Os resultados foram obtidos através de implementação na linguagem C++ juntamente com o pacote de otimização linear mista CPLEX. Foi observado que o algoritmo *Branch and Cut* proposto é mais eficiente do que o do CPLEX sozinho e que o pré-processamento reduz o tempo do algoritmo, principalmente se executado juntamente com a heurística.

Palavras-chave: Clusterização. Edição de Arestas. Método Exato

Abstract

The BICLUSTER GRAPH EDITING PROBLEM (BGEP) is a $\mathcal{NP}$ -complete problem such that, given a bipartite graph $G = (V, U, E)$ and an integer $k \geq 0$, asks whether it is possible to add and/or remove at most k edges in order to make G a union of complete bipartite subgraphs (bicliques). In this paper, an exact method for the BGEP is presented. It is composed by a Branch-and-Cut method with a separation algorithm, a primal heuristics and a pre-processing step. The instances for the problem are obtained by random bipartite graphs that have been generated using the model $\mathbb{G}(m, n, p)$, also known as binomial model. It is also proposed an application to the MANUFACTURING CELL FORMATION PROBLEM (MCFP), by developing an exact algorithm that solved 27 of 35 instances from the literature of the MCFP, and a new linear fractional model for this problem. The results were obtained through the implementation in C++ with the mixed linear optimization package CPLEX. It was observed that the proposed Branch-and-Cut is more efficient than the CPLEX alone, and the preprocessing reduces the time of the algorithm, especially if performed with the heuristic.

Keywords: Clusterization. Graph Editing. Exact Method.

Sumário

Glossário	vii
Lista de Figuras	viii
Lista de Tabelas	ix
1 Introdução	1
1.1 Motivação	1
1.2 Descrição e Literatura do Problema	1
1.3 Algumas Aplicações para o PBEA	4
1.3.1 Análise de Dados Biológicos	4
1.3.2 Projeto de Redes <i>Multicast</i>	5
1.3.3 Mineração de Dados	5
1.3.4 Formação de Células de Manufatura	6
1.4 Estrutura do Trabalho	6
2 Proposta de algoritmo Branch-and-Cut	7
2.1 Métodos Exatos	7
2.1.1 <i>Branch-and-Bound</i>	8
2.1.2 Método de Planos de Corte	9
2.1.3 <i>Branch-and-Cut</i>	10
2.2 Formulação	10
2.3 Algoritmo de Separação	12
2.4 Heurísticas Primais	13
2.4.1 Algoritmo Determinístico 11-aproximado de Amit [3]	13
2.4.2 Algoritmo Randômico 4-aproximado de Guo et al. [28]	17
2.4.3 Comparação Amit $\times$ Guo et al.	18
2.5 Pré-processamento	18
2.6 Instâncias	22
3 Aplicação para o Problema de Formação de Células de Manufatura	24
3.1 Problema de Formação de Células de Manufatura	24
3.1.1 Literatura do PFCM	25
3.1.2 Tamanho da Célula	26
3.1.3 Funções Objetivo do PFCM	27
3.2 Algoritmo Exato para o PFCM	27
3.3 Nova Formulação Matemática para o PFCM	33
3.4 Instâncias do PFCM	33

4	Resultados Computacionais	35
4.1	Ferramentas	35
4.2	Métricas de Avaliação para o BGEP	35
4.2.1	Média Geométrica	36
4.2.2	Fator de aceleração	36
4.3	Resultados para o PBEA	37
4.4	Resultados para a Célula de Manufatura	42
5	Conclusões e Trabalhos Futuros	45
A	Instâncias	47
A.1	PBEA	47
A.2	PFCM	51
B	Soluções Ótimas do PBEA	60
C	Soluções Ótimas para o PFCM	64
C.1	Soluções sem Restrição <i>Singleton</i>	64
C.2	Soluções com Restrição <i>Singleton</i>	73
D	Erros encontrados na literatura do PFCM	83
D.1	Instâncias Erradas	83
D.2	Resultados Incoerentes	83
	Bibliografia	90

Glossário

B&B	:	Branch and Bound
B&C	:	Branch and Cut
BGEP	:	BICLUSTER GRAPH EDITING PROBLEM
BGEPS	:	Problema de Edição de Grafos Bicusters com <i>Singleton</i>
BGEPS(k)	:	BGEPS Parametrizado
LI	:	Limite Inferior
LS	:	Limite Superior
PFCM	:	Problema de Formação de Células de Manufatura
PI	:	Programação Inteira
PL	:	Programação Linear

Lista de Figuras

1.1	Exemplo do PBEA.	2
1.2	Estruturas de biclusterização.	4
2.1	Árvore do Branch-and-Bound [4].	9
2.2	Subgrafo proibido P_4	11
2.3	Restrição Violada para o PBEA	12
2.4	Teorema 2.5.1: Caso 1a	19
2.5	Teorema 2.5.1: Caso 1b	20
2.6	Teorema 2.5.1: Caso 2	21
3.1	Exemplo de uma resolução do PFCM.	25
3.2	Exemplo de transformação PBEA $\leftrightarrow$ PFCM.	28
3.3	Exemplo: Algoritmo Exato PFCM.	32

Lista de Tabelas

2.1	Probabilidade de modificação de aresta do algoritmo ApproxBicluster . . .	17
3.1	Instâncias da Literatura do PFCM	34
4.1	Exemplo: média geométrica.	36
4.2	Cenários considerados.	37
4.3	Restrições geradas pelo Pré-Processamento	39
4.4	Resultado PBEA.	40
4.5	Fator de aceleração	41
4.6	Resultado do Algoritmo Exato com restrição de <i>Sigleton</i> 2×2	43
4.7	Resultado do Algoritmo Exato sem restrição de <i>Sigleton</i>	44

Capítulo 1

Introdução

Neste capítulo é apresentada, primeiramente, a motivação para o problema estudado neste trabalho. Em seguida, descreve-se o problema, juntamente com uma revisão bibliográfica e mostram-se diversas aplicações para o problema. Por fim, a estrutura do trabalho é apresentada.

1.1 Motivação

No ponto de vista prático do problema, pode-se citar várias aplicações nas quais o PROBLEMA DE BICLUSTERIZAÇÃO POR EDIÇÃO DE ARESTAS (PBEA) é aplicado, dentre elas: análise de dados biológicos, projetos de redes *multicast*, mineração de dados e formação de células de manufatura (proposta neste trabalho). Quanto ao ponto de vista teórico, a versão de otimização do PBEA pertence à classe $\mathcal{NP}$ -difícil [3]. Assim, encontrar uma forma viável de resolver esse problema já corresponde a um desafio. Além disso, o fato de o problema ser pouco explorado na literatura acrescenta uma motivação extra por haver muito trabalho a ser desenvolvido.

1.2 Descrição e Literatura do Problema

O PBEA é um problema $\mathcal{NP}$ -completo em que, dado um grafo bipartido $G = (V, U, E)$ e um inteiro $k \geq 0$, onde V e U são conjuntos não vazios de vértices e E é um conjunto de arestas, o objetivo é adicionar ou remover no máximo k arestas de forma a tornar G uma união disjunta de subgrafos bipartidos completos maximais (bicliques).

Um problema semelhante é o CLUSTER GRAPH EDITING PROBLEM, primeiramente estudado por Gupta & Palit [29]. Ele consiste em tornar G uma união disjunta de subgrafos completos (cliques), onde o grafo de entrada G não é necessariamente bipartido. Ambos são casos de problemas de partição em grafos.

Vários tipos de algoritmos de particionamento são descritos por Pothen [52] e são

aplicados em diversas áreas da ciência da computação. Já Alpert & Kahng [2] mostram como explorar o espaço de soluções viáveis através de algoritmos gulosos, buscas locais, *simulated annealing* e algoritmos evolucionários. No entanto, pouca coisa é apresentada para resolver o PBEA.

O conceito de biclusterização foi introduzido em meados da década de 70 [30], mas o primeiro uso do termo se deu no contexto da biologia computacional com o trabalho de Cheng & Church [16]. Desde então, algoritmos têm sido propostos e utilizados neste e em outros campos de aplicação [1, 6, 22].

Nomes como co-clusterização, clusterização bidimensional, entre outros, são frequentemente utilizados na literatura para se referir ao mesmo problema. Em suma, este conceito denota a existência de submatrizes “significativas” em uma dada matriz, ou seja, um subconjunto de linhas e de colunas que possuem padrões únicos baseados em um certo método de classificação.

No entanto, existem diversas formulações e contextos para o problema de biclusterização. É possível representar a matriz através de um grafo bipartido, considerando as linhas e colunas como vértices, no qual cada conjunto de vértices (linhas e colunas) pertencem a partes diferentes da mesma bipartição.

Em um grafo bipartido sem pesos, um *bicluster* ideal é um subgrafo bipartido completo (biclique). A presença de bicliques indicaria um alto grau de similaridade entre os dados agrupados. Em especial, chama-se de grafo *bicluster* (ou biclusterizado) o grafo bipartido no qual cada uma de suas componentes conexas forma uma biclique.

Neste trabalho, é focada a versão de otimização de um dos diversos problemas de biclusterização apresentados e definidos em Amit [3], o PBEA.

PROBLEMA DE BICLUSTERIZAÇÃO POR EDIÇÃO DE ARESTAS

INSTÂNCIA: Um grafo bipartido $G = (U, V, E)$ e um inteiro $k \geq 0$.

QUESTÃO: É possível realizar no máximo k edições de arestas (remoções e adições) de forma a tornar G um grafo *bicluster*?

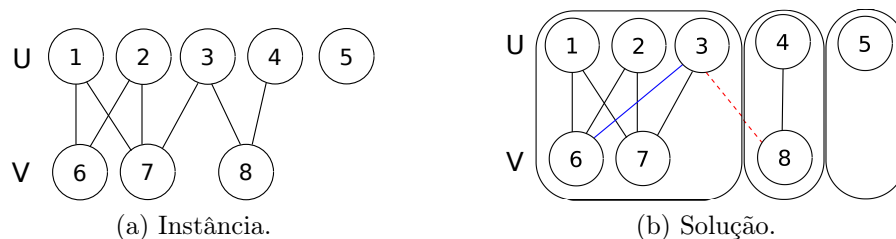


Figura 1.1: Exemplo do PBEA.

A Figura 1.1 mostra um exemplo de um grafo onde uma adição, aresta $(3, 6)$, e uma remoção, aresta $(3, 8)$, tornam-o em um grafo biclusterizado. Observe que esta não é a solução ótima, uma vez que bastaria uma edição, remover a aresta $(3, 7)$, para torná-lo um grafo *bicluster*. Além disso, considera-se um único vértice como uma biclique,

representada na figura pelo vértice 5.

Amit [3] provou que o PBEA (em sua versão de decisão) é um problema $\mathcal{NP}$ -completo, a partir de uma redução polinomial do problema 3-EXACT 3-COVER, e propõe uma formulação em programação inteira binária e um algoritmo 11-aproximado para o problema. Já o trabalho de Protti et al. [53] propõe um algoritmo que usa uma nova estratégia baseada em técnicas de decomposição modular. O algoritmo resolve o problema em tempo $O(4^k + n + m)$, em que k é o número de edições no grafo, n é o número de vértices e m o número de arestas. Com base no trabalho de Amit [3], Guo et al. [28] propõem um algoritmo randômico 4-aproximado que será explorado na Seção 2.4.2.

Outras variações do problema são apresentadas por Amit [3]. Algumas variações tornam-se polinomiais, como encontrar a biclique com a máxima soma de pesos em seus vértices. Porém, a maioria continua na classe $\mathcal{NP}$ -completo, como:

- encontrar uma biclique cujas partições possuem cardinalidades a e b respectivamente;
- encontrar uma biclique com partes com cardinalidades maiores que um inteiro k ;
- encontrar uma biclique com mais que k arestas;
- encontrar a biclique com a máxima soma de pesos em suas arestas.

No que diz respeito a meta-heurísticas, os recentes trabalhos de Pinheiro et al. [51] e Sousa Filho et al. [57] abordam o problema através do método GRASP. Em Pinheiro et al. [51], o GRASP é comparado com um método exato adicionado ao pré-processamento descrito neste trabalho. Já Sousa Filho et al. [57] comparam seu GRASP com as heurísticas da literatura — apresentadas na Seção 2.4 —, e propõem uma regra de redução.

Segundo Madeira & Oliveira [42], algoritmos de biclusterização assumem uma das seguintes situações: ou existe apenas um *bicluster* (como na Figura 1.2a), ou existem K *biclusters*, em que K é geralmente definido *a priori*. Quando o algoritmo assume a existência de vários *biclusters*, as estruturas da biclusterização podem variar de acordo com o problema, como mostra a Figura 1.2. Essas estruturas podem ser:

- (a) *bicluster* único;
- (b) *biclusters* com linhas e colunas exclusivas;
- (c) *biclusters* não sobrepostos com estrutura xadrez;
- (d) *biclusters* linhas exclusivas;
- (e) *biclusters* colunas exclusivas;
- (f) *biclusters* não sobrepostos com estrutura de árvore;

- (g) *biclusters* não sobrepostos e não exclusivos;
- (h) *biclusters* sobrepostos com estrutura hierárquica e
- (i) *biclusters* sobrepostos arbitrariamente.

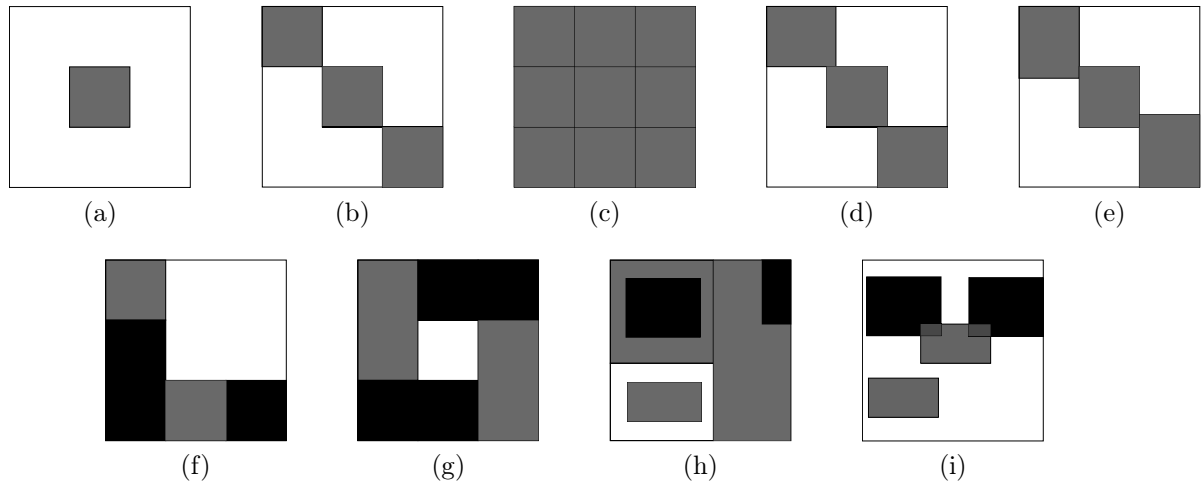


Figura 1.2: Estruturas de biclusterização.

No PBEA, o número K de *biclusters* não é definido *a priori*, o que deixa o problema mais difícil. Sua estrutura de biclusterização consiste em *biclusters* com linhas e colunas exclusivas, conforme apresentado na Figura 1.2b.

1.3 Algumas Aplicações para o PBEA

O PBEA pode ser aplicado para resolver qualquer problema que tenha como objetivo uma biclusterização com linhas e colunas exclusivas. Nesta seção, algumas aplicações para o problema serão introduzidas; dentre elas, a análise de dados biológicos — que é a principal e mais estudada aplicação — e o problema de formação de células de manufatura (aplicação proposta neste trabalho).

1.3.1 Análise de Dados Biológicos

O conceito de agrupamento de dados em *clusters* surge em muitos contextos e disciplinas diferentes. A modelagem através do PBEA produz boas soluções para um problema da biologia computacional que consiste em encontrar um bom agrupamento de genes que satisfaça a um critério de homogeneidade, ou seja, genes dentro de um agrupamento devem ter uma alta semelhança entre si e a separação entre os agrupamentos deve ser bem definida [16, 42, 61].

A análise de dados biológicos tem como entrada uma matriz gene (linha) x característica ou fenótipo (coluna). As características podem ser de indivíduos diferentes ou

até de diferentes condições experimentais do mesmo indivíduo, como por exemplo tecidos diferentes: um canceroso e um saudável. A linha de um determinado gene é chamada de padrão de expressão do referido gene. Já a coluna é chamada de perfil de expressão da condição.

A matriz gene-característica é uma poderosa fonte de informação para obter dados biológicos. O principal objetivo neste tipo de estudo é obter uma melhor compreensão das funções dos genes. Para isso, utiliza-se a ideia de que os genes com padrões de expressão semelhantes tendem a ter uma funcionalidade semelhante. Com isso, é possível identificar perfis de doenças, decifrar mecanismos de regulação do organismo, realizar genotipagem (processo laboratorial para definir genótipos), desenvolver drogas etc.

Com a matriz gene-característica cria-se um grafo bipartido onde os genes e as características serão as partes da bipartição. Isto transforma o problema do agrupamento de genes no PBEA.

1.3.2 Projeto de Redes *Multicast*

Uma sessão de *multicast* é definida como um subconjunto de clientes que requerem a mesma informação. Cada cliente pode exigir várias sessões *multicast*. A principal limitação é que uma rede de telecomunicações não pode controlar muitas sessões *multicast* ao mesmo tempo. A solução encontrada é agrupar as sessões em um número limitado de sessões.

Um exemplo de sessão *multicast* é TV sobre IP. Nele, um subconjunto de clientes pode requerer o mesmo canal ou o mesmo subconjunto de canais, durante um dado período de tempo. Faure et al. [22] mostram que uma variante do PBEA ajuda no projeto de uma rede *multicast*. Neste problema, as partes da bipartição são formadas pelos conjuntos de clientes e sessões.

1.3.3 Mineração de Dados

Dados são normalmente armazenados em uma base de dados (ou uma matriz de dados) na qual cada registro possui uma coluna de atributos [1]. Em aplicações com centenas e milhares de registros, uma alternativa para descobrir novas informações e conhecimento se dá através da biclusterização.

Imagine itens a serem vendidos em um supermercado. O objetivo é a descoberta de padrões ocultos. Descobrir compras semelhantes pode ajudar ao supermercado a melhor dispor seus produtos, identificar padrões de seus clientes, criar regras de associação que ajudem em promoções etc. Todas essas informações não seriam perceptíveis já que muitos dos padrões de compra não são aparentes.

Ao dispor de uma matriz compra-item (registro da compra x item comprado), um

algoritmo de biclusterização pode nos dar uma boa visão das relações ocultas nos dados. Diferentemente do uso de algoritmos de clusterização, em que cada compra é avaliada usando todos os seus itens, a biclusterização apresenta, neste caso, melhores resultados que uma clusterização, já que nela todos os itens de uma compra podem não estar contribuindo para esse *cluster*.

Da mesma forma que a aplicação para um supermercado, é possível fazer uma associação com classificação de documentos, onde se obtém uma matriz documentos–palavras [6]. A classificação em si ocorre de forma semelhante à aplicação de dados biológicos descrita na Seção 1.3.1.

1.3.4 Formação de Células de Manufatura

A formação de células de manufatura vem sendo utilizada nos últimos anos como uma estratégia eficiente para a redução de custos e tempo de produção em um sistema de manufatura. O PROBLEMA DE FORMAÇÃO DE CÉLULAS DE MANUFATURA (PFCM) tem por objetivo formar células (um galpão por exemplo) em um sistema de manufatura, nas quais são agrupadas máquinas dedicadas à produção de uma família de partes (ou produtos). Um sistema de manufatura pode ser representado como uma matriz parte–máquina. Nele, deseja-se obter células com um alto nível de similaridade entre as partes e máquinas que são utilizadas em sua fabricação. Como consequência disso, obtém-se uma redução no número de deslocamentos de partes entre células e um melhor aproveitamento das máquinas na célula.

No Capítulo 3, mostra-se a associação entre o PBEA e o PFCM e propõe-se um método exato para o PFCM baseado em métodos para o PBEA. Como os dois problemas possuem uma grande similaridade, os métodos utilizados em um problema podem ser aproveitados no outro.

1.4 Estrutura do Trabalho

A continuação deste trabalho está organizado da seguinte maneira:

- Capítulo 2: contém a proposta de um algoritmo exato para a resolução do PBEA;
- Capítulo 3: apresenta a aplicação do PBEA para o PFCM;
- Capítulo 4: exhibe e discute os resultados computacionais;
- Capítulo 5: apresenta conclusões e trabalhos futuros.

Capítulo 2

Proposta de algoritmo Branch-and-Cut

Neste capítulo apresenta-se a proposta para um algoritmo *Branch-and-Cut* (B&C) para o PBEA. Primeiramente, é descrito o processo de B&C na Seção 2.1 e a formulação matemática para o problema na Seção 2.2. Propõe-se um algoritmo de separação para ser incorporado no B&C na Seção 2.3. Finalmente, são apresentadas as heurísticas primais na Seção 2.4, e o pré-processamento na Seção 2.5.

2.1 Métodos Exatos

Nesta seção, são discutidos alguns conceitos gerais dos algoritmos *Branch-and-Bound* (B&B) para Programação Inteira (PI). Um problema de PI pode ser representado da seguinte forma genérica:

$$\max \quad c^T x \tag{2.1}$$

$$\text{s.a.} \quad Ax \leq b \tag{2.2}$$

$$x \in \mathbb{Z}^n. \tag{2.3}$$

em que x é um vetor de variáveis de decisão. O objetivo é maximizar a função $c^T x$, enquanto que as variáveis devem atender ao conjunto de restrições lineares (2.2). A dificuldade em resolver um problema de PI da forma (2.1) – (2.3) encontra-se nas restrições de integralidade (2.3), que tornam o problema $\mathcal{NP}$ -difícil [32].

Uma tentativa de resolver o problema acima é omitir a restrição de integralidade (2.3); dessa forma obtém-se o problema relaxado. A solução relaxada pode ser obtida em tempo polinomial através de Programação Linear (PL). Essa solução pode ser fracionária ou inteira, mas, de qualquer forma, a PL fornece um limite dual (ou limite superior) para o problema de maximização da PI. No caso de a solução ser inteira, observe que o problema de PI também foi resolvido.

O fato de que pode-se resolver um problema de PL em tempo polinomial, e que sua

resolução sempre fornece um limite dual, possibilita a utilização da PL em algoritmos de enumeração, como o B&B e suas variantes. Para se resolver um problema de PL utiliza-se, normalmente, o algoritmo *Simplex* de Dantzig [20], que é exponencial no pior caso. Entretanto, ele apresenta no caso médio resultados melhores que os demais algoritmos.

2.1.1 *Branch-and-Bound*

O *Branch-and-Bound* (B&B) é um método exato utilizado para resolver problemas de programação inteira [63]. O primeiro algoritmo B&B para PI foi proposto por Land & Doig [40] em 1960. Três anos depois, Little et al. [41] apresentaram uma aplicação computacional bem-sucedida do B&B para o PROBLEMA DO CAIXEIRO-VIAJANTE (PCV). Eles foram os primeiros a chamar o método de B&B [4].

Neste método, o problema é dividido em subproblemas com domínios disjuntos. Cada um deles é representado por um nó em uma estrutura de árvore. A cada iteração, são calculados limites primal e dual em um nó da árvore e é executada a ramificação do problema em dois novos subproblemas. Esses limites são utilizados para eliminar grandes subconjuntos de candidatos infrutíferos. Eles podem ser obtidos, por exemplo, pela solução relaxada e por uma solução heurística respectivamente.

O método começa resolvendo a relaxação de uma determinada instância do problema, por exemplo:

$$\max \quad c^T x \quad \text{s.a.} \quad x \in \mathcal{S}, \quad (2.4)$$

na qual $\mathcal{S}$ é um conjunto de restrições que podem ser representadas por $Ax \leq b$. Com a relaxação resolvida, encontra-se a solução x^* . Se x^* for uma solução inteira então o problema está resolvido. Caso contrário, o algoritmo escolhe uma variável x_j^* com valor fracionário. Como o valor de x_j^* é fracionário, faz-se:

$$\beta' = \lfloor x_j^* \rfloor \quad \text{e} \quad \beta'' = \lceil x_j^* \rceil. \quad (2.5)$$

Com isso, o algoritmo faz a ramificação e o problema original (2.4) se transforma em dois subproblemas:

$$\max \quad c^T x \quad \text{s.a.} \quad x \in \mathcal{S} \text{ e } x_j^* \leq \beta' \quad (2.6)$$

e

$$\max \quad c^T x \quad \text{s.a.} \quad x \in \mathcal{S} \text{ e } x_j^* \geq \beta''. \quad (2.7)$$

Isso se repete para cada nó da árvore, e dessa forma, a cada etapa do B&B, novos nós são

criados. A Figura 2.1 mostra o problema inicial (2.4) como a raiz da árvore. Em seguida, a mesma foi dividida em dois subproblemas (2.6) e (2.7).

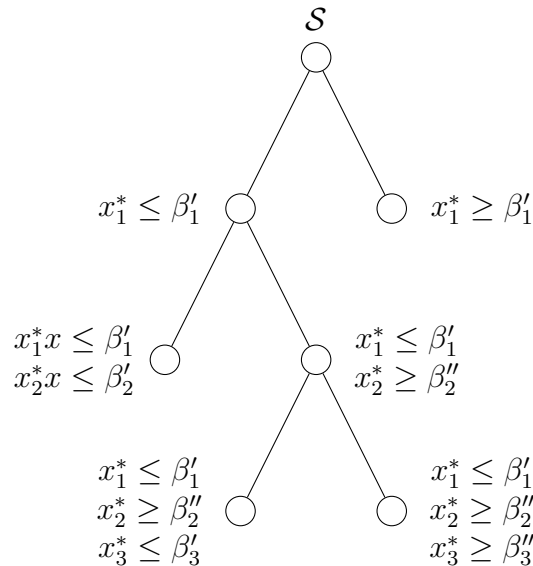


Figura 2.1: Árvore do Branch-and-Bound [4].

Se o algoritmo continuasse dessa forma, obteria apenas uma simples enumeração. O que diferencia o B&B de uma enumeração são as podas. Com as podas elimina-se uma grande quantidade de soluções sem nem mesmo avaliá-las. As podas podem ser:

- por otimalidade, quando os limites inferior e superior de um ramo são iguais, ou seja, a solução ótima deste subproblema foi encontrada, não havendo necessidade de examiná-la;
- por limite, quando o limite superior de um nó é menor (maximização) que o limite inferior do problema, e conclui-se que o ótimo do problema não se encontra neste ramo;
- por inviabilidade, quando não existe solução no ramo.

Terminada a poda, o algoritmo faz uma nova iteração, sempre seguindo essa ordem: escolhe um nó, resolve a relaxação, obtém os limites primais e duais, realiza a poda ou faz a ramificação e inicia uma nova iteração. Esse processo continua até que todos os nós sejam avaliados.

2.1.2 Método de Planos de Corte

Um método de planos de corte consiste em uma busca iterativa com o objetivo de refinar um conjunto viável ou função objetivo por meio de inequações lineares, chamadas cortes [63]. Em 1954, Dantzig et al. [19] resolveram o “problema das 49 cidades”. Este

problema consiste em resolver o PCV formado pelas 48 capitais dos EUA mais a capital do país, Washington. Dantzig et al. reduziram o problema para 42 cidades e resolveram-no usando o primeiro algoritmo de plano de corte.

Gomory [25, 26] obteve grandes resultados com algoritmos de planos de corte fracionários. Outro importante resultado teórico é o trabalho de Chvátal [17], onde propõe-se o procedimento de Chvátal-Gomory e prova-se que toda inequação válida pode ser encontrada aplicando o procedimento proposto um número finito de vezes.

A ideia de um algoritmo de plano de corte consiste em resolver a relaxação e obter uma solução x^* ; se a solução for fracionária, deve-se adicionar novas restrições (cortes) que tornem a solução fracionária inviável. Para encontrar uma nova restrição, é necessário conhecer *a priori* uma família $\mathcal{F}$ de inequações válidas para o problema. Conhecendo $\mathcal{F}$ e x^* , o algoritmo responsável por encontrar uma equação violada por x^* é chamado de *algoritmo de separação*. Se o algoritmo de separação encontra uma restrição violada, deve-se adicioná-la à formulação do problema e resolver novamente a relaxação linear. Esse ciclo continua até que se encontre uma solução inteira ou o algoritmo de separação não encontre um novo corte.

Caso o plano de corte termine sem uma solução inteira, a formulação composta pelo problema original juntamente com todas as restrições adicionadas pelo plano de corte podem servir de entrada para um algoritmo B&B.

2.1.3 *Branch-and-Cut*

Com o surgimento do B&B e dos métodos de plano de corte, o progresso dos métodos de resolução de problemas de PI ficou limitado. Até que, no meio dos anos 80, foi apresentada uma solução chamada *Branch-and-Cut* (B&C). Seu nome foi introduzido por Padberg & Rinaldi [48], apesar de já existirem trabalhos anteriores.

O B&C é uma generalização do B&B. É a união dos conceitos de B&B com plano de corte. Consiste em um B&B em que, a cada iteração, é executado um algoritmo de separação. Esse algoritmo tem como objetivo encontrar novas restrições (cortes) que são adicionadas aos subproblemas, deixando assim o problema mais restrito e, consequentemente, mais próximo da solução inteira.

Nas próximas seções, apresentar-se-á o algoritmo B&C proposto, juntamente com a explicação dos detalhes da formulação: algoritmo de separação, heurísticas primais e pré-processamento utilizado.

2.2 Formulação

A única formulação encontrada na literatura para o PBEA é a proposta por Amit [3]. Ela se baseia no fato de que grafos biclusterizados têm o P_4 como subgrafo proibido. Em

outras palavras, toda vez que houver um subgrafo induzido P_4 em G , significa que G não é um grafo biclusterizado. O P_4 é um caminho composto por 4 vértices, como mostra a Figura 2.2.

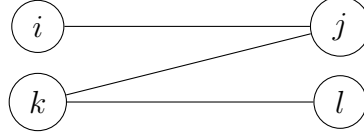


Figura 2.2: Subgrafo proibido P_4 .

O P_4 é um subgrafo proibido, uma vez que sempre será necessário realizar alguma edição de aresta para torná-lo um grafo biclusterizado. Se i, j, k e l pertencem à mesma biclique então todas as arestas entre eles devem existir, o que não acontece, já que (i, l) não existe. Caso os vértices pertençam a bicliques diferentes então pelo menos uma das arestas deve ser removida.

A formulação é definida como:

$$\min \sum_{+(ij)} x_{ij} + \sum_{-(ij)} (1 - x_{ij}) \quad (2.8)$$

$$\text{s.a. } x_{ij} \leq x_{il} + x_{kj} + x_{kl} \quad \forall i, k \in U \text{ e } j, l \in V \quad (2.9)$$

$$x_{ij} \in \{0, 1\} \quad \forall i \in U \text{ e } j \in V. \quad (2.10)$$

Na formulação, x_{ij} vale 0 se a solução contém a aresta entre os vértices i e j ; $+(ij) = \{ij | (ij) \in E\}$ e $-(ij) = \{ij | (ij) \notin E\}$. Na equação (2.8), a função objetivo indica quantas edições foram realizadas. O primeiro e o segundo somatórios representam a quantidade de remoções e adições respectivamente. A restrição (2.9) elimina os subgrafos induzidos P_4 . E por fim (2.10) é a restrição de integralidade.

Por motivos didáticos, pode-se inverter a formulação de Amit, a fim de torna-la mais intuitiva. Dessa forma, se cada variável x_{ij} for substituída por $1 - y_{ij}$, surge a seguinte formulação:

$$\min \sum_{+(ij)} (1 - y_{ij}) + \sum_{-(ij)} y_{ij} \quad (2.11)$$

$$\text{s.a. } y_{il} + y_{kj} + y_{kl} \leq 2 + y_{ij} \quad \forall i, k \in U \text{ e } j, l \in V \quad (2.12)$$

$$y_{ij} \in \{0, 1\} \quad \forall i \in U \text{ e } j \in V. \quad (2.13)$$

em que, y_{ij} vale 1 se a solução contém a aresta entre os vértices i e j . Agora, fica fácil perceber que a restrição só será violada se y_{il} , y_{kj} e y_{kl} valerem 1 e y_{ij} valer 0, formando assim um P_4 entre i, l, k e j .

Veja que o número de restrições desta formulação é $|U|^2|V|^2$. Por isso, torna-se lenta a utilização da formulação completa desde o início do algoritmo. Ela foi levada em consi-

deração, porém não foi viável. Como alternativa, propõe-se que o modelo se inicie vazio e que um algoritmo de separação acrescente as restrições de acordo com a demanda do problema, ou seja, quando elas forem violadas.

No decorrer das iterações, o algoritmo de separação encontra restrições violadas e as adiciona no modelo. Na próxima seção, será proposto um algoritmo de separação específico para o PBEA. Uma alternativa à utilização deste algoritmo é deixar que o *solver* se responsabilize por adicionar as restrições violadas através de seu próprio algoritmo de separação. Assim, neste trabalho, serão avaliadas duas formas de se adicionar as restrições no decorrer do B&C:

1. todas as restrições são adicionadas no *pool* do CPLEX e ele gerencia a adição e remoção das restrições;
2. uso do algoritmo de separação proposto.

Na primeira forma, deixa-se que o *solver* seja responsável por adicionar as restrições no decorrer do algoritmo. Para isso, é necessário adicionar todas as restrições em um *pool* chamado de *lazy constraints*. Já a segunda forma consiste em utilizar o algoritmo de separação proposto na Seção 2.3.

2.3 Algoritmo de Separação

Como o número de restrições é alto, iniciar o modelo com todas as restrições deixa o processo lento a ponto de torná-lo inviável. Por isso, propõe-se um algoritmo de separação que adiciona as restrições violadas ao modelo à medida em que forem encontradas. Para encontrar as restrições violadas de forma eficiente, o algoritmo foi construído usando o método de programação dinâmica reversa (também chamada de memoização [18]). A finalidade deste o algoritmo consiste em pesquisar na família de inequações $x_{ij} - x_{il} - x_{kj} - x_{kl} > 0$ (2.9) as restrições que estão violadas (Figura 2.3).

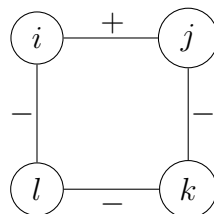


Figura 2.3: Restrição violada.

As equações abaixo mostram como foi definida a programação dinâmica.

$$d_{ij}^1 = x_{ij}^*$$

$$d_{ik}^{2,1} = \min_{l \in V} \{d_{il}^1 + d_{kl}^1\}$$

$$\begin{aligned}
l_{ik}^2 &= \arg \min_{l \in V} \{d_{il}^1 + d_{kl}^1\} \\
d_{ik}^{2,2} &= \min_{l \in V \setminus \{l_{ik}^2\}} \{d_{il}^1 + d_{kl}^1\} \\
d_{ikj}^2 &= \begin{cases} d_{ik}^{2,1}, & \text{se } j \neq l_{ik}^2 \\ d_{ik}^{2,2}, & \text{se } j = l_{ik}^2 \end{cases} \\
d_{ij}^3 &= \min_{k \in U \setminus \{i\}} \{d_{ikj}^2 + d_{kj}^1\} \\
d_{ij}^4 &= \{d_{ij}^1 - d_{ij}^3\}
\end{aligned}$$

O algoritmo de separação está representado pelo Algoritmo 2.1, onde se procura a restrição mais violada para cada par de vértices (i, j) , tendo como entrada a relaxação linear. Veja que, ao calcular o valor de uma restrição (Figura 2.3), calcula-se o valor de uma “sub-restrição” P_3 . O algoritmo de separação proposto possui ordem $O(|U|^2|V|^2)$. No entanto, cada sub-restrição é calculada apenas uma vez pelo Algoritmo 2.2. Dessa forma, em uma nova restrição, uma parte do cálculo já estará feito, aumentando assim o desempenho do método.

O principal problema em encontrar as “sub-restrições” P_3 através da programação dinâmica se deve ao fato de que para encontrar d_{ij}^3 precisa-se de d_{ikj}^2 . Com isso, existe a possibilidade de o algoritmo retornar l e j iguais. Dessa forma, deve-se retornar sempre duas opções, a menor e a segunda menor distância, além do próprio vértice l . Assim, cabe ao Algoritmo 2.2 avaliar se os vértices são os mesmos.

2.4 Heurísticas Primais

Uma forma de acelerar o processo do *branch* é encontrar um limite primal para um determinado ramo da árvore de soluções. Para isso, basta encontrar uma solução viável dentro do ramo. Um modo de encontrar uma solução é através de uma heurística primal.

Nesta seção, serão apresentadas as duas heurísticas utilizadas neste trabalho. A primeira consiste em um algoritmo 11-aproximado proposto por Amit [3], e a segunda é um algoritmo randômico 4-aproximado proposto por Guo et al. [28].

2.4.1 Algoritmo Determinístico 11-aproximado de Amit [3]

Dizer que um algoritmo A é α -aproximado significa (no caso de minimizar) que para qualquer instância I pertencente ao domínio do problema, o valor de retorno do algoritmo, $A(I)$, satisfaz $A(I) \leq \alpha \times OPT(I)$, em que $OPT(I)$ é o valor ótimo da instância.

No caso do algoritmo 11-aproximado de Amit [3], a solução do algoritmo é menor ou igual a 11 vezes o valor do ótimo. A demonstração desta aproximação pode ser encontrada

Algoritmo 2.1 Algoritmo de Separação

```

1: função ALGORITMO DE SEPARAÇÃO( $x^*$ ) ▷ Relaxação Linear como entrada
2:    $L \leftarrow \emptyset$ 
3:   para  $i \in U$  faça
4:     para  $j \in V$  faça
5:        $d_{ij}^1 \leftarrow x_{ij}^1$  ▷ Inicializando as variáveis com a relaxação linear
6:        $d_{ij}^{2,1}.val \leftarrow \infty$  ▷ Inicializando as variáveis que serão utilizadas em
       BUSCA_P3
7:     fim para
8:   fim para
9:   para  $i \in U$  faça
10:    para  $j \in V$  faça
11:       $d_{ij}^3.val \leftarrow \infty$ 
12:      para  $k \in U \setminus \{i\}$  faça
13:         $(d_{ikj}^2.val, l) \leftarrow \text{BUSCA\_P3}(i, j, k)$  ▷ Encontra o  $P_3$  entre  $i$  e  $k$  passando
        por  $l \neq j$ 
14:         $temp \leftarrow d_{ikj}^2.val + d_{kj}^1$ 
15:        se  $temp < d_{ij}^3.val$  então
16:           $d_{ij}^3.val \leftarrow temp$ 
17:           $d_{ij}^3.k \leftarrow k$ 
18:           $d_{ikj}^2.l \leftarrow l$ 
19:        fim se
20:      fim para
21:      se  $d_{ij}^1 - d_{ij}^3.val > 0$  então ▷ Teste de violação
22:         $L \leftarrow L \cup \{i, j, d_{ij}^3.k, d_{ikj}^2.l\}$ 
23:      fim se
24:    fim para
25:  fim para
26:  retorne  $L$  ▷ Retorna uma lista de cortes  $L$ 
27: fim função

```

Algoritmo 2.2 Busca P_3

```

1: função BUSCA_P3( $i, j, k$ )                                ▷ Entrada vértices  $i, j$  e  $k$ 
2:   se  $d_{ik}^{2,1}.val = \infty$  então                          ▷ Verifica se já foi calculado
3:      $temp \leftarrow \infty$ 
4:     para  $l \in V$  faça                                       ▷ Calcula menor  $P_3$ 
5:        $temp \leftarrow d_{il}^1 + d_{kl}^1$ 
6:       se  $temp < d_{ik}^{2,1}.val$  então
7:          $d_{ik}^{2,1}.val \leftarrow temp$ 
8:          $d_{ik}^{2,1}.l \leftarrow l$ 
9:       fim se
10:    fim para                                                 ▷ Calcula segundo menor  $P_3$ 
11:     $d_{ik}^{2,2}.val \leftarrow \infty$ 
12:    para  $l \in V \setminus \{d_{ik}^{2,1}.l\}$  faça
13:       $temp \leftarrow d_{il}^1 + d_{kl}^1$ 
14:      se  $temp < d_{ik}^{2,2}.val$  então
15:         $d_{ik}^{2,2}.val \leftarrow temp$ 
16:         $d_{ik}^{2,2}.l \leftarrow l$ 
17:      fim se
18:    fim para
19:  fim se
20:  se  $j \neq d_{ik}^{2,1}.l$  então
21:    retorne  $(d_{ik}^{2,1}.val, d_{ik}^{2,1}.l)$ 
22:  senão
23:    retorne  $(d_{ik}^{2,2}.val, d_{ik}^{2,2}.l)$ 
24:  fim se
25: fim função

```

no próprio trabalho de Amit.

A primeira etapa do algoritmo consiste em resolver o problema relaxado, ou seja, a integralidade das variáveis passa de $x_{ij} \in \{0, 1\}$ para $0 \leq x_{ij} \leq 1$. O algoritmo se baseia em determinar as bicliques através de vizinhanças definidas pela função distância d , que nada mais é do que a própria relaxação linear. Com a PL resolvida, calcula-se $S = V \cup U$ e repetem-se os passos abaixo.

- Passo 1:

- Selecione uma aresta de $\{(u, v) : d(u, v) < \frac{1}{11}\}$, caso não exista esta aresta vá para o passo 3;
- calcule $N_u = \{i \in U, i \neq u \mid d(i, u) \leq \frac{5}{11}\}$ e $N_v = \{i \in V, i \neq v \mid d(i, v) \leq \frac{5}{11}\}$;
- calcule $N'_u = \{i \in U, i \neq u \mid d(i, u) \leq \frac{3}{11}\}$ e $N'_v = \{i \in V, i \neq v \mid d(i, v) \leq \frac{3}{11}\}$;
- calcule α_u , a média das distâncias entre u e os vértices de N_u . Se $N_u = \emptyset$ então faça $\alpha_u = 1$.
- calcule α_v , a média das distâncias entre v e os vértices de N_v . Se $N_v = \emptyset$ então faça $\alpha_v = 1$.

- Passo 2:

$$\text{– Seja } B = \begin{cases} \{u, v\}, & \text{se } \alpha_u, \alpha_v > \frac{3}{11} \\ & \text{ou se } \frac{1}{11} < \alpha_u \leq \frac{3}{11} \text{ e } \alpha_v > \frac{3}{11} \\ & \text{ou se } \alpha_u > \frac{3}{11} \text{ e } \frac{1}{11} < \alpha_v \leq \frac{3}{11}, \\ \{u, v\} \cup N_u \cup N_v, & \text{se } \alpha_u, \alpha_v \leq \frac{3}{11}, \\ \{u, v\} \cup N_u \cup N'_v, & \text{se } \alpha_u \leq \frac{1}{11} \text{ e } \alpha_v > \frac{3}{11}, \\ \{u, v\} \cup N'_u \cup N_v, & \text{se } \alpha_u > \frac{3}{11} \text{ e } \alpha_v \leq \frac{1}{11}. \end{cases}$$

- Retorne B como uma biclique, faça $S = S \setminus B$ e volte ao passo 1.

- Passo 3: Quando não houver arestas com distância menor que $\frac{1}{11}$, retorne cada vértice de S como uma biclique independente (também chamado de *singleton*).

O algoritmo de Amit foi o primeiro algoritmo aproximativo aplicado ao PBEA. Apesar de ter um desempenho favorecido na aplicação do B&C — já que no B&C, cada nó executada o algoritmo *Simplex* que é necessário na primeira etapa do algoritmo — ele não apresenta uma aproximação tão boa quanto o algoritmo apresentado na próxima seção.

2.4.2 Algoritmo Randômico 4-aproximado de Guo et al. [28]

Da forma similar ao algoritmo determinístico, um algoritmo randômico R é β -aproximado (no caso de minimização) se para qualquer instância I pertencente ao domínio do problema, o valor esperado do algoritmo satisfaz $\mathbb{E}(R(I)) \leq \beta \times OPT(I)$ em que $OPT(I)$ é o valor ótimo da instância. APPROXBICLUSTER é o algoritmo randômico 4-aproximado proposto por Guo et al. [28] com complexidade $O(n^2)$ para o PBEA.

Algoritmo 2.3 APPROXBICLUSTER

```

1: função APPROXBICLUSTER( $G = (V_1, V_2, E)$ )
2:    $G' \leftarrow (\emptyset, \emptyset, \emptyset)$ 
3:   enquanto  $V_1 \cup V_2 \neq \emptyset$  faça
4:     Seleciona aleatoriamente um vértice pivô  $i \in V_1 \cup V_2$ 
5:      $C \leftarrow N[i]$ 
6:     para  $j \in N^2(i)$  faça                                 $\triangleright$  quais vértices  $j$  estão na mesma biclique que  $i$ ?
7:       se  $N(j) = N(i)$  então
8:         Adiciona  $j$  em  $C$ 
9:       senão
10:        Adiciona  $j$  em  $C$  com probabilidade  $1/2$ 
11:     fim se
12:   fim para
13:   Transforma  $G[C]$  em uma biclique
14:    $G' \leftarrow G' \cup G[C]$                                  $\triangleright$  adiciona  $G[C]$  como um novo componente de  $G'$ 
15:    $G \leftarrow G[(V_1 \cup V_2) \setminus C]$                          $\triangleright$  remove  $C$  de  $G$ 
16: fim enquanto
17: retorne Conjunto de modificações de arestas de  $G$  para  $G'$ 
18: fim função

```

O algoritmo 2.3 apresenta o pseudocódigo do APPROXBICLUSTER. A ideia consiste em escolher, aleatoriamente, um vértice i como pivô, e a partir dos conjuntos $N(i)$ e $N^2(i)$ (vizinhos de segundo grau), determinar sua biclique.

Uma vez que existe uma restrição violada entre os vértices i, j, l e k e um deles seja escolhido como pivô, o algoritmo adiciona/remove uma das quatro arestas com a mesma probabilidade — apesar de não ser explícito — como mostra a Tabela 2.1, em que E_{ij} denota o evento de modificar a aresta (i, j) .

pivô	$\Pr(E_{ij})$	$\Pr(E_{il})$	$\Pr(E_{kj})$	$\Pr(E_{kl})$
i	0	0	$1/2$	$1/2$
j	0	$1/2$	0	$1/2$
k	$1/2$	$1/2$	0	0
l	$1/2$	0	$1/2$	0
$i \vee j \vee k \vee l$	$1/4$	$1/4$	$1/4$	$1/4$

Tabela 2.1: Probabilidade de modificação de cada aresta [28].

2.4.3 Comparação Amit $\times$ Guo et al.

Teoricamente, assegura-se que no caso médio o algoritmo de Guo et al. provê resultados melhores que o de Amit. Sousa Filho et al. [57] mostram isso através de testes empíricos com diferentes tipos de instância. O resultado mostrou que, realmente, o algoritmo de Guo et al. apresenta um desempenho melhor que o de Amit, chegando até mesmo a igualar o resultado em algumas instâncias com a meta-heurística proposta pelos autores. Por isso, nos testes computacionais deste trabalho, só é levada em consideração a heurística de Guo et al.

2.5 Pré-processamento

Nesta seção, apresenta-se um pré-processamento que fixará e/ou gerará novas restrições para o problema. Como base do pré-processamento, propõe-se o Teorema 2.5.1. E a partir dele, serão geradas novas restrições que serão adicionadas ao B&C.

Teorema 2.5.1. *Sejam a e b vértices quaisquer em um grafo bipartido $G(V, U, E)$, e seja $d(a, b)$ a distância entre eles. Se $d(a, b) \geq 4$, então existe uma solução ótima na qual a e b pertencem a bicliques distintas.*

PROVA. Deve-se provar que sempre será mais custoso manter a e b na mesma biclique, quando $d(a, b) \geq 4$, do que mantê-los em bicliques distintas. Para tal, serão definidas duas funções $rem(X, Y)$ e $adc(X, Y)$.

Sejam $X \subseteq V$ e $Y \subseteq U$. A função $rem(X, Y)$ representa o custo de remover todas as arestas entre os conjuntos de vértices X e Y . A função $adc(X, Y)$ representa o custo de adicionar todas as arestas entre X e Y que faltam para criar uma biclique $B = \{X, Y\}$. Com isso, definem-se $N(a)$ como a vizinhança de a , e $N^2(a) = \{v \in V | d(a, v) = 2\}$. Para os casos em que $d(a, b) = \infty$, ou seja, não existe um caminho entre a e b , eles já estão em bicliques diferentes. Restam então dois possíveis casos: a e b na mesma parte (Caso 1) e em partes diferentes (Caso 2). O caso em que a e b estão na mesma parte é dividido em dois sub-casos: $d(a, b) > 4$ (Caso 1a) e $d(a, b) = 4$ (Caso 1b).

Nesta prova serão considerados os conjuntos $N_a \subseteq N(a)$, $N_b \subseteq N(b)$, $N_a^2 \subseteq N^2(a)$ e $N_b^2 \subseteq N^2(b)$.

Caso 1a:

Neste caso temos que a e b pertencem à mesma parte e $N^2(a) \cap N^2(b) = \emptyset$, ou seja, $d(a, b) > 4$. Este caso é representado pela Figura 2.4, em que P representa um caminho qualquer entre $N^2(a)$ e $N^2(b)$.

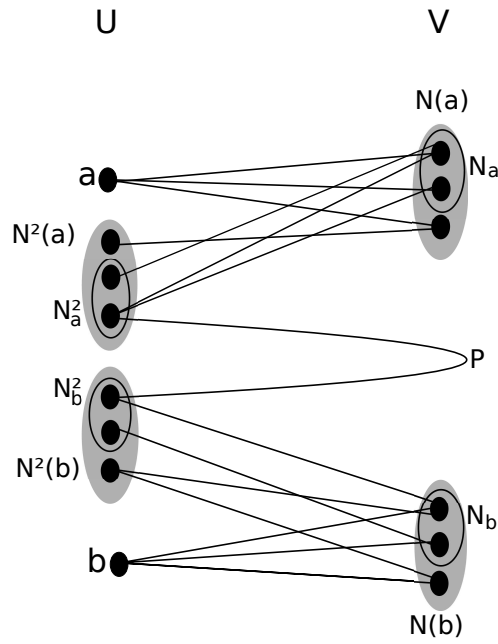


Figura 2.4: Caso 1a.

- A biclique $B = \{\{a, b\} \cup N_a^2 \cup N_b^2, N_a \cup N_b\}$ tem o custo:

$$\begin{aligned}
 & \text{adc}(a, N_b) + \text{rem}(a, N(a) \setminus N_a) + \\
 & + \text{adc}(N_a^2, N_a) + \text{adc}(N_a^2, N_b) + \text{rem}(N_a^2, N(a) \setminus N_a) + \text{rem}(N_a^2, V \setminus (N(a) \cup N(b))) + \\
 & + \text{adc}(N_b^2, N_b) + \text{adc}(N_b^2, N_a) + \text{rem}(N_b^2, N(a) \setminus N_a) + \text{rem}(N_b^2, V \setminus (N(b) \cup N(a))) + \\
 & + \text{adc}(b, N_a) + \text{rem}(b, N(b) \setminus N_b).
 \end{aligned}$$

Porém, a criação das bicliques $B_1 = \{\{a\} \cup N_a^2, N_a\}$ e $B_2 = \{\{b\} \cup N_b^2, N_b\}$ tem custo igual a:

$$\begin{aligned}
 & \text{rem}(a, N(a) \setminus N_a) + \\
 & + \text{adc}(N_a^2, N_a) + \text{rem}(N_a^2, N(a) \setminus N_a) + \text{rem}(N_a^2, V \setminus (N(a) \cup N(b))) + \\
 & + \text{adc}(N_b^2, N_b) + \text{rem}(N_b^2, N(a) \setminus N_a) + \text{rem}(N_b^2, V \setminus (N(b) \cup N(a))) + \\
 & + \text{rem}(b, N(b) \setminus N_b).
 \end{aligned}$$

Logo, a separação em duas bicliques tem um custo menor.

Caso 1b:

Neste caso, tem-se que a e b pertencem à mesma parte e que $N^2(a) \cap N^2(b) = C$, ou seja, $d(a, b) = 4$. Este caso é representado pela Figura 2.5. Nele serão considerados os conjuntos $N_{ac}^2 \subseteq N^2(a) \setminus C$, $N_{bc}^2 \subseteq N^2(b) \setminus C$ e $C' \subseteq C$.

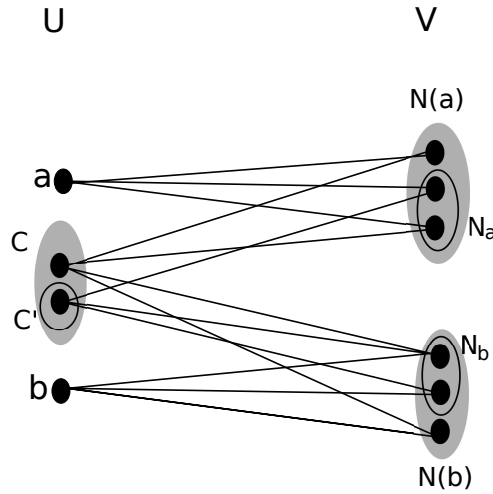


Figura 2.5: Caso 1b.

- A biclique $B = \{\{a, b\} \cup C', N_a \cup N_b\}$ tem o custo:

$$\begin{aligned}
 Custo1 = & \text{adc}(b, N_a) + \text{adc}(a, N_b) + \text{rem}(N_a, N^2(a) \setminus C) + \\
 & + \text{rem}(N_b, N^2(b) \setminus C) + \text{adc}(C', N_a) + \text{adc}(C', N_b) + \\
 & + \text{rem}(a, N(a) \setminus N_a) + \text{rem}(b, N(b) \setminus N_b) + \\
 & + \text{rem}(C', N(a) \setminus N_a) + \text{rem}(C', N(b) \setminus N_b).
 \end{aligned}$$

Por outro lado, sem perda de generalidade, se $|N_b| \leq |N_a|$, então a criação das bicliques $B_1 = \{a \cup C', N_a \cup N_b\}$ e $B_2 = \{b, \emptyset\}$ teriam o custo:

$$\begin{aligned}
 Custo2 = & \text{rem}(b, N(b)) + \text{adc}(a, N_b) + \text{rem}(N_a, N^2(a) \setminus C) + \\
 & + \text{rem}(N_b, N^2(b) \setminus C) + \text{adc}(C', N_a) + \text{adc}(C', N_b) + \\
 & + \text{rem}(a, N(a) \setminus N_a) + \text{rem}(C', N(a) \setminus N_a) + \text{rem}(C', N(b) \setminus N_b).
 \end{aligned}$$

Calculando a diferença entre os custos:

$$\begin{aligned}
 Custo1 - Custo2 = & \text{adc}(b, N_a) + \text{rem}(b, N(b) \setminus N_b) - \text{rem}(b, N(b)) = \\
 = & |N_a| + |N(b)| - |N_b| - |N(b)| = |N_a| - |N_b|.
 \end{aligned}$$

Como $|N_b| \leq |N_a|$, a separação em duas bicliques tem custo menor ou igual.

Caso 2:

Neste caso, a e b pertencem a partes diferentes. Sua representação é dada pela Figura 2.6.

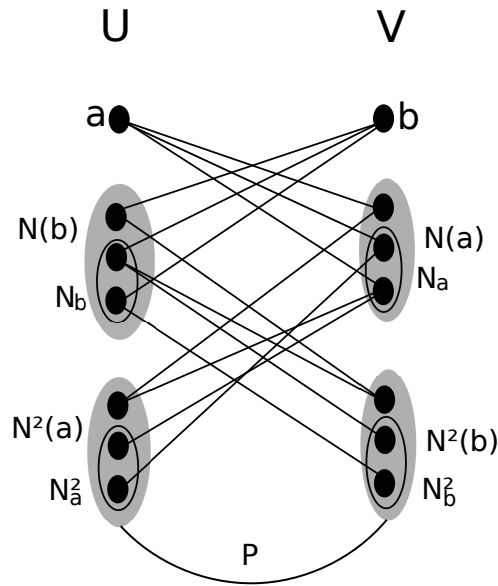


Figura 2.6: Caso 2.

- A biclique $B = \{a \cup N_b \cup N_a^2, b \cup N_a \cup N_b^2\}$ tem o custo:

$$\begin{aligned}
 & \text{adc}(a, b) + \text{adc}(N_a, N_b) + \text{adc}(N_a, N_a^2) + \text{adc}(N_b, N_b^2) + \\
 & + \text{adc}(N_a^2, N_b^2) + \text{rem}(N_a^2, V \setminus N(a)) + \text{rem}(N_b^2, U \setminus N(b)) + \\
 & + \text{rem}(a, N(a) \setminus N_a) + \text{rem}(b, N(b) \setminus N_b) + \\
 & + \text{rem}(N_a, N^2(a) \setminus N_a^2) + \text{rem}(N_b, N^2(b) \setminus N_b^2).
 \end{aligned}$$

Porém separar em $B_1 = \{a \cup N_a^2, N_a\}$ e $\{N_b, \{b\} \cup N_b^2\}$ possui um custo menor:

$$\begin{aligned}
 & \text{adc}(N_a, N_a^2) + \text{adc}(N_b, N_b^2) + \text{rem}(N_a^2, V \setminus N(a)) + \text{rem}(N_b^2, U \setminus N(b)) + \\
 & + \text{rem}(a, N(a) \setminus N_a) + \text{rem}(b, N(b) \setminus N_b) + \\
 & + \text{rem}(N_a, N^2(a) \setminus N_a^2) + \text{rem}(N_b, N^2(b) \setminus N_b^2).
 \end{aligned}$$

Logo, a separação em duas bicliques tem um custo menor. ■

Com este resultado, pode-se fixar vértices em bicliques diferentes, antes de executar o B&C. Na prática, utiliza-se o algoritmo de Dijkstra para calcular a distância entre cada par de vértices. Com a distância atendendo o teorema, fixam-se variáveis de decisão ou criam-se cortes. Caso u e v estejam em partes diferentes, simplesmente fixa-se a variável $y_{uv} = 0$. Já no caso em que u e v estejam na mesma parte U , serão criados os cortes

$y_{uw} + y_{vw} \leq 1, \forall w \in V$. Todo o processo pode ser visto no Algoritmo 2.4.

Como dito na Seção 2.2, para utilizar o algoritmo de separação do CPLEX basta adicionar todas as restrições de proibição de P_4 em um *pool* chamado de *lazy constraints*. No caso dos cortes gerados pelo pré-processamento, eles são adicionados em um outro *pool* chamado de *user cuts*.

Algoritmo 2.4 Pré-Processamento

```

1: função PRE-PROCESSAMENTO( $G = (U, V, E)$ )
2:   Utilize o algoritmo de Dijkstra em  $G$  para criar  $d$ 
3:   para  $(i, j) \in \{i, j \in U \mid i \neq j \text{ e } d(i, j) \geq 4\}$  faça
4:     para  $w \in V$  faça
5:       Crie o corte  $y_{iw} + y_{jw} \leq 1$ 
6:     fim para
7:   fim para
8:   para  $(i, j) \in \{i, j \in V \mid i \neq j \text{ e } d(i, j) \geq 4\}$  faça
9:     para  $w \in U$  faça
10:      Crie o corte  $y_{wi} + y_{wj} \leq 1$ 
11:     fim para
12:   fim para
13:   para  $(i, j) \in \{i \in U, j \in V \mid d(i, j) \geq 4\}$  faça
14:     Fixe  $y_{ij} = 0$ 
15:   fim para
16: fim função

```

2.6 Instâncias

Como o PBEA ainda é pouco explorado, não existem instâncias criadas na literatura. Dessa forma propõe-se a criação de instâncias — grafos bipartidos aleatórios — utilizando o modelo $\mathbb{G}(m, n, p)$, também conhecido como modelo binomial. Este modelo é um caso específico do proposto por Gilbert [24] para grafos bipartidos. Com ele, são construídos grafos bipartidos $G(U, V, E)$ em que $|U| = m$, $|V| = n$ e cada aresta de E é incluída no grafo com probabilidade p , independentemente das outras arestas. Neste modelo, o número de arestas de um grafo G , $E_G = |E(\mathbb{G}(m, n, p))|$, é uma variável aleatória com esperança $\mathbb{E}(E_G) = mnp$. Um exemplo de como gerar um grafo $\mathbb{G}(m, n, p)$ pode ser visto no Algoritmo 2.5.

Algoritmo 2.5 Algoritmo para Criação de Instâncias

```
1: função CRIAINSTANCIAS(Inteiro  $m$ , Inteiro  $n$ , Real  $p$ )
2:   Cria Matrix  $M_{m \times n}$ 
3:   para  $i = 1 \rightarrow m$  faça
4:     para  $j = 1 \rightarrow n$  faça
5:        $rand \leftarrow RAND\_UNIF(0, 1)$   $\triangleright$  Gera um número pseudo-aleatório entre 0
        e 1 com distribuição uniforme
6:       se  $rand < p$  então
7:          $M[i][j] \leftarrow 1$ 
8:       senão
9:          $M[i][j] \leftarrow 0$ 
10:      fim se
11:    fim para
12:  fim para
13:  retorne  $M$ 
14: fim função
```

Capítulo 3

Aplicação para o Problema de Formação de Células de Manufatura

Neste capítulo apresentaremos uma aplicação do PBEA para o PROBLEMA DE FORMAÇÃO DE CÉLULAS DE MANUFATURA (PFCM). Faremos uma apresentação do problema na seção 3.1 juntamente com um apanhado geral de sua literatura. Na seção 3.2 mostraremos sua semelhança com o PBEA e proporemos um algoritmo exato para resolvê-lo baseado no próprio PBEA. Por fim, proporemos uma nova formulação linear fracionária para o PFCM na Seção 3.3.

3.1 Problema de Formação de Células de Manufatura

Uma *manufatura celular* é uma aplicação do conceito de *Tecnologia de Grupo*. Ela se propõe a identificar e agrupar partes similares com o objetivo de otimizar o processo de manufatura. Seu conceito foi proposto originalmente por Flanders [23] em 1924, e difundido por Mitrofanov [45] em 1966.

Em 1971, Burbidge [9] propôs uma das primeiras técnicas para a criação de um sistema de manufatura celular, a *análise de fluxo de produção*. A partir deste trabalho, o modelo de manufatura foi utilizado como uma estratégia eficiente com grandes vantagens de utilização. As principais vantagens de uma indústria adotar a formação de células de manufatura são: redução do tempo necessário para preparar a máquina para a produção; redução do tempo de fabricação; redução nos custos de transporte das matérias-primas; simplificação do fluxo de produção; e melhora na qualidade dos produtos.

Como dito anteriormente na Seção 1.3.4, pode-se representar um sistema de manufatura por uma matriz binária parte-máquina em que seus elementos possuem valor 1 se a parte é fabricada pela máquina e 0 caso contrário. O objetivo do PFCM é formar células nas quais são agrupadas máquinas dedicadas à produção de uma família de partes. E assim, obter células com um alto nível de similaridade entre as partes e entre as máquinas

utilizadas em sua fabricação. Como consequência disso, obtém-se uma redução no número de deslocamentos de partes entre células e um melhor aproveitamento das máquinas na célula.

Para se obter essas células, faz-se permutações entre as linhas e colunas da matriz a fim de formar submatrizes de 1's na diagonal principal e submatrizes de 0's fora. A Figura 3.1 mostra um exemplo de uma célula de manufatura à esquerda e a clusterização formada pelas permutações à direita.

	M_1	M_2	M_3	M_4	M_5		M_2	M_3	M_5	M_1	M_4	
P_1	0	1	1	0	1	$\Rightarrow$	P_1	1	1	1	0	0
P_2	1	0	0	1	0		P_3	1	1	0	0	0
P_3	0	1	1	0	0		P_7	0	1	1	0	0
P_4	1	0	0	1	0		P_2	0	0	0	1	1
P_5	1	0	0	0	1		P_4	0	0	0	1	1
P_6	1	0	1	1	0		P_5	0	0	1	1	0
P_7	0	0	1	0	1		P_6	0	1	0	1	1

Figura 3.1: Exemplo de uma resolução do PFCM.

3.1.1 Literatura do PFCM

O PFCM é uma problema que tem sido muito explorado, principalmente nas últimas três décadas. Não apenas na forma que em foi apresentado, o PFCM pode ser generalizado e acrescido de outros fatores que o tornam mais realista. Por exemplo, acrescentar ao modelo novas restrições, como: operações que são requisitadas, tempo de preparo e processamento de cada operação. Em 1998, Selim et al. [56] fizeram uma revisão da literatura tendo como foco principal o problema generalizado. Da mesma forma, Papaioannou & Wilson [50] continuaram a revisão a partir de 1998, destacando o problema generalizado. Nestes trabalhos, são apresentadas as principais técnicas empregadas ao longo dos anos, dentre elas:

- algoritmos descritivos;
- técnicas de clusterização;
- técnicas de particionamento em grafos;
- inteligência artificial e meta-heurísticas, como:
 - lógica *fuzzy*;
 - redes neurais;
 - *simulated annealing*;

- algoritmos evolucionários;
- colônia de formigas;
- busca tabu.

Com relação à variante estudada neste trabalho e apresentada anteriormente, os primeiros algoritmos nessa linha foram o ZODIAC e o GRAPHICS [14, 58]. Com relação a meta-heurísticas, o principal trabalho foi publicado em 2004 por Gonçalves & Resende [27]. Nele, apresenta-se uma técnica que combina algoritmo genético com uma heurística de busca local. Alguns de seus resultados ainda são os melhores ainda hoje, juntamente com os trabalhos de Wu et al. [65] e de Pailla et al. [49].

No que diz respeito às formulações matemáticas do problema, com exceção de apenas uma, todas as formulações encontradas são não lineares. A única formulação linear encontrada na literatura é a de Boctor [7]; porém, ela possui uma função objetivo não linear que foi linearizada. Além disso, ela considera apenas os 1's fora das células. Por esses motivos, este trabalho não será comparado com o método proposto. Na Seção 3.1.3, encontra-se uma discussão mais aprofundada sobre as funções objetivo. Vale ressaltar que não foi encontrado nenhum trabalho que resolvesse de forma exata as instâncias da literatura.

3.1.2 Tamanho da Célula

Alguns trabalhos definem um valor máximo para o tamanho da célula. Sua finalidade é manter as operações sob controle ou, no pior caso, limitar o espaço disponível. A grande maioria dos trabalhos proíbe toda célula que possuir menos que duas partes ou duas máquinas; a esta célula dá-se o nome de *singleton*. No entanto, não existe um consenso com relação ao tamanho das células. ZODIAC e GRAPHICS, por exemplo, proíbem *singletons*, assim como Gonçalves & Resende [27]. Outros trabalhos não consideram nenhuma limitação de tamanho mínimo, permitindo até a existência de células vazias, como é o caso do trabalho de Pailla et al. [49]. Há autores que ficam no meio termo, como por exemplo James et al. [31], que permitem *singletons* e proíbem células vazias.

O principal problema dessa divergência é que muitos trabalhos não comentam qual modelo estão utilizando e até mesmo comparam com trabalhos que utilizam modelos diferentes. Neste trabalho resolve-se o PFCM das duas maneiras mais encontradas na literatura:

1. sem restrições *singleton*, permitindo células vazias;
2. com restrições *singleton* (células com tamanho mínimo 2×2).

3.1.3 Funções Objetivo do PFCM

No decorrer dos anos, diferentes formas de avaliar uma solução do problema foram propostas, todas sempre se baseando nas seguintes definições:

- N_1 é o número total de 1's na matriz, ou seja, o número de relações parte-máquina;
- N_1^{out} é o número total de 1's fora dos *biclusters*, ou seja, o número de partes que são utilizadas em máquinas de *biclusters* diferentes;
- N_0^{in} é o número total de 0's dentro dos *biclusters*, ou seja, o número de máquinas subutilizadas no *bicluster*.

No primeiros trabalhos sobre o tema, como a formulação de Boctor [7], apenas N_1^{out} era utilizado. Isto ocorre até o trabalho de Chandrasekharan & Rajagopalan [12], na qual os autores argumentam que, além de minimizar N_1^{out} , também era preciso minimizar N_0^{in} . A abordagem utilizada nos trabalhos atuais é a eficácia de agrupamento μ , proposta por Kumar & Chandrasekharan [35]. Pode-se defini-la como:

$$\mu = \frac{N_1 - N_1^{out}}{N_1 + N_0^{in}}. \quad (3.1)$$

Observe que o objetivo é maximizar a eficácia; assim, quanto mais próximo de 1 for este índice, melhor será o agrupamento.

3.2 Algoritmo Exato para o PFCM

Nesta seção, propõe-se um algoritmo exato para o PFCM baseado no PBEA. Como dito na Seção 3.1.2, a literatura do PFCM restringe o problema a um tamanho mínimo das células. Com isso, será definida uma nova variante do PBEA que será chamada de PROBLEMA DE BICLUSTERIZAÇÃO POR EDIÇÃO DE ARESTAS COM SINGLETON (PBEAS).

O PBEAS consiste na adaptação PBEA para poder se adequar com as restrições do PFCM. A ideia é adicionar restrições que limitem as bicliques a um tamanho mínimo. O PBEAS é definido com a seguinte formulação matemática:

$$\min \quad \sum_{+(ij)} (1 - y_{ij}) + \sum_{-(ij)} y_{ij} \quad (3.2)$$

$$\text{s.a} \quad y_{il} + y_{kj} + y_{kl} \leq 2 + y_{ij} \quad \forall i, k \in U \text{ e } j, l \in V \quad (3.3)$$

$$\sum_{j \in V} y_{ij} \geq s_c \quad \forall i \in U \quad (3.4)$$

$$\sum_{i \in U} y_{ij} \geq s_l \quad \forall j \in V \quad (3.5)$$

$$y_{ij} \in \{0, 1\} \quad \forall i \in U \text{ e } j \in V. \quad (3.6)$$

em que s_l representa a quantidade mínima de vértices da partição U em cada biclique e s_c a quantidade mínima de vértices da partição V em cada biclique. Dessa forma, com as restrições (3.4) e (3.5), será possível relacionar as bicliques do PBEA com as células formadas pelo PFCM.

Antes da apresentação do algoritmo exato e de sua prova, dois lemas serão provados. O primeiro mostra que uma solução do PBEAS é uma solução viável do PFCM (Lema 3.2.1). Já o segundo define um Limite Inferior (LI) e um Limite Superior (LS) para o PFCM (Lema 3.2.2).

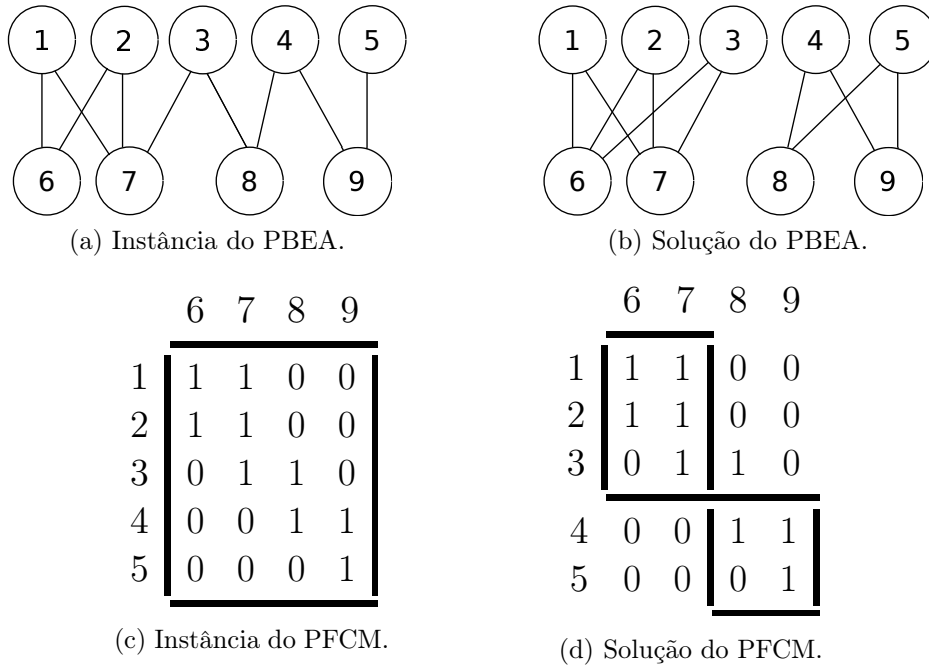


Figura 3.2: Exemplo de transformação PBEA $\leftrightarrow$ PFCM.

Lema 3.2.1. *No PBEAS e no PFCM com restrições de mesmo tamanho de bicluster/células, seus espaços de solução são isomórficos.*

PROVA. Para provar este lema, basta mostrar uma bijeção entre os conjuntos. Seja I uma instância do PBEA. Ela também corresponde a uma instância viável para o PFCM. As Figuras 3.2a e 3.2c mostram um exemplo de uma instância I representada por um grafo bipartido e por uma matriz parte-máquina.

Sejam também B e C os espaços de solução para o PBEA e PFCM, respectivamente, para a instância I . Define-se $f : B \rightarrow C$, como a transformação de uma solução de B em uma solução em C . Para cada vértice da partição U do grafo bipartido cria-se uma parte na matriz da célula. E para cada vértice da partição V cria-se uma máquina na matriz. A criação das células é feita a partir da solução do *bicluster*, como mostram as Figuras 3.2b e 3.2d. Como isso, cada solução de B é levada em apenas uma solução de C .

Da mesma forma, define-se $g : C \rightarrow B$ como transformação inversa de f que leva cada solução de C em apenas uma solução de B . Para isso, dada uma solução de C , cria-se um

vértice em U para cada parte e um vértice em V para cada máquina. Como $f : B \rightarrow C$ e $g : C \rightarrow B$ são duas funções injetoras, então obtém-se uma bijeção entre B e C , e consequentemente, B e C são isomórficos. ■

Como consequência do Lema 3.2.1, uma solução viável do PBEA pode ser representada como uma solução viável para o PFCM. A Figura 3.2 mostra uma instância para o PBEA e sua transformação para o PFCM, Figuras 3.2a e 3.2c. Já suas correspondentes soluções são representadas pela Figuras 3.2b e 3.2d.

Lema 3.2.2. *Seja $b^* = (u^*, z^*)$ a solução ótima do PBEA em que u^* e z^* são o número de adições e remoções. Então, $\frac{m}{m+u^*+z^*}$ é um limite superior para o PFCM e $\frac{m-u^*}{m+z^*}$ é um limite inferior para o mesmo.*

PROVA. Seja $f(b) = \frac{m-u}{m+z}$ a função objetivo do PFCM, em que u e z representam, respectivamente, o número de uns fora e o número de zeros dentro das células. Considere também que $u + z = k$ e que k é uma constante.

Colocando $f(b)$ em função de z obtém-se que

$$\frac{m-u}{m+z} = \frac{m-(k-z)}{m+z}.$$

Calculando sua derivada,

$$\begin{aligned} \frac{df}{dz} &= \frac{(m+z) - 1(m-k+z)}{m^2 + 2mz + z^2} = \\ &= \frac{k}{m^2 + 2mz + z^2} > 0. \end{aligned}$$

Como $\frac{df}{dz} > 0$ para $u + z = k$, a função $f(z)$ é crescente, e dessa forma:

$$\frac{m}{m+u+z} > \frac{m-u}{m+z},$$

ou seja, no melhor dos casos, as k edições seriam em z (adições de aresta) já que sempre que z aumenta f também aumenta.

Considere então que $f(b) = u + z$ é uma solução qualquer do PBEA. Então, $u^* + z^* \leq u + z$ e

$$\frac{m}{m+u^*+z^*} \geq \frac{m}{m+u+z} > \frac{m-u}{m+z}.$$

Logo, $\frac{m}{m+u^*+z^*}$ é um limite superior para o PFCM.

Mostrar que $\frac{m-u^*}{m+z^*}$ é um limite inferior é trivial pois (u^*, z^*) é uma solução viável para o PFCM como mostra o Lema 3.2.1. ■

O PBEAS PARAMETRIZADO (PBEAS(η)), consiste em encontrar uma solução do PBEAS com um número de exatamente η de edições. Ao incorporá-lo no Algoritmo 3.1, foi utilizada uma adaptação com função objetivo que prioriza o menor número de remoções de arestas — consequentemente, mais adições. A formulação utilizada para defini-lo foi:

$$\min \sum_{+(ij)} (1 - y_{ij}) \quad (3.7)$$

$$\text{s.a } y_{il} + y_{kj} + y_{kl} \leq 2 + y_{ij} \quad \forall i, k \in U \text{ e } j, l \in V \quad (3.8)$$

$$\sum_{j \in V} y_{ij} \geq s_c \quad \forall i \in U \quad (3.9)$$

$$\sum_{i \in U} y_{ij} \geq s_l \quad \forall j \in V \quad (3.10)$$

$$\sum_{+(ij)} (1 - y_{ij}) + \sum_{-(ij)} y_{ij} = \eta \quad (3.11)$$

$$\sum_{+(ij)} (1 - y_{ij}) \leq U_{opt} - 1 \quad (3.12)$$

$$x_{ij} \in \{0, 1\} \quad \forall i \in U \text{ e } j \in V. \quad (3.13)$$

Algoritmo 3.1 Algoritmo Exato para Célula de Manufatura

```

1: função ECM(instância  $i$ )
2:    $x^* \leftarrow (u^*, z^*) \leftarrow PBEAS(i)$  ▷ Resolve o PBEAS para a instância  $i$ 
3:    $LS \leftarrow \frac{m}{m+u^*+z^*}$ 
4:    $LI \leftarrow \frac{m-u^*}{m+z^*}$ 
5:    $U_{opt} \leftarrow u^*$  ▷ Menor número de remoções já encontrado
6:    $cont \leftarrow 0$ 
7:   enquanto  $LS > LI$  faça
8:      $(u, z) \leftarrow BGEPS(i, u^* + z^* + cont, U_{opt})$  ▷ Resolve o PBEAS parametrizado
9:     se  $U_{opt} > u$  então
10:        $U_{opt} \leftarrow u$ 
11:     fim se
12:     se  $\frac{m-u}{m+z} > f(x^*)$  então
13:        $LI \leftarrow \frac{m-u}{m+z}$ 
14:        $x^* \leftarrow (u, z)$ 
15:     fim se
16:      $LS \leftarrow \frac{m}{m+u+z}$ 
17:      $cont \leftarrow cont + 1$ 
18:   fim enquanto
19:   retorne  $x^*$ 
20: fim função

```

Nesta formulação, encontra-se dentre todas as soluções com η edições, aquele com menor número de remoções. Como no algoritmo proposto resolve-se o PBEAS(η) várias

vezes, foi adicionada a restrição (3.12) que acelera o processo, reduzindo o espaço de busca. Sem essa restrição, a resolução de algumas instâncias seriam inviáveis. Nesse caso, U_{opt} corresponde ao menor número de remoções já encontrado pelo algoritmo (uns fora da célula).

A Figura 3.3 mostra um exemplo de execução do Algoritmo 3.1, facilitando assim, o entendimento da prova do Teorema 3.2.1. A Figura 3.3a mostra o espaço das soluções do problema. O espaço de busca é dividido em vários retângulos para cada número de edições entre a solução representada e a instância original. As soluções são representadas pelos pontos dentro dos retângulos. Se um ponto está no retângulo $\eta + i$, significa que esta é solução com $\eta + i$ edições. Sua altura determina seu valor na função objetivo do PFCM. Assim, o ponto mais alto entre todos os retângulos representa a solução ótima para o PFCM. Note que pode haver soluções com mesma função objetivo para o PFCM. Ainda na figura, η representa o ótimo do PBEA e a curva *lim sup* representa o limite superior para cada subproblema parametrizado.

Com a solução inicial do PBEA — representada pelo ponto vermelho na Figura 3.3b —, calcula-se o LI e LS, linhas vermelha e azul respectivamente. Note que a solução encontrada pelo PBEA é uma boa solução para o PFCM, porém não é ótima. No primeiro *loop* do algoritmo (Figura 3.3c) encontra-se a melhor solução com η edições e atualiza-se o valor do LI.

O algoritmo segue atualizando o LS a cada *loop*. O LI só é atualizado se for encontrada uma solução melhor, como mostram as Figuras 3.3d e 3.3e. A Figura 3.3f mostra o fim do algoritmo, em que o critério de parada é atendido. Nela, o valor do LI é maior que o valor do LS. Dessa forma, conclui-se que não serão encontradas soluções melhores.

Teorema 3.2.1. *O algoritmo ECM (Algoritmo 3.1) retorna a solução ótima do PFCM.*

PROVA. A ideia do Algoritmo 3.1 consiste em buscar a solução ótima iterativamente através de execuções do PBEAS(η), tendo como início o número de edições ótimas do PBEA. A cada iteração novos limites são calculados. Isso ocorre até que os limites se encontrem, significando que a partir desse ponto não existe nenhuma solução melhor.

De acordo com o Lema 3.2.2, sabe-se que as linhas 3, 4 e 13 realmente calculam os limites inferior e superior. Falta então mostrar que a linha 16 calcula de fato um limite superior e que o algoritmo para.

Mesmo já sendo uma boa solução, a solução ótima do PBEA não é necessariamente ótima para o PFCM. Neste caso, o ótimo do PFCM possui mais edições que o do PBEA. Para encontrá-lo, utilizam-se as chamadas do PBEAS(η). Da forma como foi definida a função objetivo do PBEAS(η), obtém-se como retorno a solução com maior número de adições de arestas dentre todas as soluções com η edições.

Ao executar o PBEAS(η), atualiza-se o limite superior para as soluções com $\eta = z^* + u^* + cont$ edições (linha 8). A linha 16 calcula de fato um limite superior, pois se

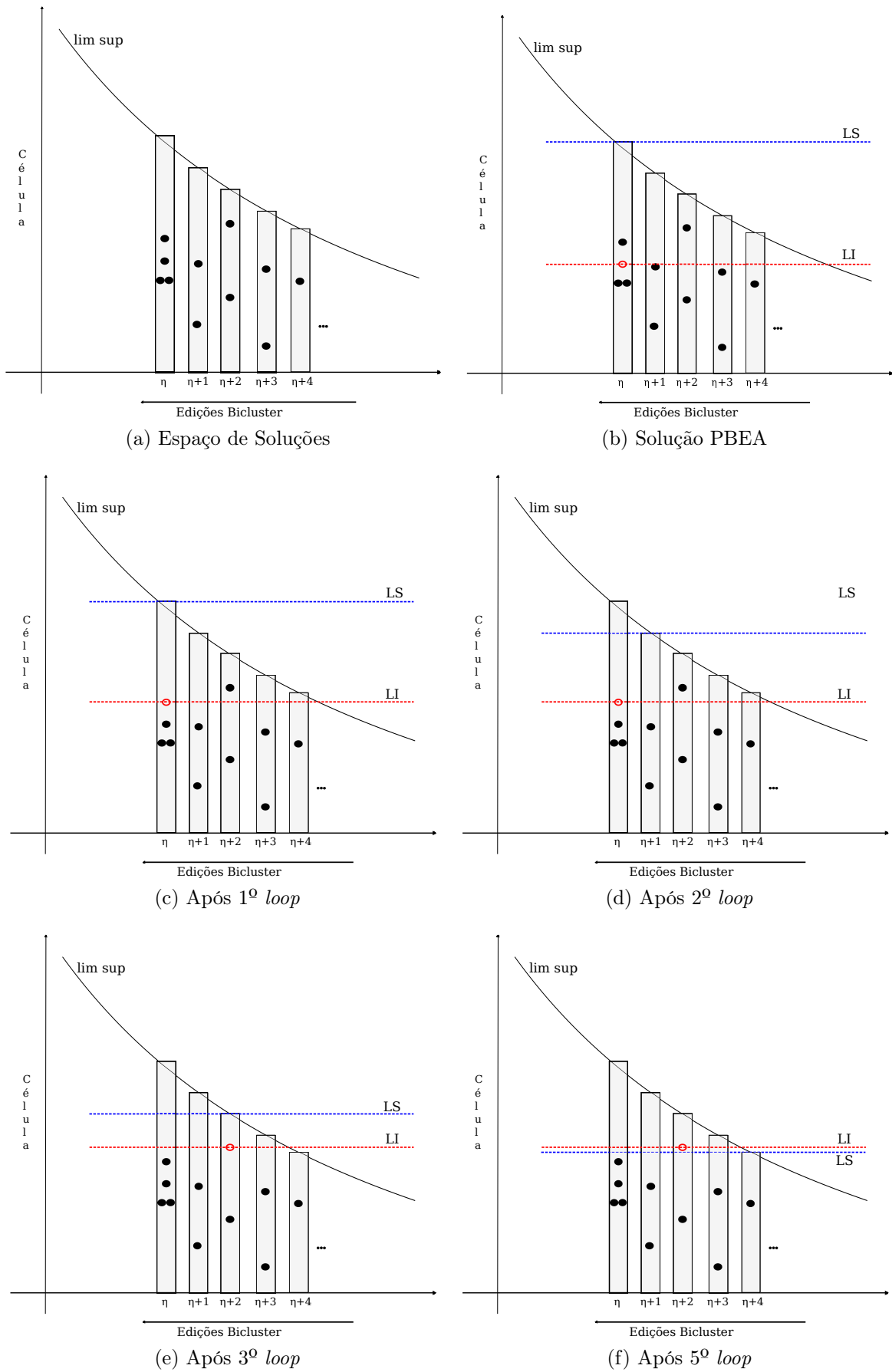


Figura 3.3: Exemplo: Algoritmo Exato PFCM.

existisse uma solução acima deste novo limite e abaixo do antigo, o algoritmo já a teria encontrado em uma iteração anterior. Como LS diminui a cada iteração conclui-se que o algoritmo para. ■

3.3 Nova Formulação Matemática para o PFCM

De acordo com o Lema 3.2.1, sabe-se que o PBEA e o PFCM possuem o mesmo espaço de solução. Com isso, as restrições da formulação do PBEA podem ser utilizadas em uma nova formulação para o PFCM. Faltava apenas adaptar a função objetivo de acordo com a eficácia de agrupamento descrita na Seção 3.1.3. Sabendo que $N_1^{out} = \sum_{+(ij)} (1 - y_{ij})$ e $N_0^{in} = \sum_{-(ij)} y_{ij}$, propõe-se uma nova formulação para o PFCM baseada em programação fracionária linear.

$$\max \quad \frac{m - \sum_{+(ij)} (1 - y_{ij})}{m + \sum_{-(ij)} y_{ij}} \quad (3.14)$$

$$\text{s.a} \quad y_{il} + y_{kj} + y_{kl} \leq 2 + y_{ij} \quad \forall i, k \in U \text{ e } j, l \in V \quad (3.15)$$

$$\sum_{j \in V} y_{ij} \geq s_c \quad \forall i \in U \quad (3.16)$$

$$\sum_{i \in U} y_{ij} \geq s_l \quad \forall j \in V \quad (3.17)$$

$$y_{ij} \in \{0, 1\} \quad \forall i \in U \text{ e } j \in V. \quad (3.18)$$

3.4 Instâncias do PFCM

Como o PFCM é muito explorado pela literatura, muitos trabalhos propuseram instâncias para o problema. Neste trabalho, utiliza-se 35 instâncias disponíveis em Gonçalves & Resende [27] e utilizadas nos principais trabalhos sobre o tema. A Tabela 3.1 apresenta as instâncias juntamente com suas dimensões e seu artigo fonte.

Tabela 3.1: Instâncias da Literatura do PFCM

Instância	Dimensão	Fonte
King1982	05×07	King & Nakornchai [33]
Waghodekar1984	05×07	Waghodekar & Sahu [62]
Seifoddini1989	05×18	Seifoddini [55]
Kusiak1992	06×08	Kusiak & Cho [38]
Kusiak1987	07×11	Kusiak & Chow [39]
Boctor1991	07×11	Boctor [7]
Seifoddini1986	08×12	Seifoddini & Wolfe [54]
Chandrasekharan1986a	08×20	Chandrasekharan & Rajagopalan [13]
Chandrasekharan1986b	08×20	Chandrasekharan & Rajagopalan [12]
Mosier1985a	10×10	Mosier & Taube [46]
Chan1982	10×15	Chan & Milner [11]
Askin1987	14×24	Askin & Subramanian [5]
Stanfel1985	14×24	Stanfel [60]
McCormick1972a	16×24	McCormick et al. [44]
Srinivasan1990	16×30	Srinivasan et al. [59]
King1980	16×43	King [34]
Carrie1973a	18×24	Carrie [10]
Mosier1985b	20×20	Mosier & Taube [47]
Kumar1986	20×23	Kumar et al. [36]
Carrie1973b	20×35	Carrie [10]
Boe1991	20×35	Boe & Cheng [8]
Chandrasekharan1989_1	24×40	Chandrasekharan & Rajagopalan [15]
Chandrasekharan1989_2	24×40	Chandrasekharan & Rajagopalan [15]
Chandrasekharan1989_3-4	24×40	Chandrasekharan & Rajagopalan [15]
Chandrasekharan1989_5	24×40	Chandrasekharan & Rajagopalan [15]
Chandrasekharan1989_6	24×40	Chandrasekharan & Rajagopalan [15]
Chandrasekharan1989_7	24×40	Chandrasekharan & Rajagopalan [15]
McCormick1972b	27×27	McCormick et al. [44]
Carrie1973c	28×46	Carrie [10]
Kumar1987	30×51	Kumar & Vannelli [37]
Stanfel1985_1	30×50	Stanfel [60]
Stanfel1985_2	30×50	Stanfel [60]
King1982	30×90	King & Nakornchai [33]
McCormick1972c	37×53	McCormick et al. [44]
Chandrasekharan1987	40×100	Chandrasekharan & Rajagopalan [14]

Capítulo 4

Resultados Computacionais

Neste capítulo os resultados computacionais referentes ao PBEA e ao PFCM serão apresentados. Primeiramente, apresentam-se as ferramentas utilizadas no testes computacionais na Seção 4.1, e discute-se na Seção 4.2 as métricas utilizadas na avaliação dos resultados. Em seguida, na Seção 4.3, analisam-se os resultados para o PBEA, e na Seção 4.4 os resultados do PFCM com as diferentes restrições *singleton*. Todas as instâncias propostas estão disponibilizadas no anexo A, da mesma forma que os resultados ótimos para o PBEA e para o PFCM estão disponíveis nos anexos B e C.

4.1 Ferramentas

Como ferramenta utiliza-se o CPLEX, que é um dos pacotes de *software* de otimização linear mista mais utilizados na literatura. O CPLEX é responsável por gerenciar todo o processo do B&C, inclusive:

- escolhas de variáveis para o *branch*;
- executar o algoritmo de separação;
- adicionar os cortes gerados pelo pré-processamento.

Todos os métodos desenvolvidos foram implementados em C++ e executados em uma máquina Intel Core i7-2600 com 3,40 GHz e 32 GB de memória RAM no sistema operacional Arch Linux 3.3.4.

4.2 Métricas de Avaliação para o BGEP

Nesta seção, serão apresentadas as métricas utilizadas para comparar os diferentes cenários em relação às instâncias utilizadas. Para isso, utiliza-se a Média Geométrica dos tempos computacionais. Com relação às instâncias, o fator de aceleração compara o

desempenho de cada cenário sobre uma instância específica, já que cada cenário pode ter um comportamento distinto para cada tipo de instância.

4.2.1 Média Geométrica

A Média Geométrica sobre um conjunto de dados $\{a_1, a_2, \dots, a_n\}$ é definida como:

$$G = \sqrt[n]{a_1 a_2 \cdots a_n}. \quad (4.1)$$

Esta métrica permite a comparação de diferentes dados sem necessidade de normalizá-los. A vantagem de sua utilização em relação à média aritmética é que, se a mesma fosse utilizada, daria maior peso a itens de grande valor. A Tabela 4.1a mostra tempos computacionais de um exemplo em que diferentes instâncias foram executadas em diferentes cenários.

Tabela 4.1: Exemplo: média geométrica.

(a) Tempos originais.

	Cenário A	Cenário B	Cenário C
Instância 1	1	10	20
Instância 2	1000	100	20
Média Aritmética	500,5	55	20
Média Geométrica	31,62	31,62	20

(b) Normalização no cenário A.

	Cenário A	Cenário B	Cenário C
Instância 1	1	10	20
Instância 2	1	0,1	0,02
Média Arit.	1	5,05	10,01
Média Geom.	1	1	0,63

(c) Normalização no cenário B.

	Cenário A	Cenário B	Cenário C
Instância 1	0,1	1	2
Instância 2	10	1	0,2
Média Arit.	5,05	1	1,1
Média Geom.	1	1	0,63

Tanto a média aritmética quanto a média geométrica mostram que o cenário C é o mais rápido de acordo com os tempos originais. Porém, com a normalização pelo cenário A (Tabela 4.1b), o cenário mais rápido de acordo com a média aritmética passa a ser o cenário A. Entretanto, a média geométrica continua apontando o cenário C como o mais rápido. Veja que o mesmo ocorre se o cenário B fosse normalizado, como mostra a Tabela 4.1c. Na Seção 4.3, verificam-se os tempos computacionais para oito cenários e a utilização da média geométrica como forma de comparação entre os cenários.

4.2.2 Fator de aceleração

A avaliação dos resultados sobre uma determinada instância pode ser feita através do fator de aceleração. Ele consiste em uma normalização dos resultados através da divisão de seu tempo computacional pelo menor tempo naquela instância. Por exemplo, suponha que

serão executados n algoritmos, tendo como entrada a mesma instância, resultando assim nos seguintes tempos computacionais: $T_{A_1}, T_{A_2}, \dots, T_{A_{n-1}}$ e T_{A_n} . O fator de aceleração do algoritmo A_i sobre a instância é definido como:

$$S_{A_i} = \frac{T_{A_i}}{\min\{T_{A_1}, \dots, T_{A_n}\}}. \quad (4.2)$$

Veja que quanto menor o fator de aceleração melhor o resultado do algoritmo.

4.3 Resultados para o PBEA

No PBEA, foi aplicado o algoritmo proposto nas instâncias criadas (Seção 2.6). Todas as possíveis combinações entre algoritmo de separação, heurística e pré-processamento foram comparadas, havendo assim um total de oito possíveis cenários, como mostra a Tabela 4.2.

Tabela 4.2: Cenários considerados.

Cenário	Alg. de Sep.	Heurística	Pré-Proces.
-l	CPLEX		
-l -g	CPLEX	✓	
-l -R	CPLEX		✓
-l -g -R	CPLEX	✓	✓
-d	Prog. Dinâmica		
-d -g	Prog. Dinâmica	✓	
-d -R	Prog. Dinâmica		✓
-d -g -R	Prog. Dinâmica	✓	✓

A Tabela 4.3 apresenta as instâncias geradas, seus resultados ótimos, o número de variáveis fixadas e os cortes gerados pelo pré-processamento. Um exemplo notação utilizada para apresentar as instâncias é 10x11_i1_p0.4, em que i1 é sua identificação e a mesma foi gerada com o modelo binomial $\mathbb{G}(m, n, p)$ com $m = 10$, $n = 11$ e $p = 0,4$. A associação entre o resultado do pré-processamento e a densidade da instância também pode ser vista nesta tabela. Verifica-se que, para instâncias esparsas — principalmente com $p = 0,2$ —, ocorre uma grande quantidade de fixações e cortes, diferentemente dos casos em que as instâncias são densas. Isso ocorre porque, em uma instância esparsa, a probabilidade de dois vértices estarem a uma distância maior ou igual a 4 (e atender o Teorema 2.5.1) é maior que em uma instância densa.

Um comparativo dos tempos computacionais dos cenários é apresentado na Tabela 4.4. Note que o melhor resultado obtido foi do cenário com programação dinâmica, heurística primal e pré-processamento. Com relação ao algoritmo de separação, pode-se concluir que ele apresenta melhores resultados que o algoritmo do CPLEX.

Já no que diz respeito ao pré-processamento, conclui-se que na maioria das instâncias em que cortes foram gerados, o pré-processamento conseguiu reduzir o tempo computacional. Seu resultado geral só não foi melhor por causa de seu desempenho nas instâncias i28 e i29. Acredita-se que o mecanismo do CPLEX que adiciona os cortes ao modelo não foi eficiente por causa do grande número de cortes gerados nessas instancias (5452 e 3590).

A Tabela 4.5 mostra o fator de aceleração do PBEA. Observa-se que o cenário com programação dinâmica obteve melhores tempos em 8 instâncias. Em contrapartida, as menores somas de fator de aceleração foram obtidas pelo algoritmo do CPLEX. Isso ocorre devido a um desempenho bem inferior nas instâncias i19, i23 e i26, que são consideradas fáceis. Em outras palavras, a programação dinâmica obteve um desempenho superior nas instâncias difíceis e inferior nas fáceis. Com relação à heurística, conclui-se que ela não apresenta bons resultados, a não ser quando executada juntamente com o pré-processamento.

Tabela 4.3: Restrições geradas pelo Pré-Processamento

Instância	Valor Ótimo	Variáveis Fixadas	Cortes Gerados
10x11_i1_p0.4	17	15	291
10x11_i2_p0.4	20	2	208
10x18_i3_p0.4	36	0	366
11x13_i4_p0.4	34	0	48
11x15_i5_p0.4	35	0	217
11x16_i6_p0.4	30	0	404
11x18_i7_p0.4	50	0	187
12x13_i8_p0.4	28	6	404
12x13_i9_p0.4	34	0	176
12x19_i10_p0.4	52	0	230
12x19_i11_p0.4	53	0	168
13x19_i12_p0.4	61	0	84
15x16_i13_p0.4	54	0	267
16x17_i14_p0.4	63	0	215
16x17_i15_p0.4	61	0	217
10x11_i16_p0.6	26	0	0
10x14_i17_p0.6	40	0	0
11x11_i18_p0.6	34	0	0
12x14_i19_p0.6	49	0	0
12x15_i20_p0.6	51	0	12
15x25_i21_p0.8	88	0	0
15x26_i22_p0.8	74	0	0
16x16_i23_p0.2	24	56	2112
16x18_i24_p0.2	24	77	2726
16x29_i25_p0.8	94	0	0
17x18_i26_p0.2	23	98	3002
17x29_i27_p0.8	111	0	0
20x28_i28_p0.2	62	47	5452
21x22_i29_p0.2	56	32	3590
21x22_i30_p0.8	56	0	0

Tabela 4.4: Resultado PBEA.

Instância	Tempo (s) -l	Tempo (s) -d	Tempo (s) -l -R	Tempo (s) -d -R	Tempo (s) -d-g-R	Tempo (s) -l -g	Tempo (s) -d -g	Tempo (s) -l-g-R
10x11_i1_p0.4	0.04	0.08	0.04	0.18	0.08	0.06	0.10	0.06
10x11_i2_p0.4	0.09	0.31	0.09	0.13	0.19	0.10	0.25	0.13
10x18_i3_p0.4	7.25	15.76	7.18	15.84	9.57	10.76	18.30	13.56
11x13_i4_p0.4	4.25	4.34	4.23	4.36	3.22	4.10	3.83	3.75
11x15_i5_p0.4	11.15	14.09	11.21	14.08	19.43	25.55	17.07	21.04
11x16_i6_p0.4	0.51	1.95	0.51	1.96	2.17	0.88	2.32	0.94
11x18_i7_p0.4	743.29	202.75	746.77	203.69	222.23	551.62	257.26	551.43
12x13_i8_p0.4	1.51	2.45	1.63	2.35	2.43	1.56	2.81	1.48
12x13_i9_p0.4	13.31	15.05	13.19	15.15	20.41	17.38	16.54	17.60
12x19_i10_p0.4	1415.27	1016.15	1403.06	1015.33	1120.70	1972.83	908.89	3254.65
12x19_i11_p0.4	3015.50	1575.33	2981.79	1569.51	1942.72	4376.48	2296.79	5496.54
13x19_i12_p0.4	29262.02	8025.49	29250.63	8048.23	7452.90	33587.71	7035.40	36987.39
15x16_i13_p0.4	2815.28	867.07	2780.84	861.15	1238.03	2870.80	977.69	3384.44
16x17_i14_p0.4	33186.61	12941.98	32362.63	12827.84	11924.25	67715.43	13016.25	49879.48
16x17_i15_p0.4	9003.68	3012.89	9052.72	3016.28	4354.71	14982.92	4223.17	10976.73
10x11_i16_p0.6	0.17	0.09	0.17	0.09	0.10	0.08	0.10	0.08
10x14_i17_p0.6	2.58	2.16	2.57	2.16	1.75	2.32	4.57	2.06
11x11_i18_p0.6	2.89	5.36	2.87	5.43	3.12	1.82	4.91	2.49
12x14_i19_p0.6	19.06	42.73	18.99	42.97	9.83	9.81	7.86	4.01
12x15_i20_p0.6	162.74	147.52	161.67	148.65	62.48	174.53	144.08	210.49
15x25_i21_p0.8	0.73	0.32	0.72	0.32	0.34	0.73	0.33	0.72
15x26_i22_p0.8	0.39	0.21	0.39	0.21	0.22	0.39	0.22	0.39
16x16_i23_p0.2	0.22	1.99	0.29	1.98	2.95	0.72	2.27	0.79
16x18_i24_p0.2	0.11	0.57	0.10	0.49	0.68	0.26	0.64	0.20
16x29_i25_p0.8	1.07	0.38	1.07	0.38	0.39	1.08	0.39	1.08
17x18_i26_p0.2	0.10	0.47	0.09	0.59	0.49	0.37	1.40	0.26
17x29_i27_p0.8	1.84	0.60	1.86	0.60	0.62	1.84	0.62	1.85
20x28_i28_p0.2	36609.40	36974.78	45255.89	47836.95	31751.47	110931.15	43031.79	137176.92
21x22_i29_p0.2	14786.80	12362.10	21635.84	13692.32	16097.36	29482.76	14886.42	46239.86
21x22_i30_p0.8	0.86	0.31	0.85	0.31	0.37	1.10	0.33	1.00
TOTAL	131068.72	77235.28	145699.89	89329.53	76245.21	266727.14	86862.60	294231.42
Média Geométrica	16.33	15.92	16.68	16.11	15.14	21.19	17.24	21.29

Tabela 4.5: Fator de aceleração

Instância	Tempo (s) -l	Tempo (s) -d	Tempo (s) -l -R	Tempo (s) -d -R	Tempo (s) -d-g-R	Tempo (s) -l -g	Tempo (s) -d -g	Tempo (s) -l-g-R
10x11_i1_p0.4	1.00	2.00	1.00	4.50	2.00	1.50	2.50	1.50
10x11_i2_p0.4	1.00	3.44	1.00	1.44	2.11	1.11	2.78	1.44
10x18_i3_p0.4	1.01	2.19	1.00	2.21	1.33	1.50	2.55	1.89
11x13_i4_p0.4	1.32	1.35	1.31	1.35	1.00	1.27	1.19	1.16
11x15_i5_p0.4	1.00	1.26	1.01	1.26	1.74	2.29	1.53	1.89
11x16_i6_p0.4	1.00	3.82	1.00	3.84	4.25	1.73	4.55	1.84
11x18_i7_p0.4	3.67	1.00	3.68	1.00	1.10	2.72	1.27	2.72
12x13_i8_p0.4	1.02	1.66	1.10	1.59	1.64	1.05	1.90	1.00
12x13_i9_p0.4	1.01	1.14	1.00	1.15	1.55	1.32	1.25	1.33
12x19_i10_p0.4	1.56	1.12	1.54	1.12	1.23	2.17	1.00	3.58
12x19_i11_p0.4	1.92	1.00	1.90	1.00	1.24	2.79	1.46	3.50
13x19_i12_p0.4	4.16	1.14	4.16	1.14	1.06	4.77	1.00	5.26
15x16_i13_p0.4	3.27	1.01	3.23	1.00	1.44	3.33	1.14	3.93
16x17_i14_p0.4	2.78	1.09	2.71	1.08	1.00	5.68	1.09	4.18
16x17_i15_p0.4	2.99	1.00	3.00	1.00	1.45	4.97	1.40	3.64
10x11_i16_p0.6	2.13	1.13	2.13	1.13	1.25	1.00	1.25	1.00
10x14_i17_p0.6	1.47	1.23	1.47	1.23	1.00	1.33	2.61	1.18
11x11_i18_p0.6	1.59	2.95	1.58	2.98	1.71	1.00	2.70	1.37
12x14_i19_p0.6	4.75	10.66	4.74	10.72	2.45	2.45	1.96	1.00
12x15_i20_p0.6	2.60	2.36	2.59	2.38	1.00	2.79	2.31	3.37
15x25_i21_p0.8	2.28	1.00	2.25	1.00	1.06	2.28	1.03	2.25
15x26_i22_p0.8	1.86	1.00	1.86	1.00	1.05	1.86	1.05	1.86
16x16_i23_p0.2	1.00	9.05	1.32	9.00	13.41	3.27	10.32	3.59
16x18_i24_p0.2	1.10	5.70	1.00	4.90	6.80	2.60	6.40	2.00
16x29_i25_p0.8	2.82	1.00	2.82	1.00	1.03	2.84	1.03	2.84
17x18_i26_p0.2	1.11	5.22	1.00	6.56	5.44	4.11	15.56	2.89
17x29_i27_p0.8	3.07	1.00	3.10	1.00	1.03	3.07	1.03	3.08
20x28_i28_p0.2	1.15	1.16	1.43	1.51	1.00	3.49	1.36	4.32
21x22_i29_p0.2	1.20	1.00	1.75	1.11	1.30	2.38	1.20	3.74
21x22_i30_p0.8	2.77	1.00	2.74	1.00	1.19	3.55	1.06	3.23
TOTAL	59.60	69.68	60.40	71.20	64.87	76.23	77.47	76.59
Melhores Resultados	5	8	7	7	5	2	2	3

4.4 Resultados para a Célula de Manufatura

No PFCM, foi aplicado o algoritmo proposto na Seção 3.2 nas instâncias encontradas na literatura. A Tabela 4.6 compara o ótimo com o melhor resultado da literatura com a restrição de *singleton* 2×2 . Além disso, também compara o ótimo do PBEA com o número de edições do resultado ótimo do PFCM. Na última coluna tem-se o último parâmetro na qual o PBEAS(η) foi executado.

Observe que o algoritmo encontrou o ótimo em 27 de 35 instâncias. Inclusive, na instância King1980, foi encontrada uma solução ainda desconhecida. Além disso, em 17 dos 27 ótimos encontrados, a solução ótima do PBEAS também é ótima para o PFCM. Isso reforça que a função objetivo linear para o PFCM apresenta bons resultados para o problema.

A Tabela 4.7 mostra o resultado para o PFCM sem as restrições de *singleton*. Da mesma forma que o resultado anterior, o algoritmo encontrou o ótimo em 26 de 35 instâncias. Em 3 instâncias, o algoritmo encontra soluções ainda desconhecidas na literatura (King1980, Kumar1987 e Stanfel1985_1). A instância Chandrasekaran1989_5 foi resolvida apenas com a restrição de *singleton*; as restantes ou não foram resolvidas em nenhum dos dois casos ou foram resolvidas nos dois casos. Com relação à diferença de edições entre o ótimo do PBEA com o ótimo do PFCM, ela é em média 1,81 com restrição *singleton* e 1,13 sem a restrição.

Os resultados da literatura para o caso com restrição *singleton* foram extraídos dos trabalhos:

Wu et al. [64]: *simulated annealing*;

Wu et al. [66]: algoritmo *water flow-like*;

Gonçalves & Resende [27]: algoritmo evolucionário.

Já no caso sem restrições, os trabalhos com os melhores resultados são:

Wu et al. [65]: método de coeficientes de similaridade com função de Boltzmann e operador de mutação;

Pailla et al. [49]: *simulated annealing*;

Elbenani et al. [21]: algoritmo genético híbrido.

Nas duas tabelas, houve instâncias em que na literatura foram encontrados resultados “melhores que os exatos”. Isso se deve por conta de prováveis erros de digitação. No anexo D apontam-se os erros.

Tabela 4.6: Resultado do Algoritmo Exato com restrição de *Sigleton* 2×2 .

Instância	Literatura	Exato	Ótimo PBEAS	Edições Célula	Último η
King1982	73.68	73.68	5	5	19
Waghodekar1984	62.50	62.50	9	9	15
Seiffodini1989	79.59	79.59	10	10	12
Kusiak1992	76.92	76.92	5	5	26
Kusiak1987	53.13	53.12	15	15	21
Boctor1991	70.37	70.37	7	7	24
Seiffodini1986	68.30	68.29	13	13	25
Chandrasekaran1986a	85.25	85.24	9	9	33
Chandrasekaran1986b	58.72	58.71	43	45	69
Mosier1985a	70.59	70.58	10	10	10
Chan1982	92.00	92.00	4	4	4
Askin1987	69.86	69.86	21	22	28
Stanfel1985	69.33	69.33	22	23	28
McCormick1972a	51.96	51.96	46	49	83
Srinivasan1990	67.83	67.83	46	46	75
King1980	55.90	56.52	68	70	103
Carrie1973a	54.46	54.46	47	51	75
Mosier1985b	42.96				
Kumar1986	49.65	49.64	64	71	115
Carrie1973b	76.54	76.54	37	37	42
Boe1991	58.15	58.15	73	77	116
Chandrasekharan1989_1	100.00	100.00	0	0	0
Chandrasekharan1989_2	85.11	85.10	21	21	52
Chandrasekharan1989_3-4	73.51	73.50	39	39	67
Chandrasekharan1989_5	51.97	51.97	70	73	127
Chandrasekharan1989_6	47.37				
Chandrasekharan1989_7	44.87				
McCormick1972b	54.27				
Carrie1973c	46.06				
Kumar1987 ¹	58.58	58.94	60	62	93
Stanfel1985_1 ¹	59.66	59.65	71	71	105
Stanfel1985_2	50.51				
King1982	42.64				
McCormick1972c	59.85				
Chandrasekharan1987	84.03	84.03	73	73	106

¹Erros encontrados nessas instâncias. Ver anexo D

Tabela 4.7: Resultado do Algoritmo Exato sem restrição de *Sigleton*.

Instância	Literatura	Exato	Ótimo PBEAS	Edições Célula	Último η
King1982 ¹	75	75	4	4	19
Waghodekar1984	69.57	69.56	7	7	12
Seiffodini1989	80.85	80.85	9	9	12
Kusiak1992	79.17	79.16	5	5	6
Kusiak1987	60.87	60.86	9	9	16
Boctor1991	70.83	70.83	7	7	24
Seiffodini1986	69.44	69.44	11	11	17
Chandrasekaran1986a	85.25	85.24	9	9	31
Chandrasekaran1986b	58.72	58.71	43	45	65
Mosier1985a	75	75	7	7	8
Chan1982	92	92	4	4	4
Askin1987	74.24	74.24	17	17	22
Stanfel1985	72.86	72.85	18	19	23
McCormick1972a	53.33	53.33	42	42	86
Srinivasan1990	69.92	69.92	39	40	61
King1980	57.96	58.04	59	60	102
Carrie1973a	57.73	57.73	40	41	65
Mosier1985b	43.26				
Kumar1986	50.81	50.80	60	61	114
Carrie1973b	78.40	78.40	32	33	36
Boe1991	58.79	58.79	70	75	114
Chandrasekharan1989_1	100	100	0	0	0
Chandrasekharan1989_2	85.11	85.10	21	21	52
Chandrasekharan1989_3-4	73.51	73.50	39	40	67
Chandrasekharan1989_5	53.29				
Chandrasekharan1989_6	48.63				
Chandrasekharan1989_7	46.15				
McCormick1972b	54.82				
Carrie1973c	47.68				
Kumar1987 ¹	62.86	63.04	51	51	76
Stanfel1985_1 ¹	59.66	59.77	67	70	104
Stanfel1985_2	50.83				
King1982	47.93				
McCormick1972c	61.16				
Chandrasekharan1987	84.03	84.03	73	73	106

Capítulo 5

Conclusões e Trabalhos Futuros

Vimos que a resolução do PBEA através de métodos exatos é satisfatória em instâncias de até 140 vértices (Chandrasekharan1987). Percebemos que o algoritmo de separação proposto apresenta desempenho superior ao utilizado pelo CPLEX, principalmente se combinado com o pré-processamento e a heurística primal. O pré-processamento apresentou fixações em poucos casos. Isso ocorre pois as instâncias utilizadas são densas, diferentemente do que teria ocorrido com instâncias esparsas. Em compensação, conseguiu gerar uma grande quantidade de cortes.

Com relação ao PFCM, vimos que sua relação com o PBEA representa uma nova abordagem que trará bons resultados práticos e teóricos. Nesta abordagem, foi obtida a solução ótima de instâncias criadas há quase 40 anos. Além disso, fica claro que a utilização de uma função objetivo linear para o PFCM apresenta bons resultados.

Dentre as contribuições deste trabalhos podemos citar:

- algoritmo B&C para o PBEA com:
 - algoritmo de separação;
 - pré-processamento;
- relação do PBEA com o PFCM;
- algoritmo exato para o PFCM;
- nova formulação linear fracionária para o PFCM.
- criação de instâncias para o PBEA e resolução exata;
- resolução exata da maioria das instâncias do PFCM;
- apresentação do impacto da utilização de uma função objetivo linear no PFCM.

Vale ressaltar a dificuldade encontrada em examinar os trabalhos da literatura do PFCM, um vez que muitos deles não explicam detalhes da formulação utilizada, principalmente com relação ao tamanho mínimo da célula. Outros problemas foram os erros de digitação das instâncias encontrados em alguns trabalhos. Todos esse erros encontrados estão disponíveis no anexo D.

Para trabalhos futuros, serão pesquisadas novas melhorias que acelerem o *branch*, como incorporá-lo a uma meta-heurística ou desenvolver novas heurísticas primais. Será realizado um estudo mais aprofundado com relação às instâncias mais difíceis, além de desenvolver novas restrições e procurar novas maneiras de gerenciar os cortes gerados, já que, como foi visto na Seção 4.3 para instâncias com $p = 0,2$, o CPLEX não gerenciou bem os cortes. Outro campo para novos trabalhos pode ser encontrado na exploração de outras variantes do PBEA, assim como a incorporação de técnicas do PFCM a este problema.

Apêndice A

Instâncias

A.1 PBEA

```
0101110011
0101001101
0011010000
0110101011
0010100110
0010001010
0000010000
0110101010
0000100010
0000010000
0110000100
```

10x11_i01_p0.4

```
1000101010
0010110111
0001010100
0011100101
1100000000
0110000100
0100001001
1010101100
0000101001
1010101010
0001001011
```

10x11_i02_p0.4

```
1011110110
0011101001
0001010100
1101010000
1100111001
1010001110
0000001000
0101111001
0100111001
1100001101
0011110000
0000000001
0000010000
0000000111
1000101100
0001100000
0001100001
1001010110
```

10x18_i03_p0.4

```
10001110011
11011000010
10100110010
01100100111
00111010110
10001011001
10001000111
10011001010
11011010111
10010011001
10100011011
11011001110
00011000100
```

11x13_i04_p0.4

```
00011100000
01100110000
10101100111
11010001110
10000111001
00100000100
01010111010
01000001000
10011100011
10110001001
00000110100
00010100000
11001000010
00111001000
10100110011
```

11x15_i05_p0.4

```
11101010111
01101001010
11000000100
11100100000
11100000100
00000101001
00001001110
11000001100
00110010000
10000110111
01110000001
00110000100
00001000011
01101010110
00000011010
00110000000
```

11x16_i06_p0.4

```
00001100011
10001011110
11011110001
11110000110
11100010010
01000110001
11101110100
00101000010
00100100101
01100111001
1111100010
01110110001
00111010100
10010011101
0000000100
01000001000
10011101010
11111010010
```

11x18_i07_p0.4

```
101011011011
000000110010
101000100010
000000001111
000100000000
10000001010
000011100001
000100000000
110011011010
010000100011
101101011111
010010110010
001000001000
```

12x13_i08_p0.4

```

0 1 0 0 1 0 1 0 1 0 0 1
0 0 0 0 1 1 0 0 1 0 1 0
0 1 0 0 0 0 0 0 1 1 0 0
0 0 0 0 0 0 0 1 1 1 0 1
1 0 0 0 0 0 0 1 0 1 0 0
1 0 0 0 0 0 1 1 0 0 1 0
0 1 1 0 1 0 0 0 1 1 0 0
1 0 0 1 1 1 1 1 0 0 0 0
0 1 0 0 1 0 1 1 1 0 1 0
0 0 1 0 0 0 0 0 1 1 0 0
1 0 0 1 0 0 1 0 0 1 1 0
1 0 0 0 1 0 0 0 1 1 0 1
0 1 1 0 1 1 0 0 0 1 0 1

```

13x19 i12 p0.4

```

1 0 0 1 1 0 0 1 0 1 0 0 0 0 0 1
1 0 1 0 1 0 1 0 0 1 0 1 0 1 0 0
0 0 0 0 1 1 0 1 0 1 0 1 0 1 0 0
0 1 1 1 0 1 1 1 0 1 1 0 0 0 1 0
0 0 1 1 0 1 0 1 0 0 0 0 1 0 0 1
0 0 0 0 0 0 0 1 1 1 0 1 0 1 0 1
0 0 1 0 0 1 0 0 0 0 0 0 1 0 1 0 0
1 0 0 0 0 1 1 0 0 0 0 1 0 1 1 1 1
0 1 1 0 1 1 1 0 1 0 1 1 0 0 0 0 1
1 0 0 0 1 1 0 1 1 0 0 0 0 1 0 1 0
1 0 1 0 0 1 0 0 0 0 0 0 0 1 0 0 1
1 0 0 1 0 1 0 0 0 0 0 0 1 0 1 0 0
0 0 0 0 0 1 0 0 1 0 1 0 0 1 0 0 1
0 0 0 0 0 1 1 0 0 0 1 0 0 1 0 0 1
1 0 0 0 0 1 1 1 1 1 0 0 1 0 0 0 1
1 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 1 0 0 1 0 0 0 0 0 1 1 0 1

```

```

0 1 1 0 0 0 1 0 0 1 1 0 1 1 0 0
0 0 0 1 1 0 0 1 0 1 0 0 0 0 0 0
0 0 0 1 0 0 0 1 1 0 1 1 1 1 0 0
1 0 1 0 1 1 1 1 1 1 0 1 0 0 1 0
0 0 1 1 0 0 1 1 0 1 0 1 0 0 1 0
1 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 0 1 0 1 1 0 1
0 0 0 0 1 1 0 0 1 1 0 0 1 1 1 0
0 1 1 0 1 0 0 0 0 0 0 1 0 0 0 0
1 0 0 0 1 0 1 0 1 0 1 0 0 0 0 1 1
0 0 1 1 0 1 0 0 1 0 1 1 0 1 1 0
0 1 0 0 1 1 1 1 0 0 0 1 0 0 0 0
0 1 1 1 0 0 1 0 1 1 1 0 0 1 1 0 0
1 0 1 0 0 0 0 1 1 1 1 0 0 0 1 1
1 0 0 0 1 0 0 0 1 1 0 0 1 1 0 0
0 0 1 0 0 1 1 0 0 0 1 0 0 0 0 1
0 0 1 1 1 1 0 1 1 0 1 1 0 0 0 0

```

16x17_i15_p0.4

```

1 0 0 1 1 1 1 0 1 1
0 0 1 1 1 1 1 0 1 1
1 0 1 0 1 1 1 0 0 1
1 1 0 0 0 0 1 0 1 0
1 1 0 1 1 0 1 1 1 1
1 0 0 0 1 1 1 1 1 1
0 1 0 1 1 0 1 1 0 1
0 1 1 1 1 1 1 1 0 0
1 1 1 1 1 0 1 1 1 1
1 0 0 1 1 1 0 1 1 1
0 0 1 1 1 0 1 1 1 0
0 0 0 1 1 1 1 0 1 1
1 1 1 1 1 0 0 1 0 1
1 0 1 0 1 1 1 0 1 1

```

```

1 1 0 1 1 1 0 1 1 1 1
1 1 0 1 0 0 1 0 0 0 0
0 1 1 0 0 0 1 0 0 0 1
1 0 1 1 0 0 0 1 1 1 1
1 1 1 1 0 0 0 1 1 1 1
1 0 0 1 0 0 1 0 1 0 1
0 1 1 1 1 1 1 0 1 1 1
1 0 1 1 0 1 1 0 0 1 1
0 1 0 0 0 1 1 1 1 1 1
1 1 1 1 0 0 0 0 0 1 0
0 0 1 1 1 1 0 0 1 1 1

```

```

1 1 1 1 1 1 1 1 0 1 1
0 0 0 1 0 1 1 0 1 1 1
1 0 1 0 0 1 1 0 1 1 0
1 1 1 1 1 1 1 1 1 1 0
1 1 1 0 0 0 1 1 0 1 0
1 1 0 1 1 1 1 1 0 0 1
1 1 1 1 0 1 1 1 1 0 0
0 0 1 1 1 0 0 1 0 1 0
1 1 0 1 1 1 0 1 1 1 0
1 1 1 1 0 0 1 1 1 1 0
0 1 1 1 0 0 1 0 1 0 0
1 1 1 0 0 0 0 1 0 1 1
1 1 1 1 1 0 1 1 1 1 1
1 1 1 1 1 1 1 0 1 1 0

```

12x14_i19_p0.6

```

110101101010
110011111001
111001110111
110001110100
111101100010
100001011000
111100010100
111011110110
101110100100
011011110101
111001000111
000000101001
101110101001
110010001111
001101110111

```

12x15_i20_p0.6

```

001110101101111
111111011010111
111111111110111
101111101100011
100111101110111
111110101111110
101111111101111
011111111111101
110111111110111
010001111111010
011011111110011
111111111101110
111101010111010
110111111100101
011110001111110
101111111110111
110101111111010
101111111011110
111101011001110
111111111111110
111001001101110
111001110101111
101111011111111
111100101111011
101111111101111

```

15x25_i21_p0.8

```

111001011111111
101011101111011
000111111101101
110111111111111
111011111011011
111100111111101
110110011111101
111110111100111
111111110110111
111011100110101
111111110110111
111101011101111
110111011111111
110111011111111
111110011110111
111111111011111
111110111110111
111110111111011
111110111110011
111011111011111
101100100011101
111111111010111
111111111011111
110011111111110
111111111010111
111111101111111
110111011110101
111111101100111
111110111111111

```

15x26_i22_p0.8

```

0100010110001000
0000100000001000
0100010000000010
0001000000000100
0010000000001000
000001001001100
0001011100000000
0100100000010100
0100000100001011
0000010000000000
0001111000000010
1000001001001000
0001000000100000
0000000100000000
0000001000000010
0000010000000000

```

16x16_i23_p0.2

```

1001000000100000
0000001000000000
0000001001000001
0110000000010010
0000100000001000
0010000000001001
0100000000100000
0000000100000110
0000000010100100
0010000000000000
0010110001000010
0011000000001001
0000000010000000
0010000000001001
1000000010000000
1000011000000110
0000110010000000
0001000001010110

```

16x18_i24_p0.2

```

1111111101111011
1001111100101011
1111010110111101
1111101111101101
1100011110110101
1010111111111110
1010110111111011
1111111110111101
1001101111011111
1110111111110011
1101101111101110
1111001101110110
1111101011101011
1111100101111111
1111111101111111
1011111111011001
1101011111111101
1111111110011110
0111110111101100
1011011111111101
0011111111111111
1101111110111111
1110111110111101
1101111111101111

```

16x29_i25_p0.8

	111110111111101101	00000000100000010000000
	1011011110000101111	10000001000001000001001
	1111101111101110001	0100001001000000000100
	011111101111101111	010001001000100000001
	011111111110111111	00000001000000000000
	1111101110111111011	00000001000000000000
	1011001111101111011	1000100111000000000000
	1101101111110111001	000100000000000010011
	1110111111110101111	000000001000100000110
	010111010111101111	001000101000001100000
	1111011100111100111	010100100001000000000
	1111010111101111111	01001010000000000100
	1111011000111111101	00000000000000001010
	1111011111111100111	00100000000100000000
	0110101110111111010	010010101000000000100
	1111111111111111111	000000000000000001010
	1111111011111101111	00100000000100000000
	0011111111111111111	0000000101100000010000
	1010111101111111111	010000100001000000000
	1010101110101011110	010101110000000100001
	1011001111110001111	000001010000100101110
	1111111111101111101	00000010100010010000
	1011111101111001111	100010000000000000000
	1101111111111110111	001000010100010100000
	1111011111111110111	010100000000001000000
	1111100100010100011	001010001000000000000
	1111101011111111111	100010111000010010010
	1111110011111001111	001000000001100101000
	0111110000000111111	0110010000100000000100

17x18_i26_p0.2

17x29_i27_p0.8

20x28_i28_p0.2

```

00000000010101000010000
00100000100001001000100
0001000001000010000101
0000111100000010000000
0100001110000101000110
010000000001001001001
1001000001000010000100
0000000011000010000010
0010000100000000010001
000100000000100010000
100100100100000011000
1000000000001001100110
1000000000001000001000
0010000000001100000000
0010000000100000000110
10000000010100000100000
1000000000000010100000
01000000100000000010000
00000000101000000100000
00000000100110001000000
0001000000010011101100
00000001110100001100000

```

21x22_i29_p0.2

```

10111111101011111010111
11011111101011101111111
1011111111101111111000
1111111111111111111110
1010010111111011111011
10111111100111111111111
11101111110111101011111
11111111101111111111111
00111111111111101101101
11011111100111110111111
11100111111011111001011
11111111111111101111111
111111101000111111101
1111111111111011110001
11110011111111111111111
1100111011101111110010
1010110111111011111011
10111110110101010010111
1110111111111101010110
1011111011111111111110
1011000111010111111001
1101101010111110101111

```

21x22_i30_p0.8

																1 1 1 1 1 1 1																																
																1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6																																
																1	0 0 0 0 0 1 1 1 1 0 1 0 0 0 0 0 0 0																															
																2	0 1 0 0 0 1 0 1 1 1 0 0 0 0 0 1 0 1																															
																3	0 0 0 0 0 0 0 0 1 0 0 1 0 1 0 0 0 0																															
																4	0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0																															
																5	0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 1 0																															
																6	0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0																															
																7	0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1																															
																8	0 0 0 0 1 1 0 1 0 0 0 0 0 0 0 0 0 0																															
																9	0 0 0 1 1 0 0 1 1 0 0 1 0 0 0 0 0 0																															
																10	0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1																															
																11	0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0																															
																12	0 0 0 0 0 1 0 1 0 1 0 0 0 0 0 0 0 0																															
																13	0 0 0 0 0 1 1 0 0 1 0 0 0 0 0 0 0 0																															
																14	0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 1 0																															
																15	0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0																															
																16	0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0																															
																17	0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0																															
																18	0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1																															
																19	0 0 0 1 1 1 0 1 0 0 0 0 0 0 0 0 1 0																															
																20	0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0																															
																21	0 0 0 1 1 0 0 1 0 0 0 0 0 0 0 0 1 0																															
																22	0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0																															
																23	0 0 0 0 1 1 1 0 1 0 0 0 0 0 0 0 0 0																															
																24	0 0 0 0 0 0 0 0 1 0 0 1 1 1 0 0 0 0																															
																25	0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0																															
																26	0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0																															
																27	0 0 0 0 0 0 0 1 0 0 1 1 0 0 0 0 0 0																															
																28	0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																															
																29	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																															
																30	0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0																															
																31	0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0																															
																32	0 1 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 1																															
																33	0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 0																															
																34	0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0																															
																35	0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0																															
																36	0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																															
																37	1 1 0 0 0 1 0 1 1 0 0 0 0 0 0 0 0 1																															
																38	0 1 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1																															
																39	0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0																															
																40	0 1 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0																															
																41	0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0																															
																42	1 1 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0																															
																43	0 0 0 0 0 1 1 0 1 0 0 0 0 0 0 0 0 1																															
																1 1 1 1 1 1 1																																
																1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6																																
																1	0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																															
																2	0 0 0 1 0 0 1 1 0 0 1 1 0 0 0 0 0 0																															
																3	0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0																															
																4	1 1 0 1 0 0 1 1 0 0 0 1 0 0 0 0 0 0																															
																5	1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0																															
																6	0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 1 0 1																															
																7	1 0 0 1 0 0 1 1 0 0 1 1 0 0 0 0 1 1																															
																8	0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 1 1																															
																9	1 0 0 1 0 0 0 1 0 0 1 0 0 0 0 0 0 0																															
																10	0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0																															
																11	0 0 0 0 1 0 0 0 0 1 1 0 0 1 0 0 0 0																															
																12	1 0 0 0 0 0 1 1 0 0 1 1 0 0 0 0 0 0																															
																13	0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0																															
																14	0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 1 1																															
																15	0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 1 1																															
																16	0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																															
																17	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																															
																18	1 1 0 1 0 0 1 1 0 1 1 0 0 0 0 0 0 0																															
																19	0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0																															
																20	0 1 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0																															
																21	0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0																															
																22	1 0 0 1 0 0 1 1 0 0 0 1 0 0 0 0 0 0																															
																23	0 0 1 0 0 1 0 0 1 0 0 0 0 0 0 0 1 0																															
																24	0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 1																															
																25	0 0 1 0 0 1 1 0 1 0 0 0 0 0 0 1 0																															
																26	0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 1 1																															
																27	0 0 1 0 0 1 0 0 1 0 0 0 0 0 1 1 1																															
																28	0 0 1 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0																															
																29	0 0 1 0 0 0 1 0 0 0 0 0 0 0 1 0 1 0																															
																30	1 0 0 1 0 0 1 0 0 0 0 1 0 1 0 1 0 0																															
																1 1 1 1 1 1 1																																
1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6																																																
1	1 0 1 0 0 0 0 1 1 0 0 0 0 0 0 1 0 0																																															
2	0 0 0 0 0 1 0 0 0 0 1 0 1 1 0 0 0 0																																															
3	1 0 0 0 0 1 0 1 0 0 0 0 1 0 0 0 0 0																																															
4	1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0																																															
5	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0																																															
6	1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0																																															
7	1 0 0 0 0 0 0 0 1 1 0 1 0 1 0 0 0 0																																															
8	1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0																																															
9	0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 1 1																																															
10	1 1 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0																																															
11	1 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1																																															
12	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0																																															
13	0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0																																															
14	0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 1 0 1																																															
15	1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0																																															
16	0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0																																															
17	0 1 1 0 0 1 0 0 0 1 0 0 0 0 0 0 0 1																																															
18	1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0																																															
19	1 0 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0																																															
20	0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 1 1																																															
21	0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0																																															
22	1 1 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0																																															
23	1 1 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0																																															
24	0 0 0 0 0 0 0 1 0 1 1 1 0 1 0 1 0 1																																															

McCormick1972

Srinivasan1990

King1980

1 1 1 1 1 1 1 1 1																	
1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8																	
1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0
2	0	0	1	1	0	1	0	0	1	0	0	0	0	0	0	0	0
3	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0
4	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	1	1
5	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0
8	0	0	1	1	1	0	0	0	0	0	1	0	0	0	0	1	0
9	0	0	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0
10	1	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0
11	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1
12	0	0	1	1	0	1	0	0	1	0	0	0	0	0	0	0	0
13	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0
14	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
15	0	0	1	1	1	0	0	0	0	0	0	0	0	0	1	0	0
16	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1
17	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	1	0
18	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0
19	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0
20	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0
21	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
22	0	0	0	1	0	0	0	0	0	0	1	1	0	0	0	0	0
23	1	1	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0
24	0	0	0	0	1	1	0	1	1	0	0	0	0	0	0	0	0

1 1 1 1 1 1 1 1 1 1 2																				
1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0																				
1	1	0	1	1	1	1	0	1	0	0	0	1	1	1	1	0	0	1	1	0
2	0	0	1	0	0	1	0	0	1	1	0	1	0	1	0	0	0	0	0	0
3	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1
4	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
5	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
8	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0
10	1	0	0	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
11	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0	0	0
12	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
13	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
15	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
21	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
22	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
23	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

Carrie1973

Mosier1985b

Kumar1986

											1	1	1	1	1	1	1	1	2	
	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0
1		1	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	1	0	0
2		0	1	0	1	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0
3		0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0
4		0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	1	0	1	0
5		0	0	1	0	1	0	1	1	0	0	0	0	0	0	0	0	0	1	0
6		1	0	0	0	0	0	0	0	0	0	1	1	0	0	1	1	0	0	0
7		0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0
8		0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0
9		0	0	0	0	0	0	0	1	0	0	1	1	0	0	1	1	0	0	1
10		0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
11		1	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	1	0
12		0	1	0	1	0	0	1	0	0	0	0	0	1	1	0	1	0	1	0
13		0	1	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	1	0
14		1	0	0	1	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0
15		1	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	1	0
16		0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
17		0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	1	0
18		1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0
19		0	1	0	0	0	1	1	0	1	1	0	0	0	0	0	0	0	0	0
20		1	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	1
21		0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	1	0
22		0	0	0	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0
23		1	0	0	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0	1
24		1	1	0	1	0	0	1	0	0	0	0	0	1	1	0	0	0	1	0
25		1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0
26		0	0	0	1	0	1	0	0	1	0	0	0	0	0	1	0	0	0	1
27		0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
28		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
29		0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
30		1	0	0	0	0	1	1	0	0	0	0	0							

Boe1991

[illegible]

Chandrasekharan1989 1

[illegible]

Carrie1973

[illegible]

Chandrasekharan1989 2

[illegible]

Chandrasekharan1989 5

[illegible]

Chandrasekharan1989 7

```

1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2
1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7
1 1 1 0 1 1 0 1 1 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0
2 1 1 0 1 0 0 1 1 0 0 0 0 1 0 0 1 0 0 1 0 0 1 0 0 0 0 0 0 0
3 0 0 1 1 1 0 1 1 0 0 0 0 0 1 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0
4 1 1 1 1 0 0 0 0 0 0 0 0 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
5 1 0 1 0 1 0 1 1 1 1 0 0 0 0 0 1 0 0 1 0 1 1 0 0 0 0 0 0
6 0 0 0 0 0 1 0 0 1 1 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 1 1
7 1 1 1 0 1 0 1 1 1 1 1 0 1 1 0 1 1 0 1 0 0 0 0 0 1 0 0 0 0 0
8 1 1 1 0 1 0 1 1 1 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
9 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
10 0 0 0 0 1 1 1 1 1 1 1 1 1 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
11 0 0 0 0 0 0 1 0 0 1 1 1 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0
12 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
13 0 1 0 1 0 0 1 0 0 1 0 1 1 1 0 0 1 0 0 1 0 0 1 0 0 0 0 0 0
14 0 0 1 1 0 0 1 0 0 1 0 1 0 1 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0
15 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 1 1 0 0 0 0 0 1
16 0 1 1 1 1 0 1 1 0 0 0 0 1 1 1 1 1 1 1 0 1 1 0 0 0 0 0 0
17 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0 1 1 0 1 0 0 0 0 1
18 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 1 0 0 1 0 0
19 0 1 1 1 1 0 0 0 0 1 0 1 1 1 0 1 0 0 1 0 0 1 0 0 0 1 0 0 0 0
20 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 1 0 0 1 1 0 0 1 0 1 0
21 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 1 1 0 0 1 0 1 1
22 1 1 0 0 1 0 1 0 0 0 0 0 1 0 1 1 0 1 1 0 0 1 0 0 0 0 0 0 0
23 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 1
24 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 0 1 1
25 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 1
26 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 0 1 0
27 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 1 0 1 1 1 0 1

```

McCormick1972b

```

1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8
1 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 1 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0
2 1 1 0 0 0 0 0 0 1 1 1 1 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
3 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 1 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0
4 1 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
5 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0
6 0 0 0 0 1 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0
7 1 0 0 0 1 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
8 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0
9 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0
10 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 1 0 0 0 0 0 0 0
11 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 0 0 0 0 0 0 0
12 1 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
13 0 0 1 0 0 1 1 1 1 0 1 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
14 0 0 1 1 1 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0
15 0 0 1 1 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
16 0 0 1 0 0 1 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
17 0 0 1 0 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0
18 0 0 1 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
19 0 0 1 1 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
20 0 0 1 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
21 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
22 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0
23 0 0 1 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0
24 0 0 1 1 0 0 0 1 1 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0
25 0 0 1 1 1 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0
26 0 0 1 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0
27 0 0 1 0 0 1 1 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0
28 0 0 1 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
29 0 0 1 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
30 0 0 1 1 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
31 0 0 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
32 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
33 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
34 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0
35 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0
36 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0
37 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
38 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
39 0 0 0 0 0 0 0 0 0 0 1 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
40 0 0 0 0 0 0 0 0 0 1 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0
41 0 0 0 0 0 0 0 0 0 1 0 0 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
42 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0
43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0
44 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
45 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
46 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

```

Carrie1973c

```

1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 3
1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0
1 00000000100000000000011000000000011
2 00000000001000000000000000000100000000
3 0000000000100000000000011000000000011
4 000000000000000010000000000001000000
5 000010000000000100000000000000000000
6 000001000000000001000000000000000000
7 000100000000000000010000000000000000
8 0000000010000000000000000000000001100
9 0000000010000000000000000000000000010
10 10000000000100000000000110000000000
11 010000000000100000000000000000000000
12 011000000001010000000000011100000000
13 000000000100000000000000000000000010
14 000000001000000000001000000000000000
15 0001000000000000000000000000000001000
16 0000001000000000000110000000010000
17 000010000000000001000000000000000000
18 010000000000011100000000000000000000
19 000000000000010000000000000000000000
20 000000000010000000000000000000000000
21 00000000010000000000011000000000011
22 000000000000000000000011000000000001
23 001000000001010000000000011000000000
24 000100000000010000000000000000000000
25 000100000000010000000000000000000000
26 000001000000000100000000000000000000
27 000000100000000001100000000000000000
28 000000000000000000000000000000000000
29 000000000010000000000000000000000000
30 00000000000000000000011000000000011
31 101000000000000000000000011100000000
32 111000000000000000000000011000000000
33 110000000001000000000001000000000000
34 000010100000000000000100000000100000
35 000100000000000000000000000000000100
36 000010100000000000010000000001000000
37 000000000000000001100000000000000000
38 000000000000010000000000010000000000
39 111000000001110000000000011000000000
40 001000000000010000000000110000000000
41 110000000000100000000000000000000000

```

Kumar1987

```

1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 3
1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0
1 1001100001010101000000000000000000000
2 0110100001000000000000000000000000000
3 0100100001001000000000000000000000000
4 0010000100110000000000000000000000000
5 1000001010000010000000000000000000000
6 1010000001100100000000000000000000000
7 0001100010111000000000000000000000000
8 1010000000000100000000000000000000000
9 0100001100010001000000000000000000000
10 0000100000100000000000000000000000000
11 0010100000100000000000000000000000000
12 0000000110000000000000000000000000000
13 0000001110000000000000000000000000000
14 0000001010000100000000000000000000000
15 0000001010000100000000000000000000000
16 0000001010000100000000000000000000000
17 0100000001000010000000000000000000000
18 1001000000110000000000000000000000000
19 0000000000000001010000000000000000000
20 0000000000000000011010000000000000000
21 0000000000000000001010000000000000000
22 0000000000000000000110000000000000000
23 00000000000000000000101100000000000000
24 00000000000000000000101000000000000000
25 00000000000000000001101000000000000000
26 00000000000000000000101000000000000000
27 00000000000000000000011000000000000000
28 00000000000000000000000111000000000000
29 000000000000000000000000011001000000000
30 000000000000000000000000010011000000000
31 0000000000000000000000000010101000000000
32 000000000000000000000000000111000000000
33 000000000000000000000000000010101000000000
34 0000000000000000000000000000011001000000000
35 000000000000000000000000000001011000000000
36 0000000000000000000000000000010110000000000
37 00000000000000000000000000000101010000000000
38 00000000000000000000000000000101010000000000
39 00000000000000000000000000000000000001000111
40 0000000000000000000000000000000000000010111
41 000000000000000000000000000000000000000101010
42 000000000000000000000000000000000000000011000001
43 00000000000000000000000000000000000000000101000
44 0000000000000000000000000000000000000000001001001
45 0000000000000000000000000000000000000000000100110
46 0000000000000000000000000000000000000000000010001
47 0000000000000000000000000000000000000000000011000
48 0000000000000000000000000000000000000000000001000011
49 0000000000000000000000000000000000000000000000111000
50 00000000000000000000000000000000000000000000001000010

```

Stanfel1985_1

										111111111122222222223									
										123456789012345678901234567890									
										1	00								

```

1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3
1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7
1 0001010111100111111111010011001110011100010
2 001101111110111111111101011111111010110
3 000101010110011011111101001000101010100
4 01101101001011101111110100100001011101
5 00010001011001001111110100100011000000
6 00110111011011111111110101101111010000
7 00001001001001101111110100101001010100
8 00010001011001101111110100100011000000
9 00110001011001101111110100100011000000
10 00000001001011101111110100100001010100
11 01001001001011111111110100100001010101
12 00000001011001101111110100100001010000
13 00001001001011101111110100100001010100
14 00110111001011111111110101101101010100
15 00001001001011111111110100100001010101
16 00110111001011111111110101101101010100
17 01101111001011111111110100100001011101
18 01111111001011111111110101101101011101
19 00001001001011101111110100100001010100
20 01101101001011101111110100100001011101
21 01001001001011101111110100100001010101
22 00000001001001101111110100100001010100
23 01101101001011101111110100100001011101
24 00010001111001111111110100110011110010
25 00110001111011111111110100110011000010
26 1011001111100100111011010111010110000
27 1011001111100100111011010110010100000
28 10110111111001011110110111111110100000
29 1001000101100100100011010111010100000
30 1001001111100100111011010111010100000
31 1011001111100100111011010111010100000
32 1011001111100100111011010111010100000
33 10110111111001011110110111111100100000
34 1001000101100100100011010111010100000
35 10110011111001001110110111111010100000
36 10110111111001011110110111111110100000
37 10110011111001011110110111111010100000
38 10110111111001011110110111111100100000
39 1001000101100100100011010111010100000
40 0001000001110100100010010111011100000
41 00000001001011101111110100100001010001
42 01000001001011101111110100100001010101
43 00010001111001111111110100110011100000
44 00010001001011101111110100100001010100
45 00000001011001001111110100100001010100
46 0001000101110100100010110111011100000
47 0001000101100100111010100100011000000
48 0001000101100100111010100100011000000
49 0001000101110100100010110111011100000
50 0001000101110100100010110111011100000
51 0001000111100100111010100111001000000
52 0001000101100100100010100100011000000
53 001100010111001001111010110111011100000

```

McCormick1972c

100 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0

Apêndice B

Soluções Ótimas do PBEA

```
01010000001
01010000001
00000100000
001010101010
001010101010
001010101010
00000100000
001010101010
00000000000
00000100000
00000001000
```

10x11_i01_p0.4

```
10000000010
00101000100
00010100000
00101000100
01000000000
00101000100
00000010001
00101000100
00000010001
1000000010
00000010001
```

10x11_i02_p0.4

```
00111000000
00111000000
00000100000
00000100000
01000010001
1000000110
00000000000
01000010001
01000010001
01000010001
00111000000
00000000000
00000100000
1000000110
1000000110
00111000000
00111000000
1000000110
```

10x18_i03_p0.4

```
10000011001
00011000110
00100100000
00100100000
00011000110
10000011001
00011000110
00011000110
00011000110
10000011001
10000011001
10000011001
00011000110
00011000110
```

11x13_i04_p0.4

```
000100000000
000001100000
10001000011
01000001000
00000110000
00100000100
00000110000
01000001000
10001000011
00010000000
00000110000
00010000000
10001000011
00010000000
00000110000
```

11x15_i05_p0.4

```
11000000100
00001000010
11000000100
11000000100
11000000100
11000000100
00000100001
00001000010
11000000100
00110000000
00000100001
00110000000
00110000000
00001000010
00001000010
00000110000
00110000000
```

11x16_i06_p0.4

```
000000000000
101110000010
01000110001
10111000010
10111000010
01000110001
01000110001
10111000010
00000000100
01000110001
10111000010
01000110001
00000000100
00000000100
00000000100
00000000100
10111000010
10111000010
```

11x18_i07_p0.4

```
100001011010
000000000000
001000000000
000000000100
000100000000
100001011010
000010000001
000100000000
100001011010
010000100000
100001011010
010000100000
001000000000
```

12x13_i08_p0.4

01001100010000	11000010000000	0000000010010	00001000110000
00000100000000	00000001000000	00111100000100	100000100000001
01001100010000	0001100000101	0011100000100	000000010000010
0000000010011	000000001010	0000010000001	000000010000010
00000000000000	0010000010000	0100000000000	100000100000001
10010010000000	1100010000000	0100000000000	00001001100000
0100100010000	0001100000101	0011100000100	011000000000000
10010010000000	1100010000000	0011100000100	00010000000100
0100100010000	000000001010	00000010000001	000000010000010
00000000000000	000000001010	10000001010000	000000000001000
00000000000000	0000000100000	00000010000001	000100000000100
00000000000000	1100001000000	00000010000001	000100000000100
00000000000000	000000001010	0011100000100	011000000000000
00000000000000	0001100000101	00000010000001	100000100000001
00000000000000	0001100000101	1000000101000	100000100000001
00000000000000	1100010000000	1000000101000	000100000000100
00000000000000	0000000100000	0000000100010	100000100000001
00000000000000	1100001000000	0100000000000	011000000000000
00000000000000	1100001000000	0100000000000	000100000001000

12x13_i09_p0.4

11x18_i07_p0.4

12x13_i08_p0.4

13x19_i12_p0.4

0010000010000100	000010010101000000	00000000011001100
0010000010000100	000010010101000000	00000000000000000
00000110010100001	000010010101000000	00000000011001100
0010000010000100	011000010101000000	00110000100010010
0000110010100001	00010000000000010	00110000100010010
1100000000001000	00000000000010100	10001000000000000
0001001000000000	00000000000010100	00000011000010000
0001001000000000	10000100000001001	00000000011001100
0001001000000000	01100001010100000	01000000000000000
0010000010000100	10000100000001001	10001000000000000
0000000000100000	10000100000001001	00110000100010010
00000110010100001	10000100000001001	00000110001000000
1100000000001000	00001001010100000	00000000011001100
0010000010000100	10000100000001001	00000000011001100
0000110010100001	10000100000001001	00000000011001100
0000000000100000	00000000000000000	00000011000010000
0000110010100001	00001001010100000	00110000100010010

15x16_i13_p0.4

16x17_i14_p0.4

16x17_i15_p0.4

00010111110	1000111011	101100000111	111111111110
00010111110	1000111011	010000010000	00000000000001
00010111110	1000111011	010000010000	1111111111110
00010111110	0111000100	101100000111	1111111111110
00010111110	0111000100	101100000111	1111111111110
00010111110	1000111011	101100000111	00000000000001
00010111110	1000111011	101100000111	1111111111110
00010111110	0111000100	101100000111	1111111111110
111000000000	1000111011	000001010000	00000000000001
00010111110	0111000100	101100000111	1111111111110
111000000000	1000111011	101100000111	1111111111110

10x11_i16_p0.6

10x14_i17_p0.6

11x11_i18_p0.6

12x14_i19_p0.6

```

1 1 1 0 0 1 1 1 0 1 1 0
0 0 0 1 1 0 0 0 1 0 0 1
1 1 1 0 0 1 1 1 0 1 1 0
1 1 1 0 0 1 1 1 0 1 1 0
1 1 1 0 0 1 1 1 0 1 1 0
0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 0 0 1 1 1 0 1 1 0
1 1 1 0 0 1 1 1 0 1 1 0
0 0 0 1 0 0 0 0 0 0 0 0
1 1 1 0 0 1 1 1 0 1 1 0
1 1 1 0 0 1 1 1 0 1 1 0
0 0 0 0 1 0 0 0 1 0 0 1
0 0 0 0 1 0 0 0 1 0 0 1
0 0 0 0 1 0 0 0 1 0 0 1
1 1 1 0 0 1 1 1 0 1 1 0

```

12x15_i20_p0.6

[illegible]

15x25_i21_p0.8

[illegible]

15x26_i22_p0.8

```

000000010000000000
000001000000000000
010000000000000010
000000000000000100
001000000000000000
000000010010010000
000000001000000000
000010000000000000
010000000000000010
000001000000000000
000010000000000000
000000010010010000
000100000010000000
000000001000000000
000001000000000000
000000001000000000
000001000000000000

```

16x16_i23_p0.2

```

1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 1
0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0
0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 1
0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 1
0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0
0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1

```

16x18_i24_p0.2

[illegible]

16x29_i25_p0.8

Apêndice C

Soluções Ótimas para o PFCM

C.1 Soluções sem Restrição *Singleton*

	2 3	1 4	5
1	1 1	0 0	1
3	1 1	0 0	0
2	0 0	1 1	0
4	0 0	1 1	0
6	0 1	1 1	0
5	0 0	1 0	1
7	0 1	0 0	1

King1982

	1	2 3 4 5
1	1	0 0 1 0
6	1	0 1 0 1
7	1	0 0 0 0
2	0	1 0 1 1
3	0	1 1 1 0
4	0	1 1 1 1
5	1	1 1 0 1

Waghodekar1984

	2 3 5	1 4
4	1 1 1	0 0
7	1 1 0	0 0
10	1 1 1	0 0
15	1 1 1	0 0
18	1 1 1	0 0
1	1 0 0	1 1
2	0 0 0	1 1
3	1 0 0	1 1
5	0 0 0	1 1
6	1 0 0	1 1
8	1 0 0	1 1
11	1 0 0	1 1
12	1 0 0	1 1
13	1 0 0	1 1
14	0 0 0	1 1
16	0 0 0	1 1
17	0 0 0	1 1
9	0 0 1	0 0

Seifoddini1989

	1 4 6	2 3 5
4	1 1 1	0 0 0
7	1 1 1	1 0 0
1	0 0 0	1 0 1
3	0 0 0	1 1 1
5	0 0 0	1 0 1
6	0 0 0	1 1 1
8	0 0 0	1 1 1
2	1 0 0	1 0 0

Kusiak1992

	1 2	3 4	5	6 7
1	1 0	1 0	0	0 0
2	1 1	0 0	0	0 0
6	1 1	0 0	0	0 0
9	0 1	0 0	0	0 0
3	0 0	1 1	1	0 0
7	0 0	1 1	0	0 0
4	0 0	0 0	1	1 0
11	0 0	1 0	1	0 0
5	0 0	0 0	0	1 1
8	0 0	0 0	0	0 1
10	0 0	0 0	0	1 1

Boctor1991

	1	2 3	4 7	5	6
2	1	0 0	0 0	0	0
7	1	0 0	0 1	0	0
11	0	1 1	0 0	0	0
3	1	0 0	1 1	0	0
6	0	0 0	1 1	0	0
5	0	1 0	0 0	1	0
8	0	0 0	0 0	1	1
1	0	1 0	1 0	0	1
4	0	0 0	0 1	0	1
9	0	0 0	0 1	0	1
10	0	0 1	0 0	0	1

Kusiak1987

	1	2 3	4 5 6	7 8
1	1	1 0	0 0 0	0 0
2	1	0 0	0 0 0	0 0
3	1	1 1	0 0 0	0 0
4	1	1 1	0 0 0	0 0
5	0	1 1	0 0 0	0 0
6	0	1 1	1 0 0	0 0
7	0	1 1	1 1 1	0 0
8	0	0 1	1 1 1	0 0
9	0	0 1	1 1 1	0 0
10	0	1 0	1 1 0	0 0
11	0	0 0	0 0 1	1 1
12	0	0 0	0 0 0	1 1

Seifoddini1986

	5 6	2 4 7 8	1 3
1	1 1	0 0 0 0	0 0
5	1 1	0 0 0 0	0 0
10	1 1	0 1 0 0	0 0
12	1 1	0 0 1 0	0 0
15	1 1	0 0 0 0	0 0
3	0 0	1 1 1 1	1 0
4	0 0	1 1 1 1	0 0
6	1 0	1 1 1 1	0 0
7	0 0	1 1 1 1	0 0
18	0 0	1 1 1 1	0 0
20	0 1	1 1 1 1	0 0
2	0 0	0 0 0 0	1 1
8	0 0	0 0 0 0	1 1
9	0 1	0 0 0 0	1 1
11	0 0	0 0 1 0	1 1
13	0 0	0 0 0 0	1 1
14	0 0	1 0 0 0	1 1
16	0 0	0 0 0 0	1 1
17	1 0	0 0 0 0	1 1
19	0 0	0 0 0 0	1 1

Chandrasekharan1986a

	1 2 4 6 8	3 5 7
1	1 0 0 1 1	0 1 0
2	0 1 0 1 1	0 1 0
3	1 1 1 0 1	0 1 0
4	1 1 1 0 1	0 0 0
7	0 1 1 1 0	1 0 1
8	0 0 1 1 1	1 0 1
9	1 1 1 0 1	0 0 0
10	1 0 1 1 1	0 1 0
14	1 0 1 1 1	0 0 0
15	1 0 1 1 0	0 1 0
18	1 1 1 1 0	0 0 0
5	0 0 0 1 1	1 0 1
6	0 1 0 0 0	1 1 1
11	0 1 0 1 0	1 0 1
12	0 0 0 1 0	1 1 1
13	0 0 1 0 1	1 1 1
16	1 0 0 1 1	1 1 1
17	0 0 1 0 1	1 1 1
19	1 0 1 1 0	1 1 1
20	0 1 1 0 0	1 1 1

Chandrasekharan1986b

	1 4	2 7 9 0	3	5 6	8
1	1 1	0 0 0 0	0	0 1	0
2	0 0	0 0 1 1	0	0 0	0
3	0 0	1 1 1 1	0	0 0	0
4	0 0	1 0 1 1	0	0 0	0
8	0 0	1 1 1 0	0	0 0	0
5	0 0	0 0 0 0	1	0 0	0
6	0 0	0 0 0 0	1	0 0	1
7	0 0	0 0 0 0	0	1 1	0
10	1 0	0 0 0 0	0	1 1	0
9	0 0	0 0 0 0	0	0 0	1

Mosier1985a

	1 7 0	3 4 6 9	2 5 8
2	1 1 1	0 0 0 0	0 0 0
10	1 1 1	0 0 0 0	0 0 0
7	0 1 1	0 0 0 0	0 0 0
11	1 1 1	0 0 0 0	0 0 0
12	1 1 1	0 0 0 0	0 0 0
1	0 0 0	1 1 1 0	0 0 0
4	0 0 0	0 1 1 1	0 0 0
6	0 0 0	1 0 1 1	0 0 0
9	0 0 0	1 1 1 1	0 0 0
14	0 0 0	1 1 1 1	0 0 0
3	0 0 0	0 0 0 0	1 1 1
5	0 0 0	0 0 0 0	1 1 1
8	0 0 0	0 0 0 0	1 1 1
13	0 0 0	0 0 0 0	1 1 1
15	0 0 0	0 0 0 0	1 1 1

Chan1982

	1 1 2 3	1 2 3 1	4 5 7	6 8 9	1 0	1 4
7	1 0 1	0 0 0	0 0 0	0 0 0	0	0
8	1 1 1	0 0 0	0 0 1	0 0 0	0	0
9	0 1 1	0 0 0	0 0 0	0 0 0	0	0
4	0 0 0	1 1 1	0 0 0	0 0 0	1	0
5	0 0 0	1 1 1	0 0 0	0 0 0	0	0
21	0 0 0	0 1 1	0 0 0	0 0 0	0	0
2	0 0 0	0 0 0	1 1 1	0 0 0	0	0
3	0 0 0	0 0 0	1 1 1	0 0 0	0	0
17	0 0 0	0 0 0	1 1 1	0 0 0	0	0
20	0 0 0	0 0 0	1 1 1	0 0 0	0	0
22	0 0 1	0 0 0	1 1 0	0 0 0	0	0
1	0 0 0	0 0 0	0 0 0	1 1 0	0	0
6	0 0 0	0 0 0	0 0 0	0 1 1	0	0
10	0 0 0	0 0 0	0 0 0	1 1 1	0	1
12	0 0 0	0 0 0	0 0 0	1 1 1	0	0
14	0 0 0	0 0 0	0 0 0	1 1 0	0	0
15	0 0 0	0 0 0	0 0 0	1 1 1	0	1
16	0 0 0	0 0 0	0 0 0	1 1 0	0	0
23	0 0 0	0 0 0	0 0 0	0 0 0	1	0
11	0 0 0	0 0 0	0 0 0	1 0 0	0	1
13	0 0 0	0 0 0	0 0 0	0 0 1	0	1
18	0 0 1	0 0 0	0 0 0	0 0 0	0	0
19	0 0 0	0 0 0	1 0 0	0 0 0	0	0

Askin1987

	1 1 1 2 3	1 2 3 1	4 5 7	6 8 9	1 0	1 4
6	1 0 1	0 0 0	0 0 0	0 0 0	0	0
7	1 1 1	0 0 0	0 0 1	0 0 0	0	0
8	0 1 1	0 0 0	0 0 0	0 0 0	0	0
3	0 0 0	1 1 1	0 0 0	0 0 0	1	0
4	0 0 0	1 1 1	0 0 0	0 0 0	0	0
21	0 0 0	0 1 1	0 0 0	0 0 0	0	0
1	0 0 0	0 0 0	1 1 1	0 0 0	0	0
2	0 0 0	0 0 0	1 1 1	0 0 0	0	0
17	0 0 0	0 0 0	1 1 1	0 0 0	0	0
20	0 0 0	0 0 0	1 1 1	0 0 0	0	0
23	0 0 1	0 0 0	1 1 0	0 0 0	0	0
5	0 0 0	0 0 0	0 0 0	0 1 1	0	0
9	0 0 0	0 0 0	0 0 0	1 1 1	0	1
10	0 0 0	0 0 0	0 0 0	1 1 0	0	0
12	0 0 0	0 0 0	0 0 0	1 1 1	0	0
14	0 0 0	0 0 0	0 0 0	1 1 0	0	0
15	0 0 0	0 0 0	0 0 0	1 1 1	0	1
16	0 0 0	0 0 0	0 0 0	1 1 0	0	0
22	0 0 0	0 0 0	0 0 0	1 1 0	0	0
24	0 0 0	0 0 1	0 0 0	0 0 0	1	0
11	0 0 0	0 0 0	0 0 0	1 0 0	0	1
13	0 0 0	0 0 0	0 0 0	0 0 1	0	1
18	0 0 1	0 0 0	0 0 0	0 0 0	0	0
19	0 0 0	0 0 0	1 0 0	0 0 0	0	0

Stanfel1985

	1 1 1 4 7 8 1 2	2	1 3 6 9 5	1 1 1 5 0 4 6	1 3
2	0 1 1 1 1 1	0	0 0 0 0	0 0 0 0	0
4	1 1 1 1 0 1	1	0 0 0 0	0 0 0 0	0
7	1 1 1 1 1 1	0	0 0 0 0	0 0 0 1	0
9	1 1 0 1 1 0	0	0 0 0 0	0 0 0 0	0
12	1 0 1 1 1 1	0	0 0 0 0	0 0 0 0	0
18	1 1 1 1 1 1	1	0 0 0 0	0 1 0 0	0
22	1 1 1 1 0 1	0	0 0 0 0	0 0 0 0	0
30	1 1 1 0 1 0	0	0 0 0 1	0 0 0 0	1
1	0 0 0 0 0 0	1	1 0 0 0	0 0 0 0	0
10	0 0 0 0 0 0	1	0 0 0 0	0 0 0 0	1
16	0 0 0 0 0 0	1	0 0 0 0	0 0 0 0	0
5	1 0 0 0 0 0	0	0 0 1 1	0 0 0 0	0
23	0 0 0 0 0 0	0	1 1 1 1	0 0 0 0	0
25	0 0 1 0 0 0	0	1 1 1 1	0 0 0 0	0
27	0 0 0 0 0 0	0	1 1 1 1	0 0 1 1	0
28	0 0 0 1 0 0	0	1 0 1 0	0 0 0 0	0
29	0 0 0 0 0 0	0	1 1 0 1	0 0 0 0	1
6	0 0 0 0 0 0	0	0 0 0 0	1 1 1 1	0
8	0 0 0 0 0 0	0	0 0 0 0	1 1 1 1	0
11	0 0 0 0 1 0	0	0 0 0 0	1 1 1 0	0
14	0 0 0 0 0 0	0	0 0 0 0	1 1 1 1	0
15	0 0 0 0 0 0	0	0 0 0 0	1 1 1 1	0
21	0 0 0 0 0 0	0	0 0 0 0	1 1 1 0	0
24	0 0 0 0 0 0	0	0 0 0 0	1 1 0 1	0
26	0 0 0 0 0 0	0	0 0 0 1	1 1 0 1	0
3	0 0 0 0 0 0	0	0 0 0 0	1 0 0 0	1
13	0 0 0 0 0 0	0	1 0 0 0	0 0 0 0	1
20	0 0 0 0 0 1	1	0 0 0 0	0 0 0 0	1
17	0 0 0 0 0 0	0	0 0 0 0	0 0 1 0	0
19	0 0 0 0 0 1	0	0 0 1 0	0 0 0 0	0

Srinivasan1990

	1 1 2 8	1 3 4	1 4 7	5	6	1 1 1 1 0 1 2 6	1 3	1 5	9
3	1 0 1	0 0	0 0	0	1	0 0 1 0	0	0	0
7	1 0 1	0 0	0 0	0	0	0 1 0 0	1	0	1
10	1 1 1	0 0	0 0	0	0	1 0 0 0	0	0	0
22	1 1 1	0 0	0 0	0	0	1 0 0 0	0	0	0
23	1 1 1	0 0	0 0	0	0	1 0 0 0	0	0	0
1	1 0 1	1 1	0 1	0	0	0 0 0 0	0	0	0
4	1 0 0	1 1	0 0	0	0	0 0 0 0	0	0	0
6	1 0 0	1 1	0 0	0	0	0 0 0 0	0	1	0
12	0 0 0	0 1	0 0	0	0	0 0 0 0	0	0	0
16	0 0 0	1 0	1 0	0	0	0 0 0 0	0	0	0
19	1 0 0	1 0	1 1	0	0	0 0 0 0	0	0	0
21	0 0 0	0 0	0 0	1	1	0 0 0 0	0	0	0
13	0 0 1	0 0	0 0	0	1	0 0 0 0	0	0	0
15	1 0 0	0 0	0 0	0	1	0 0 0 0	0	0	0
17	0 1 0	1 0	0 0	0	1	1 0 0 1	0	0	0
18	1 0 0	0 0	0 0	0	1	0 0 0 0	0	0	0
9	0 0 0	0 0	0 0	0	0	1 1 1 1	0	1	0
14	0 0 0	0 1	0 0	0	0	1 1 1 1	0	0	0
20	0 0 0	0 0	0 0	0	0	1 1 1 1	0	1	0
24	0 0 0	0 1	0 1	0	0	1 1 1 1	0	0	1
2	0 0 0	0 0	0 0	0	1	1 0 1 0	1	0	0
8	1 0 0	0 0	0 0	0	0	0 0 0 0	1	0	0
11	1 0 0	0 0	0 0	0	0	0 0 1 1	1	0	0
5	0 0 0	0 0	0 0	0	0	0 0 0 0	0	1	0

McCormick1972

	1 2 9 6	1 3 4	1 4 5 8 5	6	1 7 0	1 1 1 3	1 2	1
2	1 1 1	0 1	0 0 1 0	1	0 0	0 0	0	0
10	1 1 1	0 0	0 0 0 0	0	0 0	0 0	0	0
18	0 1 1	0 0	0 0 0 0	0	0 0	0 0	0	0
28	1 1 0	0 0	0 0 1 0	0	0 0	0 0	0	0
32	1 1 1	0 0	0 0 0 0	1	0 0	0 0	0	0
37	1 1 1	0 0	0 0 1 0	1	0 0	0 0	0	1
38	1 1 1	0 0	0 0 1 0	0	0 0	0 0	0	0
40	1 1 0	0 0	0 0 0 0	1	0 0	0 0	0	0
42	1 1 1	0 0	0 0 0 0	1	0 0	0 0	0	1
17	0 0 0	1 1	0 0 0 0	1	0 0	0 0	0	0
35	0 0 0	1 1	0 0 0 0	0	0 0	0 0	0	0
36	0 0 0	1 0	0 0 0 0	0	0 0	0 0	0	0
5	0 0 0	0 0	1 1 0 1	0	0 0	0 0	0	0
9	0 0 0	0 0	1 1 1 0	0	0 0	1 0	0	0
14	0 0 0	0 0	1 1 0 1	1	0 0	0 0	0	0
15	0 0 0	0 0	0 1 1 0	0	0 0	0 0	0	0
19	0 0 0	0 0	1 1 1 1	1	0 0	0 0	0	0
21	0 0 0	0 0	1 1 1 1	0	0 0	0 0	0	0
23	0 0 0	0 0	1 1 1 0	1	0 0	0 0	0	0
29	0 0 0	0 0	1 1 0 0	0	0 0	0 0	0	0
41	0 0 0	0 0	0 1 1 1	0	0 0	0 0	0	0
43	0 0 0	0 0	0 1 1 1	1	0 0	0 0	0	0
6	0 0 0	0 1	0 0 0 0	1	0 0	0 0	0	0
7	0 0 1	1 0	0 0 0 0	1	0 0	0 0	0	0
8	0 0 0	0 0	0 1 1 0	1	0 0	0 0	0	0
12	0 0 0	0 0	0 0 1 0	1	0 1	0 0	0	0
33	0 0 0	0 0	0 1 0 1	1	0 0	0 0	0	0
34	0 0 0	1 0	0 0 0 0	1	0 0	0 0	0	0
39	0 0 0	0 0	0 0 0 0	1	0 1	0 0	0	0
1	0 0 0	0 0	0 0 1 0	1	1 1	0 0	0	0
13	0 0 0	0 0	0 0 0 0	1	1 1	0 0	0	0
25	0 0 0	0 0	0 0 0 0	0	1 1	0 0	0	0
26	0 0 0	0 0	0 0 0 0	0	0 1	0 0	0	0
31	0 0 0	0 0	0 0 1 0	0	0 1	0 0	0	0
3	0 0 0	0 0	0 0 1 0	0	0 0	1 1	0	0
20	0 0 0	0 0	0 0 1 0	0	0 0	1 0	0	0
24	0 0 0	0 0	0 0 1 0	0	0 0	1 1	1	0
11	0 0 0	0 0	0 0 1 0	0	0 0	0 0	1	0
22	0 0 0	0 0	0 0 0 0	0	0 0	0 0	1	0
27	0 0 0	0 0	0 0 1 0	0	0 0	1 0	1	0
30	0 0 0	0 0	0 0 0 0	0	0 0	1 0	1	0
4	0 1 0	0 0	0 0 0 0	0	0 0	0 0	0	0
16	0 0 0	0 0	0 1 0 0	0	0 0	0 0	0	0

King1980

	1 2	3 4 5 6	7	8 9	1 1 0 3	1 1 1 1 2 5	1 4	1 1 6 7	1 8
23	1 1	0 0 1 1	0	1 0	0 0	0 0 0	0	0 0	0
2	0 0	1 1 1 0	1	0 0	1 0	0 0 0	0	0 0	0
5	0 0	1 1 1 1	1	0 0	0 0	0 0 0	0	0 0	0
8	0 0	1 1 1 1	0	0 0	0 1	0 0 0	0	0 0	0
9	0 0	1 1 1 1	0	0 0	0 1	0 0 0	0	0 0	0
10	1 0	0 0 1 1	0	1 0	0 0	0 0 0	0	0 0	0
12	0 0	1 1 1 0	1	0 0	1 0	0 0 0	0	0 0	0
15	0 0	1 1 1 1	0	0 0	0 0	0 0 1	0	0 0	0
17	0 0	1 1 1 1	0	0 0	0 0	0 0 1	0	0 0	0
6	0 0	0 1 0 0	1	0 0	0 0	0 1 0	0	0 0	0
1	0 0	0 0 0 0	0	1 0	0 0	0 0 1	0	0 0	0
3	0 0	0 0 0 1	0	1 1	0 0	0 0 0	0	0 0	0
24	0 0	0 0 1 1	0	1 1	0 0	0 0 0	0	0 0	0
19	0 0	0 0 0 0	0	0 0	1 1	0 1 0	0	0 0	0
22	0 0	0 1 0 0	0	0 0	0 1	0 1 0	0	0 0	0
7	0 0	0 0 0 0	0	0 0	0 0	1 1 1	0	0 0	0
13	1 0	0 0 0 0	0	0 0	0 0	0 1 1	0	0 0	0
14	0 0	0 0 0 0	0	0 0	0 1	1 1 0	0	0 0	0
18	0 0	0 0 0 0	0	0 0	0 0	1 1 1	1	0 0	0
21	0 0	0 0 0 0	0	0 0	0 0	1 1 0	1	0 0	0
20	0 0	0 0 0 0	0	1 0	0 0	0 0 0	1	0 0	0
4	0 0	0 0 1 0	0	1 0	0 0	0 0 0	0	1 1	0
11	0 0	0 0 1 0	0	0 0	0 0	0 0 0	0	0 0	1
16	0 0	0 0 1 0	0	0 0	0 0	0 0 0	0	0 0	1

Carrie1973

	1 1 1 3 5 6 2 8	1 1 1 2 1 6 9	1 2 4 8 4 0	1 7 0	1 9 5	1 3	1 7
1	1 1 1 1 1 1	0 0 0 1	1 1 1 0	0 0	0 1	1	0
2	0 1 0 1 1 0	0 0 0 0	0 0 0 0	0 1	1 0	0	0
10	1 0 1 1 1 1	0 0 0 0	0 0 0 0	0 0	1 0	1	0
11	1 1 1 0 1 0	0 0 0 0	0 0 0 0	0 0	0 0	0	0
15	1 1 1 1 1 1	1 0 0 0	0 1 1 1	0 0	0 0	1	1
5	0 0 0 0 0 0	1 1 1 1	0 0 0 0	0 0	1 0	0	0
14	0 0 0 0 0 0	1 0 1 0	0 0 0 0	0 0	0 0	0	0
3	1 0 0 0 0 1	0 0 0 1	1 1 0 1	0 0	0 0	0	0
12	0 0 0 0 0 0	0 0 0 0	1 1 1 1	0 0	0 1	0	0
16	0 0 0 0 0 0	0 0 0 0	0 1 1 1	0 1	0 0	0	0
22	1 1 0 1 0 0	0 0 0 0	1 1 1 1	0 1	1 1	0	0
7	0 1 0 0 0 0	0 1 1 0	0 0 0 0	1 1	1 0	0	1
8	0 0 0 0 0 0	0 0 0 0	0 0 0 0	1 1	0 0	0	0
9	0 0 0 0 0 0	0 0 0 0	0 0 0 0	1 1	0 0	0	0
17	0 0 0 0 0 0	0 0 0 0	1 0 0 0	0 1	0 0	0	0
18	0 1 0 0 1 0	0 0 1 0	0 0 0 0	0 0	1 1	1	1
19	0 0 0 0 0 0	0 0 0 0	0 0 0 0	0 0	1 1	0	1
21	1 0 0 0 0 0	0 0 0 0	1 0 0 0	0 0	1 1	0	0
23	1 0 0 0 0 0	0 0 0 0	1 0 0 0	0 0	1 1	0	0
4	0 0 1 0 0 0	0 0 0 0	0 0 0 0	0 0	0 0	1	0
20	0 0 0 0 1 0	0 0 1 0	0 0 0 0	0 0	0 0	1	0
6	0 0 0 0 0 0	0 0 0 0	0 0 0 0	0 1	0 0	0	1
13	0 0 0 0 0 0	1 0 0 0	0 1 0 0	0 0	0 0	0	1

Kumar1986

	1 1 6	1 1 1 2 4 3 4 8	1 3 7 8 7	2 0 5 6 9 0	1 1 2 5 9
18	1 1	1 0 0 1 0	0 0 0 0	0 0 0 0	0 0 0 0
25	1 0	0 0 0 0 0	0 0 0 1	0 0 1 0 0	0 0 0 0
30	1 1	0 0 0 0 0	0 1 0 0	0 0 1 0 0	0 0 1 1
32	1 1	0 0 0 0 0	0 1 0 0	0 0 0 1 0	0 0 0 1
34	1 0	0 0 0 0 0	0 0 0 0	0 0 0 0 0	0 0 0 0
2	0 0	1 1 1 1 1	0 0 0 0	0 0 0 0 0	0 0 0 0
7	0 0	0 1 0 1 1	0 1 0 0	0 0 0 0 0	0 0 0 0
10	0 0	1 0 0 1 0	0 0 0 0	0 0 0 0 0	0 0 0 0
12	0 1	1 1 1 1 1	0 1 0 0	0 0 0 0 0	0 0 0 0
13	0 0	1 1 1 1 1	0 0 0 0	0 0 0 0 0	0 0 0 0
24	1 0	1 1 1 1 1	0 1 0 0	0 0 0 0 0	0 0 0 0
27	0 0	0 1 0 1 0	0 0 0 0	0 0 0 0 0	0 0 0 0
31	0 0	1 0 0 1 1	1 1 0 0	0 0 0 0 0	0 0 0 0
1	1 0	0 0 0 0 0	1 1 1 1	0 0 0 0 0	0 0 0 0
3	0 0	0 0 0 0 0	1 1 0 1	0 0 0 0 0	0 0 0 0
5	0 0	0 0 0 0 0	1 1 1 1	0 1 0 0 0	0 0 0 0
15	1 0	0 0 0 0 0	1 1 1 1	0 0 0 0 0	0 0 0 0
17	0 0	0 0 0 0 0	0 1 1 1	0 1 0 0 0	0 0 0 0
20	1 0	0 0 0 0 1	0 1 1 1	0 0 0 0 0	0 0 0 0
29	0 0	0 0 0 0 0	1 1 0 0	0 0 0 0 0	0 0 0 0
35	1 0	0 0 0 0 0	0 0 1 1	0 0 0 1 0	0 0 0 0
8	0 0	0 0 0 0 0	0 0 0 0	1 0 1 1 1	0 0 0 0
14	1 0	0 0 0 0 0	0 0 0 0	1 1 1 1 1	0 0 0 0
19	0 0	1 0 0 0 0	0 1 0 0	1 0 1 1 1	0 0 0 0
22	0 0	0 0 0 0 0	0 1 0 0	0 0 1 0 1	0 0 0 0
23	1 0	0 0 0 0 0	0 1 1 0	1 1 0 1 0	0 0 0 0
26	0 1	0 0 0 0 0	0 1 0 0	1 1 0 0 1	0 0 0 0
4	0 1	0 0 0 0 0	0 0 0 0	0 0 0 0 0	1 1 1 1
6	1 1	0 0 0 0 0	0 0 0 0	0 0 0 0 0	1 1 1 0
9	0 1	0 0 0 0 0	0 0 1 0	0 0 0 0 0	1 1 1 1
11	1 0	0 0 0 0 0	0 0 0 0	0 0 0 0 0	1 1 1 1
21	0 0	0 0 0 0 0	0 0 0 0	0 0 0 0 0	1 1 1 1
28	0 0	0 0 0 0 0	0 0 0 0	0 0 0 0 0	0 0 1 1
33	0 0	0 0 0 0 0	0 0 0 0	0 0 0 0 0	1 1 0 0
16	0 0	0 0 0 0 0	0 0 0 0	0 0 0 0 1	0 0 0 0

Boe1991

	1 1 3 7 8 7	1 1 1 2 4 3 4 8	5	1 2 6 9 0 0	1 1 1 1 1 1 2 5 6 9
1	1 1 1 1 1	0 0 0 0 0	0	0 0 0 0	0 0 0 0 0
3	1 1 1 1 1	0 0 0 0 0	0	0 0 0 0	0 0 0 0 0
15	0 1 1 1 1	0 0 0 0 0	0	0 0 0 0	0 0 0 0 0
17	0 1 1 1 1	0 0 0 0 0	0	0 0 0 0	0 0 0 0 0
20	1 0 1 1 0	0 0 0 0 0	0	0 0 0 0	0 0 0 0 0
23	1 0 1 1 1	0 0 0 0 0	0	0 0 0 0	0 0 0 0 0
25	1 0 0 1 1	0 0 0 0 0	0	0 0 0 0	0 0 0 0 0
29	1 1 0 1 1	0 0 0 0 0	0	0 0 0 0	0 0 0 0 0
5	0 1 1 1 1	0 0 0 0 0	0	0 0 0 0	0 0 0 0 0
2	0 0 0 0 0	1 1 1 1 1	0	0 0 0 0	0 0 0 0 0
7	0 0 0 0 0	1 1 0 1 0	0	0 0 0 0	0 0 0 0 0
10	0 0 0 0 0	1 0 0 1 1	0	0 0 0 0	0 0 0 0 0
12	0 0 0 0 0	1 1 1 1 1	0	0 0 0 0	0 0 0 0 0
13	0 0 0 0 0	1 1 1 1 1	0	0 0 0 0	0 0 0 0 0
18	0 0 0 0 0	1 0 0 1 1	0	0 0 0 0	0 0 0 0 0
24	0 0 0 0 0	1 1 1 1 1	0	0 0 0 0	0 0 0 0 0
27	0 0 0 0 0	1 1 0 1 0	0	0 0 0 0	0 0 0 0 0
31	0 1 0 0 0	1 0 0 1 1	0	0 0 0 0	0 0 0 0 0
16	0 0 0 0 0	0 0 0 0 0	1	1 0 1 0	0 0 0 0 0
34	0 0 0 0 0	0 0 0 0 0	1	1 0 0 0	0 0 0 0 0
8	0 0 0 0 0	0 0 0 0 0	1	1 1 1 1	0 0 0 0 0
14	0 0 0 0 0	0 0 0 0 0	1	1 1 1 1	0 0 0 0 0
19	0 0 0 0 0	0 0 0 0 0	0	1 1 1 1	0 0 0 0 0
22	0 0 0 0 0	0 0 0 0 0	0	1 1 1 0	0 0 0 0 0
26	0 0 0 0 0	0 0 0 0 0	0	1 1 1 1	0 0 0 0 0
4	0 0 0 0 0	0 0 0 0 0	0	0 0 0 0	1 1 1 1 1
6	0 0 0 0 0	0 0 0 0 0	0	0 0 0 0	1 1 1 1 0
9	0 0 0 0 0	0 0 0 0 0	0	0 0 0 0	1 1 1 1 1
11	0 0 0 0 0	0 0 0 0 0	0	0 0 0 0	1 1 1 1 1
21	0 0 0 0 0	0 0 0 0 0	0	0 0 0 0	1 1 1 1 1
28	0 0 0 0 0	0 0 0 0 0	0	0 0 0 0	1 0 1 1 1
30	0 0 0 0 0	0 0 0 0 0	0	0 0 0 0	1 0 1 1 1
32	0 0 0 0 0	0 0 0 0 0	0	0 0 0 0	1 0 0 1 1
33	0 0 0 0 0	0 0 0 0 0	0	0 0 0 0	0 1 0 0 0
35	0 0 0 0 0	0 0 0 0 0	0	0 0 0 0	1 0 0 0 0

Carrie1973

	1 2 2 1 3 1 2	1 1 2 5 1 9	2 3 0	1 4 6	1 1 1 6 8 2 5 8	1 1 9 0 7	1 2 2 7 4 3 4
1	1 1 1 1	0 0 0 0	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
9	1 1 1 1	0 0 0 0	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
16	1 1 1 1	0 0 0 0	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
17	1 1 1 1	0 0 0 0	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
33	1 1 1 1	0 0 0 0	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
10	0 0 0 0	1 1 1 1	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
13	0 0 0 0	1 1 1 1	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
14	0 0 0 0	1 1 1 1	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
22	0 0 0 0	1 1 1 1	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
35	0 0 0 0	1 1 1 1	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
36	0 0 0 0	1 1 1 1	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
2	0 0 0 0	0 0 0 0	1 1	0 0	0 0 0 0 0	0 0 0	0 0 0 0
11	0 0 0 0	0 0 0 0	1 1	0 0	0 0 0 0 0	0 0 0	0 0 0 0
12	0 0 0 0	0 0 0 0	1 1	0 0	0 0 0 0 0	0 0 0	0 0 0 0
15	0 0 0 0	0 0 0 0	1 1	0 0	0 0 0 0 0	0 0 0	0 0 0 0
23	0 0 0 0	0 0 0 0	1 1	0 0	0 0 0 0 0	0 0 0	0 0 0 0
24	0 0 0 0	0 0 0 0	1 1	0 0	0 0 0 0 0	0 0 0	0 0 0 0
31	0 0 0 0	0 0 0 0	1 1	0 0	0 0 0 0 0	0 0 0	0 0 0 0
34	0 0 0 0	0 0 0 0	1 1	0 0	0 0 0 0 0	0 0 0	0 0 0 0
8	0 0 0 0	0 0 0 0	0 0	1 1	0 0 0 0 0	0 0 0	0 0 0 0
19	0 0 0 0	0 0 0 0	0 0	1 1	0 0 0 0 0	0 0 0	0 0 0 0
21	0 0 0 0	0 0 0 0	0 0	1 1	0 0 0 0 0	0 0 0	0 0 0 0
28	0 0 0 0	0 0 0 0	0 0	1 1	0 0 0 0 0	0 0 0	0 0 0 0
37	0 0 0 0	0 0 0 0	0 0	1 1	0 0 0 0 0	0 0 0	0 0 0 0
38	0 0 0 0	0 0 0 0	0 0	1 1	0 0 0 0 0	0 0 0	0 0 0 0
39	0 0 0 0	0 0 0 0	0 0	1 1	0 0 0 0 0	0 0 0	0 0 0 0
4	0 0 0 0	0 0 0 0	0 0	0 0	1 1 1 1 1	0 0 0	0 0 0 0
5	0 0 0 0	0 0 0 0	0 0	0 0	1 1 1 1 1	0 0 0	0 0 0 0
18	0 0 0 0	0 0 0 0	0 0	0 0	1 1 1 1 1	0 0 0	0 0 0 0
26	0 0 0 0	0 0 0 0	0 0	0 0	1 1 1 1 1	0 0 0	0 0 0 0
27	0 0 0 0	0 0 0 0	0 0	0 0	1 1 1 1 1	0 0 0	0 0 0 0
30	0 0 0 0	0 0 0 0	0 0	0 0	1 1 1 1 1	0 0 0	0 0 0 0
6	0 0 0 0	0 0 0 0	0 0	0 0	0 0 0 0 0	1 1 1	0 0 0 0
7	0 0 0 0	0 0 0 0	0 0	0 0	0 0 0 0 0	1 1 1	0 0 0 0
20	0 0 0 0	0 0 0 0	0 0	0 0	0 0 0 0 0	1 1 1	0 0 0 0
29	0 0 0 0	0 0 0 0	0 0	0 0	0 0 0 0 0	1 1 1	0 0 0 0
40	0 0 0 0	0 0 0 0	0 0	0 0	0 0 0 0 0	1 1 1	0 0 0 0
3	0 0 0 0	0 0 0 0	0 0	0 0	0 0 0 0 0	0 0 0	1 1 1 1
25	0 0 0 0	0 0 0 0	0 0	0 0	0 0 0 0 0	0 0 0	1 1 1 1
32	0 0 0 0	0 0 0 0	0 0	0 0	0 0 0 0 0	0 0 0	1 1 1 1

	1 2 1 2 1 1	1 2 2 3 2 2 3	4	1 5 5	1 6 6	1 1 2 7 7 8 6	8	9	1 0	1 2 3 4	1 4	1 2 2 3 9 0 9 0	2 5	2 2 7 8
10	1 0 1 1	0 0 1 0	0	0 0	0 0	0 0 0 0	0	0	0	0 0	0	0 0 0 0	0	0 0
32	1 1 0 1	1 0 1 0	0	0 0	0 0	0 0 0 0	0	0	0	0 0	0	0 0 0 0	0	0 0
33	1 1 1 1	0 0 0 0	0	0 0	0 0	0 0 0 0	0	0	0	0 0	0	0 0 0 0	0	0 0
41	1 1 1 0	0 0 0 0	0	0 0	0 0	0 0 0 0	0	0	0	0 0	0	0 0 0 0	0	0 0
12	0 1 0 1	1 1 1 1	0	0 0	0 0	0 0 0 0	0	0	1	0 0	0	0 0 0 0	0	0 0
23	0 0 0 0	1 1 1 1	0	0 0	0 0	0 0 0 0	0	0	1	0 0	0	0 0 0 0	0	0 0
31	1 0 0 1	1 0 1 1	0	0 0	0 0	0 0 0 0	0	0	0	0 0	0	0 0 0 0	0	0 0
39	1 1 1 0	1 1 1 1	0	0 0	0 0	0 0 0 0	0	0	1	0 0	0	0 0 0 0	0	0 0
40	0 0 0 1	1 1 1 0	0	0 0	0 0	0 0 0 0	0	0	0	0 0	0	0 0 0 0	0	0 0
7	0 0 0 0	0 0 0 0	1	0 0	0 0	0 1 0 0	0	0	0	0 0	0	0 0 0 0	0	0 0
15	0 0 0 0	0 0 0 0	1	0 0	0 0	0 0 0 0	0	0	0	0 0	0	0 0 0 0	0	1 0
24	0 0 0 0	0 1 0 0	1	0 0	0 0	0 0 0 0	0	0	0	0 0	0	0 0 0 0	0	0 0
25	0 0 0 0	0 0 0 0	1	0 0	0 0	0 0 0 0	0	0	0	1 0	0	0 0 0 0	0	0 0
28	0 0 0 0	0 0 0 0	1	0 0	0 1	0 0 0 0	0	0	0	0 0	0	0 0 0 0	0	0 0
35	0 0 0 0	0 0 0 0	1	0 0	0 0	0 0 0 0	0	0	0	0 0	0	0 0 0 0	0	0 1
17	0 0 0 0	0 0 0 0	0	1 1	0 0	0 0 0 0	0	0	0	0 0	0	0 0 0 0	0	0 0
6	0 0 0 0	0 0 0 0	0	0 0	1 1	0 0 0 0	0	0	0	0 0	0	0 0 0 0	0	0 0
16	0 0 0 0	0 0 0 0	0	0 0	0 0	1 1 1 1	0	0	0	0 0	0	0 0 0 0	0	0 0
27	0 0 0 0	0 0 0 0	0	0 0	0 0	1 1 1 1	0	0	0	0 0	0	0 0 0 0	0	0 0
34	0 0 0 0	0 0 0 0	0	1 0	0 0	1 0 1 1	0	0	0	0 0	0	0 0 0 0	0	0 0
36	0 0 0 0	0 0 0 0	0	1 0	0 0	1 1 0 1	0	0	0	0 0	0	0 0 0 0	0	0 0
9	0 0 0 0	0 0 0 0	0	0 0	0 0	0 0 0 0	1	0	0	0 0	0	0 0 1 0	0	0 0
14	0 0 0 0	0 0 0 0	0	0 0	0 0	0 0 1 0	1	0	0	0 0	0	0 0 0 0	0	0 0
29	0 0 0 0	0 0 0 0	0	0 0	0 0	0 0 0 0	1	0	0	0 0	0	0 0 0 0	0	0 1
13	0 0 0 0	0 0 0 0	0	0 0	0 0	0 0 0 0	0	1	0	0 0	0	0 0 1 0	0	0 0
2	0 0 0 0	0 0 0 1	0	0 0	0 0	0 0 0 0	0	0	1	0 0	0	0 0 0 0	0	0 0
20	0 0 0 0	0 0 0 0	0	0 0	0 0	0 0 0 0	0	0	1	0 0	0	0 0 0 0	0	0 0
38	0 0 0 0	0 0 0 0	0	0 0	0 0	0 0 0 0	0	0	0	1 1	0	0 0 0 0	0	0 0
5	0 0 0 0	0 0 0 0	0	1 0	0 0	0 0 0 0	0	0	0	0 0	1	0 0 0 0	0	0 0
18	0 1 0 0	0 1 0 0	0	0 0	0 0	0 0 0 0	0	0	0	1 0	1	0 0 0 0	0	0 0
26	0 0 0 0	0 0 0 0	0	0 0	1 0	0 0 0 0	0	0	0	0 0	1	0 0 0 0	0	0 0
37	0 0 0 0	0 0 0 0	0	0 1	0 0	0 0 0 0	0	0	0	0 0	1	0 0 0 0	0	0 0
1	0 0 0 0	0 0 0 0	0	0 0	0 0	0 0 0 0	1	0	0	0 0	0	1 1 1 1	0	0 0
3	0 0 0 0	0 0 0 0	0	0 0	0 0	0 0 0 0	0	1	0	0 0	0	1 1 1 1	0	0 0
21	0 0 0 0	0 0 0 0	0	0 0	0 0	0 0 0 0	0	1	0	0 0	0	1 1 1 1	0	0 0
22	0 0 0 0	0 0 0 0	0	0 0	0 0	0 0 0 0	0	0	0	0 0	0	1 1 0 1	0	0 0
30	0 0 0 0	0 0 0 0	0	0 0	0 0	0 0 0 0	0	0	0	0 0	0	1 0 1 0	0	0 0
4	0 0 0 0	0 0 0 0	0	0 0	0 0	0 0 0 0	0	0	0	0 0	1	0 0 0 0	1	0 0
8	0 0 0 0	0 0 0 0	0	0 0	0 0	0 0 0 0	1	0	0	0 0	0	0 0 0 0	0	1 1
11	0 1 0 0	0 1 0 0	0	0 0	0 0	0 0 0 0	0	0	0	0 0	0	0 0 0 0	0	0 0
19	0 0 0 0	0 1 0 0	0	0 0	0 0	0 0 0 0	0	0	0	0 0	0	0 0 0 0	0	0 0

	1 1 4 1	1 2 5 9	1 3 0	1 6 8 2	1 7 3	1 1 4 6	1 1 5 7	1 2 2 8 0 2	1 2 2 9 1 3	2 3 4 0	2 2 2 5 7 9	2 6	2 8
1	1 1 1	0 1 1	0 0	0 0 0	0 1	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
7	0 1 1	0 1 0	0 1	0 1 1	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
18	1 1 1	0 0 0	0 1	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
2	0 0 0	1 1 1	1 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
3	0 0 0	1 1 1	0 0	0 0 1	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
17	0 0 0	1 0 1	0 0	0 0 0	0 1	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
4	0 0 1	0 0 0	1 1	0 0 0	1 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
6	1 0 0	0 0 1	1 1	0 0 0	0 1	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
8	1 0 0	0 0 0	1 0	0 0 1	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
10	0 0 0	0 1 0	0 1	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
11	0 0 0	0 1 0	1 1	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
5	1 0 0	0 0 0	0 0	1 1 0	0 1	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
13	0 0 0	0 0 0	0 0	1 1 0	1 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
14	0 0 0	0 0 0	0 0	1 1 1	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
15	0 0 0	0 0 0	0 0	1 1 1	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
16	0 0 0	0 0 0	0 0	1 1 1	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
9	0 0 0	1 0 0	0 1	1 0 0	1 1	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
12	0 0 0	0 0 0	0 0	0 1 0	1 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
19	0 0 0	0 0 0	0 0	0 0 0	0 0	1 1	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
21	0 0 0	0 0 0	0 0	0 0 0	0 0	1 1	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
22	0 0 0	0 0 0	0 0	0 0 0	0 0	0 1	1 0	0 0 0	0 0 0	0 0	0 0 0	0	0
23	0 0 0	0 0 0	0 0	0 0 0	0 0	1 1	0 1	0 0 0	0 0 0	0 0	0 0 0	0	0
26	0 0 0	0 0 0	0 0	0 0 0	0 0	1 1	0 0	0 0 0	0 0 0	0 0	0 0 0	0	0
20	0 0 0	0 0 0	0 0	0 0 0	0 0	1 0	1 1	0 0 0	0 0 0	0 0	0 0 0	0	0
24	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	1 1	0 0 0	0 0 0	0 0	0 0 0	0	0
25	0 0 0	0 0 0	0 0	0 0 0	0 0	1 0	1 1	0 0 0	0 0 0	0 0	0 0 0	0	0
27	0 0 0	0 0 0	0 0	0 0 0	0 0	0 1	1 0	0 0 0	0 0 0	0 0	0 0 0	0	0
28	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	1 1 0	1 0 0	0 0	0 0 0	0	0
30	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	1 0 1	0 1 0	0 0	0 0 0	0	0
32	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	0 1 1	0 1 0	0 0	0 0 0	0	0
34	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	1 0 1	1 0 0	0 0	0 0 0	0	0
36	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	1 1 0	0 1 0	0 0	0 0 0	0	0
35	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	0 1 1	0 0 1	0 0	0 0 0	0	0
37	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	1 1 1	0 0 0	0 0	0 0 0	0	0
29	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	0 1 0	1 0 1	0 0	0 0 0	0	0
31	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	1 1 1	0 0	0 0 0	0	0
33	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	1 1 1	0 0	0 0 0	0	0
38	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	1 1 1	0 0	0 0 0	0	0
39	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	1 1	0 0 1	0	1
42	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	1 1	1 0 0	0	0
44	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	1 1	0 1 0	0	0
48	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	1 1	0 0 1	0	0
40	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 1	0 1 1	0	0
41	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	1 1 1	0	0
45	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	1 0 1	0	1
49	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	1 1 0	1	0
50	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	1 0 1	0	0
43	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	1	1
46	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 1	0 0 0	1	0
47	0 0 0	0 0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 1 0	0	1

	3	1 1 2 3	2	2 2 3 3	1 2 3 4	1 1	1 1 3	1 1 3 3 3	1 2 2 3	2 2 2
	1 3 7 2	2 0 6 1 1	4 9 0	5 8 2 3 7 9	6 2 6 8 0	1 3	4 7 5	5 8 3 4 6	9 5 8 0	4 7 9
10	1 0 1 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
18	1 1 1 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
29	1 1 0 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
33	1 1 1 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
34	1 1 1 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
37	1 1 1 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
41	0 1 1 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 1 0
44	1 1 1 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
49	1 1 1 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
50	1 1 0 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 1 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
54	1 1 1 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
55	1 1 1 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
35	0 0 0 0	1 1 1 1 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
47	0 0 0 0	1 1 1 0 1	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
53	0 0 0 0	1 1 0 1 1	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
77	0 0 0 0	0 1 1 1 1	0 0 0	0 0 0 1 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
78	0 0 0 0	1 1 1 1 0	0 0 0	0 0 0 0 0 0	0 0 0 0 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
79	0 0 0 0	1 1 1 1 1	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
83	0 0 0 0	1 1 1 1 1	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 1	0 0 0
88	0 0 0 0	1 1 1 1 1	0 0 0	0 0 0 0 0 0	0 0 1 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
93	0 0 0 0	1 1 1 1 1	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 1	0 0 0 0	0 0 0
8	0 0 0 0	0 0 0 0 0	1 1 1	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
11	0 0 0 0	0 0 0 0 0	1 1 1	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
13	0 0 0 0	0 0 0 0 0	1 1 1	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
14	0 0 0 0	0 0 0 0 0	1 1 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 1	0 0 0
20	0 0 0 0	0 0 0 0 0	1 1 1	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
22	0 0 0 0	0 0 0 0 0	1 1 1	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
23	0 0 0 0	0 0 0 0 0	1 1 1	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
32	0 0 0 0	0 0 0 0 0	1 1 1	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
63	0 0 0 0	0 0 0 0 0	0 0 0	1 1 0 0 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
66	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 0 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
68	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
69	0 0 0 0	0 0 0 0 0	0 0 0	1 0 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
82	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
84	0 0 0 0	0 0 0 0 0	0 1 0	1 1 1 1 1 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
85	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
89	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
90	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
92	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
94	0 0 0 0	0 0 0 0 0	0 0 0	1 0 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 1	0 0 0 0	0 0 0
95	0 0 0 0	0 0 0 0 0	0 0 0	1 1 0 1 1 1	0 0 0 0 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
96	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
97	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 0	0 1 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
98	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
99	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	1 0 0 0	0 0 0
100	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 1 0 0 0	0 0 0 0	0 0 0
36	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 1 1 1 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
38	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 1 1 1 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
42	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 0 1 1 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
51	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 1 1 1 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
52	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 1 1 1 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
64	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 1 1 1 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
65	0 0 0 0	0 0 1 0 0	0 0 0	0 0 0 0 0 0	1 0 1 1 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
70	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 1 1 1 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
72	0 0 0 0	0 0 0 1 0	0 0 0	0 0 0 0 0 0	1 1 1 0 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
74	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 1 1 1 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
75	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 1 1 0 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
76	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 1 1 1 1	0 0	0 0 0	0 0 0 0 1	0 0 0 0	0 0 0
80	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 1 1 1 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
81	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 1 0 1 1	0 0	0 1 0	0 0 0 0 0	0 0 1 0	0 0 0
86	0 0 0 0	0 0 1 0 0	0 0 0	0 1 0 0 0 0	1 1 1 1 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
87	0 0 0 0	0 0 0 0 0	0 0 0	1 0 0 0 0 0	1 1 1 1 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
1	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
2	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
3	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
4	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
5	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
7	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
21	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	0 1 0	0 0 0 0 0	0 0 0 0	0 0 0
25	0 0 1 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
28	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	1 0 0	0 0 0 0 0	0 0 0 0	0 0 0
48	0 1 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	0 0 0	1 0 0 0 0	0 0 0 0	0 0 0
6	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	1 0 1	0 0 0 0 0	0 0 0 0	0 0 0
15	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	1 1 1	0 0 0 0 0	0 0 0 0	0 0 0
16	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	1 1 1	0 0 0 0 0	0 0 0 0	0 0 0
24	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	1 1 1	0 0 0 0 0	0 0 0 0	0 0 0
27	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 0	1 0 1	0 0 0 0 0	0 0 0 0	0 0 0
60	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 1	1 1 1	0 0 0 0 0	0 0 0 0	0 0 0
39	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	1 0 0	0 1 0 1 1	0 0 0 0	0 0 0
45	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	1 0 1 1 1	0 0 0 0	0 0 0
67	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	1 1 1 1 1	0 0 0 0	0 0 0
71	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	1 1 1 1 0	0 1 0 0	0 0 0
73	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 1 1 1 1	1 0 0 0	0 0 0
91	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	1 1 1 1 1	0 0 0 1	0 0 0
56	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	1 1 1 1	0 0 0
57	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	1 1 1 1	0 0 0
58	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	1 1 1 1	0 0 0
59	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	1 1 1 1	0 0 0
61	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	1 1 1 1	0 0 0
62	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	1 1 1 1	0 0 0
9	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	1 1 1
12	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	1 1 1
17	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0					

C.2 Soluções com Restrição *Singleton*

	1 4	2 3 5
2	1 1	0 0 0
4	1 1	0 0 0
5	1 0	0 0 1
6	1 1	0 1 0
1	0 0	1 1 1
3	0 0	1 1 0
7	0 0	0 1 1

King1982

	1 4	2 3 5
1	1 1	0 0 0
7	1 0	0 0 0
2	0 1	1 0 1
3	0 1	1 1 0
4	0 1	1 1 1
5	1 0	1 1 1
6	1 0	0 1 1

Waghodekar1984

	1 4	2 3 5
1	1 1	1 0 0
2	1 1	0 0 0
3	1 1	1 0 0
5	1 1	0 0 0
6	1 1	1 0 0
8	1 1	1 0 0
11	1 1	1 0 0
12	1 1	1 0 0
13	1 1	1 0 0
14	1 1	0 0 0
16	1 1	0 0 0
17	1 1	0 0 0
4	0 0	1 1 1
7	0 0	1 1 0
9	0 0	0 0 1
10	0 0	1 1 1
15	0 0	1 1 1
18	0 0	1 1 1

Seifoddini1989

	1 4 6	2 3 5
2	1 0 0	1 0 0
4	1 1 1	0 0 0
7	1 1 1	1 0 0
1	0 0 0	1 0 1
3	0 0 0	1 1 1
5	0 0 0	1 0 1
6	0 0 0	1 1 1
8	0 0 0	1 1 1

Kusiak1992

	1 2	3 4 5	6 7
1	1 0	1 0 0	0 0
2	1 1	0 0 0	0 0
6	1 1	0 0 0	0 0
9	0 1	0 0 0	0 0
3	0 0	1 1 1	0 0
7	0 0	1 1 0	0 0
11	0 0	1 0 1	0 0
4	0 0	0 0 1	1 0
5	0 0	0 0 0	1 1
8	0 0	0 0 0	0 1
10	0 0	0 0 0	1 1

Boctor1991

	1 4 7	2 5	3 6
2	1 0 0	0 0	0 0
3	1 1 1	0 0	0 0
6	0 1 1	0 0	0 0
7	1 0 1	0 0	0 0
1	0 1 0	1 0	0 1
5	0 0 0	1 1	0 0
8	0 0 0	0 1	0 1
11	0 0 0	1 0	1 0
4	0 0 1	0 0	0 1
9	0 0 1	0 0	0 1
10	0 0 0	0 0	1 1

Kusiak1987

	1 2 3	4 5 6	7 8
1	1 1 0	0 0 0	0 0
2	1 0 0	0 0 0	0 0
3	1 1 1	0 0 0	0 0
4	1 1 1	0 0 0	0 0
5	0 1 1	0 0 0	0 0
6	0 1 1	1 0 0	0 0
7	0 1 1	1 1 1	0 0
8	0 0 1	1 1 1	0 0
9	0 0 1	1 1 1	0 0
10	0 1 0	1 1 0	0 0
11	0 0 0	0 0 1	1 1
12	0 0 0	0 0 0	1 1

Seifoddini1986

	5 6	2 4 7 8	1 3
1	1 1	0 0 0 0	0 0
5	1 1	0 0 0 0	0 0
10	1 1	0 1 0 0	0 0
12	1 1	0 0 1 0	0 0
15	1 1	0 0 0 0	0 0
3	0 0	1 1 1 1	1 0
4	0 0	1 1 1 1	0 0
6	1 0	1 1 1 1	0 0
7	0 0	1 1 1 1	0 0
18	0 0	1 1 1 1	0 0
20	0 1	1 1 1 1	0 0
2	0 0	0 0 0 0	1 1
8	0 0	0 0 0 0	1 1
9	0 1	0 0 0 0	1 1
11	0 0	0 0 1 0	1 1
13	0 0	0 0 0 0	1 1
14	0 0	1 0 0 0	1 1
16	0 0	0 0 0 0	1 1
17	1 0	0 0 0 0	1 1
19	0 0	0 0 0 0	1 1

Chandrasekharan1986a

	1 2 4 6 8	3 5 7
1	1 0 0 1 1	0 1 0
2	0 1 0 1 1	0 1 0
3	1 1 1 0 1	0 1 0
4	1 1 1 0 1	0 0 0
7	0 1 1 1 0	1 0 1
8	0 0 1 1 1	1 0 1
9	1 1 1 0 1	0 0 0
10	1 0 1 1 1	0 1 0
14	1 0 1 1 1	0 0 0
15	1 0 1 1 0	0 1 0
18	1 1 1 1 0	0 0 0
5	0 0 0 1 1	1 0 1
6	0 1 0 0 0	1 1 1
11	0 1 0 1 0	1 0 1
12	0 0 0 1 0	1 1 1
13	0 0 1 0 1	1 1 1
16	1 0 0 1 1	1 1 1
17	0 0 1 0 1	1 1 1
19	1 0 1 1 0	1 1 1
20	0 1 1 0 0	1 1 1

Chandrasekharan1986b

	1 4 5 6	2 7 9 0	3 8
1	1 1 0 1	0 0 0 0	0 0
7	0 0 1 1	0 0 0 0	0 0
10	1 0 1 1	0 0 0 0	0 0
2	0 0 0 0	0 0 1 1	0 0
3	0 0 0 0	1 1 1 1	0 0
4	0 0 0 0	1 0 1 1	0 0
8	0 0 0 0	1 1 1 0	0 0
5	0 0 0 0	0 0 0 0	1 0
6	0 0 0 0	0 0 0 0	1 1
9	0 0 0 0	0 0 0 0	0 1

Mosier1985a

	1 7 0	3 4 6 9	2 5 8
2	1 1 1	0 0 0 0	0 0 0
10	1 1 1	0 0 0 0	0 0 0
7	0 1 1	0 0 0 0	0 0 0
11	1 1 1	0 0 0 0	0 0 0
12	1 1 1	0 0 0 0	0 0 0
1	0 0 0	1 1 1 0	0 0 0
4	0 0 0	0 1 1 1	0 0 0
6	0 0 0	1 0 1 1	0 0 0
9	0 0 0	1 1 1 1	0 0 0
14	0 0 0	1 1 1 1	0 0 0
3	0 0 0	0 0 0 0	1 1 1
5	0 0 0	0 0 0 0	1 1 1
8	0 0 0	0 0 0 0	1 1 1
13	0 0 0	0 0 0 0	1 1 1
15	0 0 0	0 0 0 0	1 1 1

Chan1982

	1 1 1 2 3	1 2 3 1	4 5 7	6 8 9	1 1 0 4
7	1 0 1	0 0 0	0 0 0	0 0 0	0 0
8	1 1 1	0 0 0	0 0 1	0 0 0	0 0
9	0 1 1	0 0 0	0 0 0	0 0 0	0 0
18	0 0 1	0 0 0	0 0 0	0 0 0	0 0
4	0 0 0	1 1 1	0 0 0	0 0 0	1 0
5	0 0 0	1 1 1	0 0 0	0 0 0	0 0
21	0 0 0	0 1 1	0 0 0	0 0 0	0 0
2	0 0 0	0 0 0	1 1 1	0 0 0	0 0
3	0 0 0	0 0 0	1 1 1	0 0 0	0 0
17	0 0 0	0 0 0	1 1 1	0 0 0	0 0
19	0 0 0	0 0 0	1 0 0	0 0 0	0 0
20	0 0 0	0 0 0	1 1 1	0 0 0	0 0
22	0 0 1	0 0 0	1 1 0	0 0 0	0 0
1	0 0 0	0 0 0	0 0 0	1 1 0	0 0
6	0 0 0	0 0 0	0 0 0	0 1 1	0 0
10	0 0 0	0 0 0	0 0 0	1 1 1	0 1
12	0 0 0	0 0 0	0 0 0	1 1 1	0 0
14	0 0 0	0 0 0	0 0 0	1 1 0	0 0
15	0 0 0	0 0 0	0 0 0	1 1 1	0 1
16	0 0 0	0 0 0	0 0 0	1 1 0	0 0
11	0 0 0	0 0 0	0 0 0	1 0 0	0 1
13	0 0 0	0 0 0	0 0 0	0 0 1	0 1
23	0 0 0	0 0 0	0 0 0	0 0 0	1 0

Askin1987

	1 1 1 2 3	1 1 2 3 0 1	4 5 7	6 8	1 9 4
6	1 0 1	0 0 0 0	0 0 0	0 0	0 0
7	1 1 1	0 0 0 0	0 0 1	0 0	0 0
8	0 1 1	0 0 0 0	0 0 0	0 0	0 0
18	0 0 1	0 0 0 0	0 0 0	0 0	0 0
3	0 0 0	1 1 1 1	0 0 0	0 0	0 0
4	0 0 0	1 1 0 1	0 0 0	0 0	0 0
21	0 0 0	0 1 0 1	0 0 0	0 0	0 0
24	0 0 0	0 0 1 1	0 0 0	0 0	0 0
1	0 0 0	0 0 0 0	1 1 1	0 0	0 0
2	0 0 0	0 0 0 0	1 1 1	0 0	0 0
17	0 0 0	0 0 0 0	1 1 1	0 0	0 0
19	0 0 0	0 0 0 0	1 0 0	0 0	0 0
20	0 0 0	0 0 0 0	1 1 1	0 0	0 0
23	0 0 1	0 0 0 0	1 1 0	0 0	0 0
10	0 0 0	0 0 0 0	0 0 0	1 1	0 0
11	0 0 0	0 0 0 0	0 0 0	1 0	0 1
12	0 0 0	0 0 0 0	0 0 0	1 1	1 0
14	0 0 0	0 0 0 0	0 0 0	1 1	0 0
16	0 0 0	0 0 0 0	0 0 0	1 1	0 0
22	0 0 0	0 0 0 0	0 0 0	1 1	0 0
9	0 0 0	0 0 0 0	0 0 0	1 1	1 1
5	0 0 0	0 0 0 0	0 0 0	0 1	1 0
13	0 0 0	0 0 0 0	0 0 0	0 0	1 1
15	0 0 0	0 0 0 0	0 0 0	1 1	1 1

Stanfel1985

	1 2 8	3 4 7	5 6	1 9 3	1 1 1 1 0 1 2 6	1 1 4 5
1	1 0 1	1 0 1	0 0	0 0	0 0 0 0	1 0
3	1 0 1	0 0 0	0 1	0 0	0 0 1 0	0 0
10	1 1 1	0 0 0	0 0	0 0	1 0 0 0	0 0
22	1 1 1	0 0 0	0 0	0 0	1 0 0 0	0 0
23	1 1 1	0 0 0	0 0	0 0	1 0 0 0	0 0
16	0 0 0	1 1 0	0 0	0 0	0 0 0 0	0 0
19	1 0 0	1 1 1	0 0	0 0	0 0 0 0	0 0
13	0 0 1	0 0 0	0 1	0 0	0 0 0 0	0 0
15	1 0 0	0 0 0	0 1	0 0	0 0 0 0	0 0
18	1 0 0	0 0 0	0 1	0 0	0 0 0 0	0 0
21	0 0 0	0 0 0	1 1	0 0	0 0 0 0	0 0
7	1 0 1	0 0 0	0 0	1 1	0 1 0 0	0 0
8	1 0 0	0 0 0	0 0	0 1	0 0 0 0	0 0
2	0 0 0	0 0 0	0 1	0 1	1 0 1 0	0 0
9	0 0 0	0 0 0	0 0	0 0	1 1 1 1	0 1
11	1 0 0	0 0 0	0 0	0 1	0 0 1 1	0 0
14	0 0 0	0 0 0	0 0	0 0	1 1 1 1	1 0
17	0 1 0	1 0 0	0 1	0 0	1 0 0 1	0 0
20	0 0 0	0 0 0	0 0	0 0	1 1 1 1	0 1
24	0 0 0	0 0 1	0 0	1 0	1 1 1 1	1 0
4	1 0 0	1 0 0	0 0	0 0	0 0 0 0	1 0
5	0 0 0	0 0 0	0 0	0 0	0 0 0 0	0 1
6	1 0 0	1 0 0	0 0	0 0	0 0 0 0	1 1
12	0 0 0	0 0 0	0 0	0 0	0 0 0 0	1 0

McCormick1972

	1 1 1 4 7 8 1 2	1 2 3	1 3 6 9 5	1 1 1 5 0 4 6
2	0 1 1 1 1 1	0 0	0 0 0 0	0 0 0 0
4	1 1 1 1 0 1	1 0	0 0 0 0	0 0 0 0
7	1 1 1 1 1 1	0 0	0 0 0 0	0 0 0 1
9	1 1 0 1 1 0	0 0	0 0 0 0	0 0 0 0
12	1 0 1 1 1 1	0 0	0 0 0 0	0 0 0 0
18	1 1 1 1 1 1	1 0	0 0 0 0	0 1 0 0
22	1 1 1 1 0 1	0 0	0 0 0 0	0 0 0 0
30	1 1 1 0 1 0	0 1	0 0 0 1	0 0 0 0
1	0 0 0 0 0 0	1 0	1 0 0 0	0 0 0 0
3	0 0 0 0 0 0	0 1	0 0 0 0	1 0 0 0
10	0 0 0 0 0 0	1 1	0 0 0 0	0 0 0 0
13	0 0 0 0 0 0	0 1	1 0 0 0	0 0 0 0
16	0 0 0 0 0 0	1 0	0 0 0 0	0 0 0 0
20	0 0 0 0 0 1	1 1	0 0 0 0	0 0 0 0
5	1 0 0 0 0 0	0 0	0 0 1 1	0 0 0 0
19	0 0 0 0 0 1	0 0	0 0 1 0	0 0 0 0
23	0 0 0 0 0 0	0 0	1 1 1 1	0 0 0 0
25	0 0 1 0 0 0	0 0	1 1 1 1	0 0 0 0
27	0 0 0 0 0 0	0 0	1 1 1 1	0 0 1 1
28	0 0 0 1 0 0	0 0	1 0 1 0	0 0 0 0
29	0 0 0 0 0 0	0 1	1 1 0 1	0 0 0 0
6	0 0 0 0 0 0	0 0	0 0 0 0	1 1 1 1
8	0 0 0 0 0 0	0 0	0 0 0 0	1 1 1 1
11	0 0 0 0 1 0	0 0	0 0 0 0	1 1 1 0
14	0 0 0 0 0 0	0 0	0 0 0 0	1 1 1 1
15	0 0 0 0 0 0	0 0	0 0 0 0	1 1 1 1
17	0 0 0 0 0 0	0 0	0 0 0 0	0 0 1 0
21	0 0 0 0 0 0	0 0	0 0 0 0	1 1 1 0
24	0 0 0 0 0 0	0 0	0 0 0 0	1 1 0 1
26	0 0 0 0 0 0	0 0	0 0 0 1	1 1 0 1

Srinivasan1990

	1 1 3	1 2 9 6	1 3 6 4	1 4 5 8 5	1 7 0	1 1 1 2
3	0 1	0 0 0	0 0 0	0 0 1 0	0 0	1 0
16	0 0	0 0 0	0 0 0	0 1 0 0	0 0	0 0
2	0 0	1 1 1	0 1 1	0 0 1 0	0 0	0 0
4	0 0	0 1 0	0 0 0	0 0 0 0	0 0	0 0
10	0 0	1 1 1	0 0 0	0 0 0 0	0 0	0 0
18	0 0	0 1 1	0 0 0	0 0 0 0	0 0	0 0
28	0 0	1 1 0	0 0 0	0 0 1 0	0 0	0 0
32	0 0	1 1 1	0 1 0	0 0 0 0	0 0	0 0
37	1 0	1 1 1	0 1 0	0 0 1 0	0 0	0 0
38	0 0	1 1 1	0 0 0	0 0 1 0	0 0	0 0
40	0 0	1 1 0	0 1 0	0 0 0 0	0 0	0 0
42	1 0	1 1 1	0 1 0	0 0 0 0	0 0	0 0
6	0 0	0 0 0	0 1 1	0 0 0 0	0 0	0 0
7	0 0	0 0 1	1 1 0	0 0 0 0	0 0	0 0
17	0 0	0 0 0	1 1 1	0 0 0 0	0 0	0 0
34	0 0	0 0 0	1 1 0	0 0 0 0	0 0	0 0
35	0 0	0 0 0	1 0 1	0 0 0 0	0 0	0 0
36	0 0	0 0 0	1 0 0	0 0 0 0	0 0	0 0
5	0 0	0 0 0	0 0 0	1 1 0 1	0 0	0 0
8	0 0	0 0 0	0 1 0	0 1 1 0	0 0	0 0
9	0 0	0 0 0	0 0 0	1 1 1 0	0 0	1 0
14	0 0	0 0 0	0 1 0	1 1 0 1	0 0	0 0
15	0 0	0 0 0	0 0 0	0 1 1 0	0 0	0 0
19	0 0	0 0 0	0 1 0	1 1 1 1	0 0	0 0
21	0 0	0 0 0	0 0 0	1 1 1 1	0 0	0 0
23	0 0	0 0 0	0 1 0	1 1 1 0	0 0	0 0
29	0 0	0 0 0	0 0 0	1 1 0 0	0 0	0 0
33	0 0	0 0 0	0 1 0	0 1 0 1	0 0	0 0
41	0 0	0 0 0	0 0 0	0 1 1 1	0 0	0 0
43	0 0	0 0 0	0 1 0	0 1 1 1	0 0	0 0
1	0 0	0 0 0	0 1 0	0 0 1 0	1 1	0 0
12	0 0	0 0 0	0 1 0	0 0 1 0	0 1	0 0
13	0 0	0 0 0	0 1 0	0 0 0 0	1 1	0 0
25	0 0	0 0 0	0 0 0	0 0 0 0	1 1	0 0
26	0 0	0 0 0	0 0 0	0 0 0 0	0 1	0 0
31	0 0	0 0 0	0 0 0	0 0 1 0	0 1	0 0
39	0 0	0 0 0	0 1 0	0 0 0 0	0 1	0 0
11	0 0	0 0 0	0 0 0	0 0 1 0	0 0	0 1
20	0 0	0 0 0	0 0 0	0 0 1 0	0 0	1 0
22	0 0	0 0 0	0 0 0	0 0 0 0	0 0	0 1
24	0 1	0 0 0	0 0 0	0 0 1 0	0 0	1 1
27	0 0	0 0 0	0 0 0	0 0 1 0	0 0	1 1
30	0 0	0 0 0	0 0 0	0 0 0 0	0 0	1 1

King1980

	1 2	3 4 5 6 7	8 9	1 1 0 3	1 1 1 1 1 2 4 5	1 1 1 6 7 8
10	1 0	0 0 1 1 0	1 0	0 0	0 0 0 0	0 0 0
23	1 1	0 0 1 1 0	1 0	0 0	0 0 0 0	0 0 0
2	0 0	1 1 1 0 1	0 0	1 0	0 0 0 0	0 0 0
5	0 0	1 1 1 1 1	0 0	0 0	0 0 0 0	0 0 0
6	0 0	0 1 0 0 1	0 0	0 0	0 1 0 0	0 0 0
9	0 0	1 1 1 1 0	0 0	0 1	0 0 0 0	0 0 0
8	0 0	1 1 1 1 0	0 0	0 1	0 0 0 0	0 0 0
12	0 0	1 1 1 0 1	0 0	1 0	0 0 0 0	0 0 0
15	0 0	1 1 1 1 0	0 0	0 0	0 0 0 1	0 0 0
17	0 0	1 1 1 1 0	0 0	0 0	0 0 0 1	0 0 0
1	0 0	0 0 0 0 0	1 0	0 0	0 0 0 1	0 0 0
3	0 0	0 0 0 1 0	1 1	0 0	0 0 0 0	0 0 0
20	0 0	0 0 0 0 0	1 0	0 0	0 0 1 0	0 0 0
24	0 0	0 0 1 1 0	1 1	0 0	0 0 0 0	0 0 0
19	0 0	0 0 0 0 0	0 0	1 1	0 1 0 0	0 0 0
22	0 0	0 1 0 0 0	0 0	0 1	0 1 0 0	0 0 0
7	0 0	0 0 0 0 0	0 0	0 0	1 1 0 1	0 0 0
13	1 0	0 0 0 0 0	0 0	0 0	0 1 0 1	0 0 0
14	0 0	0 0 0 0 0	0 0	0 1	1 1 0 0	0 0 0
18	0 0	0 0 0 0 0	0 0	0 0	1 1 1 1	0 0 0
21	0 0	0 0 0 0 0	0 0	0 0	1 1 1 0	0 0 0
4	0 0	0 0 1 0 0	1 0	0 0	0 0 0 0	1 1 0
11	0 0	0 0 1 0 0	0 0	0 0	0 0 0 0	0 0 1
16	0 0	0 0 1 0 0	0 0	0 0	0 0 0 0	0 0 1

Carrie1973

	1 1 1 1 3 5 6 2 3 8	1 1 1 2 1 6 9	1 2 4 8 4 0	1 7 0	1 1 9 5 7
1	1 1 1 1 1 1 1	0 0 0 1	1 1 1 0	0 0	0 1 0
2	0 1 0 1 1 0 0	0 0 0 0	0 0 0 0	0 1	1 0 0
4	0 0 1 0 0 1 0	0 0 0 0	0 0 0 0	0 0	0 0 0
10	1 0 1 1 1 1 1	0 0 0 0	0 0 0 0	0 0	1 0 0
11	1 1 1 0 1 0 0	0 0 0 0	0 0 0 0	0 0	0 0 0
15	1 1 1 1 1 1 1	1 0 0 0	0 1 1 1	0 0	0 0 1
20	0 0 0 0 1 1 0	0 0 1 0	0 0 0 0	0 0	0 0 0
5	0 0 0 0 0 0 0	1 1 1 1	0 0 0 0	0 0	1 0 0
14	0 0 0 0 0 0 0	1 0 1 0	0 0 0 0	0 0	0 0 0
3	1 0 0 0 0 0 1	0 0 0 1	1 1 0 1	0 0	0 0 0
12	0 0 0 0 0 0 0	0 0 0 0	1 1 1 1	0 0	0 1 0
16	0 0 0 0 0 0 0	0 0 0 0	0 1 1 1	0 1	0 0 0
22	1 1 0 1 0 0 0	0 0 0 0	1 1 1 1	0 1	1 1 0
6	0 0 0 0 0 0 0	0 0 0 0	0 0 0 0	0 1	0 0 1
7	0 1 0 0 0 0 0	0 1 1 0	0 0 0 0	1 1	1 0 1
8	0 0 0 0 0 0 0	0 0 0 0	0 0 0 0	1 1	0 0 0
9	0 0 0 0 0 0 0	0 0 0 0	0 0 0 0	1 1	0 0 0
17	0 0 0 0 0 0 0	0 0 0 0	1 0 0 0	0 1	0 0 0
13	0 0 0 0 0 0 0	1 0 0 0	0 1 0 0	0 0	0 0 1
18	0 1 0 0 1 1 0	0 0 1 0	0 0 0 0	0 0	1 1 1
19	0 0 0 0 0 0 0	0 0 0 0	0 0 0 0	0 0	1 1 1
21	1 0 0 0 0 0 0	0 0 0 0	1 0 0 0	0 0	1 1 0
23	1 0 0 0 0 0 0	0 0 0 0	1 0 0 0	0 0	1 1 0

Kumar1986

	1 1 6	1 1 1 2 4 3 4 8	1 3 7 8 7	1 2 5 6 9 0 0	1 1 1 1 1 2 5 9
16	0 0	0 0 0 0 0	0 0 0 0	0 0 0 1 0	0 0 0 0
18	1 1	1 0 0 1 0	0 0 0 0	0 0 0 0 0	0 0 0 0
25	1 0	0 0 0 0 0	0 0 0 1	0 1 0 0 0	0 0 0 0
30	1 1	0 0 0 0 0	0 1 0 0	0 1 0 0 0	0 0 1 1
32	1 1	0 0 0 0 0	0 1 0 0	0 0 1 0 0	0 0 0 1
34	1 0	0 0 0 0 0	0 0 0 0	0 0 0 0 0	0 0 0 0
2	0 0	1 1 1 1 1	0 0 0 0	0 0 0 0 0	0 0 0 0
7	0 0	0 1 0 1 1	0 1 0 0	0 0 0 0 0	0 0 0 0
10	0 0	1 0 0 1 0	0 0 0 0	0 0 0 0 0	0 0 0 0
12	0 1	1 1 1 1 1	0 1 0 0	0 0 0 0 0	0 0 0 0
13	0 0	1 1 1 1 1	0 0 0 0	0 0 0 0 0	0 0 0 0
24	1 0	1 1 1 1 1	0 1 0 0	0 0 0 0 0	0 0 0 0
27	0 0	0 1 0 1 0	0 0 0 0	0 0 0 0 0	0 0 0 0
31	0 0	1 0 0 1 1	1 1 0 0	0 0 0 0 0	0 0 0 0
1	1 0	0 0 0 0 0	1 1 1 1	0 0 0 0 0	0 0 0 0
3	0 0	0 0 0 0 0	1 1 0 1	0 0 0 0 0	0 0 0 0
5	0 0	0 0 0 0 0	1 1 1 1	1 0 0 0 0	0 0 0 0
15	1 0	0 0 0 0 0	1 1 1 1	0 0 0 0 0	0 0 0 0
17	0 0	0 0 0 0 0	0 1 1 1	1 0 0 0 0	0 0 0 0
20	1 0	0 0 0 0 1	0 1 1 1	0 0 0 0 0	0 0 0 0
29	0 0	0 0 0 0 0	1 1 0 0	0 0 0 0 0	0 0 0 0
35	1 0	0 0 0 0 0	0 0 1 1	0 0 1 0 0	0 0 0 0
8	0 0	0 0 0 0 0	0 0 0 0	0 1 1 1 1	0 0 0 0
14	1 0	0 0 0 0 0	0 0 0 0	1 1 1 1 1	0 0 0 0
19	0 0	1 0 0 0 0	0 1 0 0	0 1 1 1 1	0 0 0 0
22	0 0	0 0 0 0 0	0 1 0 0	0 1 0 1 0	0 0 0 0
23	1 0	0 0 0 0 0	0 1 1 0	1 0 1 0 1	0 0 0 0
26	0 1	0 0 0 0 0	0 1 0 0	1 0 0 1 1	0 0 0 0
4	0 1	0 0 0 0 0	0 0 0 0	0 0 0 0 0	1 1 1 1
6	1 1	0 0 0 0 0	0 0 0 0	0 0 0 0 0	1 1 1 0
9	0 1	0 0 0 0 0	0 0 1 0	0 0 0 0 0	1 1 1 1
11	1 0	0 0 0 0 0	0 0 0 0	0 0 0 0 0	1 1 1 1
21	0 0	0 0 0 0 0	0 0 0 0	0 0 0 0 0	1 1 1 1
28	0 0	0 0 0 0 0	0 0 0 0	0 0 0 0 0	0 0 1 1
33	0 0	0 0 0 0 0	0 0 0 0	0 0 0 0 0	1 1 0 0

Boe1991

	1 1 3 7 8 7	1 1 2 4 4 8	1 5 3	1 2 6 9 0 0	1 1 1 1 1 1 2 5 6 9
1	1 1 1 1 1	0 0 0 0	0 0	0 0 0 0	0 0 0 0 0
3	1 1 1 1 1	0 0 0 0	0 0	0 0 0 0	0 0 0 0 0
5	0 1 1 1 1	0 0 0 0	0 0	0 0 0 0	0 0 0 0 0
15	0 1 1 1 1	0 0 0 0	0 0	0 0 0 0	0 0 0 0 0
17	0 1 1 1 1	0 0 0 0	0 0	0 0 0 0	0 0 0 0 0
20	1 0 1 1 0	0 0 0 0	0 0	0 0 0 0	0 0 0 0 0
23	1 0 1 1 1	0 0 0 0	0 0	0 0 0 0	0 0 0 0 0
25	1 0 0 1 1	0 0 0 0	0 0	0 0 0 0	0 0 0 0 0
29	1 1 0 1 1	0 0 0 0	0 0	0 0 0 0	0 0 0 0 0
2	0 0 0 0 0	1 1 1 1	0 1	0 0 0 0	0 0 0 0 0
7	0 0 0 0 0	1 1 1 0	0 0	0 0 0 0	0 0 0 0 0
10	0 0 0 0 0	1 0 1 1	0 0	0 0 0 0	0 0 0 0 0
12	0 0 0 0 0	1 1 1 1	0 1	0 0 0 0	0 0 0 0 0
13	0 0 0 0 0	1 1 1 1	0 1	0 0 0 0	0 0 0 0 0
18	0 0 0 0 0	1 0 1 1	0 0	0 0 0 0	0 0 0 0 0
24	0 0 0 0 0	1 1 1 1	0 1	0 0 0 0	0 0 0 0 0
27	0 0 0 0 0	1 1 1 0	0 0	0 0 0 0	0 0 0 0 0
31	0 1 0 0 0	1 0 1 1	0 0	0 0 0 0	0 0 0 0 0
33	0 0 0 0 0	0 0 0 0	0 0	0 0 0 0	0 1 0 0 0
34	0 0 0 0 0	0 0 0 0	1 0	1 0 0 0	0 0 0 0 0
35	0 0 0 0 0	0 0 0 0	0 0	0 0 0 0	1 0 0 0 0
8	0 0 0 0 0	0 0 0 0	1 0	1 1 1 1	0 0 0 0 0
14	0 0 0 0 0	0 0 0 0	1 0	1 1 1 1	0 0 0 0 0
16	0 0 0 0 0	0 0 0 0	1 0	1 0 1 0	0 0 0 0 0
19	0 0 0 0 0	0 0 0 0	0 0	1 1 1 1	0 0 0 0 0
22	0 0 0 0 0	0 0 0 0	0 0	1 1 1 0	0 0 0 0 0
26	0 0 0 0 0	0 0 0 0	0 0	1 1 1 1	0 0 0 0 0
4	0 0 0 0 0	0 0 0 0	0 0	0 0 0 0	1 1 1 1 1
6	0 0 0 0 0	0 0 0 0	0 0	0 0 0 0	1 1 1 1 0
9	0 0 0 0 0	0 0 0 0	0 0	0 0 0 0	1 1 1 1 1
11	0 0 0 0 0	0 0 0 0	0 0	0 0 0 0	1 1 1 1 1
21	0 0 0 0 0	0 0 0 0	0 0	0 0 0 0	1 1 1 1 1
28	0 0 0 0 0	0 0 0 0	0 0	0 0 0 0	1 0 1 1 1
30	0 0 0 0 0	0 0 0 0	0 0	0 0 0 0	1 0 1 1 1
32	0 0 0 0 0	0 0 0 0	0 0	0 0 0 0	1 0 0 1 1

Carrie1973

	1 2 2 1 3 1 2	1 1 2 5 1 9	2 3 0	1 4 6	1 1 1 6 8 2 5 8	1 1 9 0 7	1 2 2 7 4 3 4
1	1 1 1 1	0 0 0 0	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
9	1 1 1 1	0 0 0 0	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
16	1 1 1 1	0 0 0 0	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
17	1 1 1 1	0 0 0 0	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
33	1 1 1 1	0 0 0 0	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
10	0 0 0 0	1 1 1 1	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
13	0 0 0 0	1 1 1 1	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
14	0 0 0 0	1 1 1 1	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
22	0 0 0 0	1 1 1 1	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
35	0 0 0 0	1 1 1 1	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
36	0 0 0 0	1 1 1 1	0 0	0 0	0 0 0 0 0	0 0 0	0 0 0 0
2	0 0 0 0	0 0 0 0	1 1	0 0	0 0 0 0 0	0 0 0	0 0 0 0
11	0 0 0 0	0 0 0 0	1 1	0 0	0 0 0 0 0	0 0 0	0 0 0 0
12	0 0 0 0	0 0 0 0	1 1	0 0	0 0 0 0 0	0 0 0	0 0 0 0
15	0 0 0 0	0 0 0 0	1 1	0 0	0 0 0 0 0	0 0 0	0 0 0 0
23	0 0 0 0	0 0 0 0	1 1	0 0	0 0 0 0 0	0 0 0	0 0 0 0
24	0 0 0 0	0 0 0 0	1 1	0 0	0 0 0 0 0	0 0 0	0 0 0 0
31	0 0 0 0	0 0 0 0	1 1	0 0	0 0 0 0 0	0 0 0	0 0 0 0
34	0 0 0 0	0 0 0 0	1 1	0 0	0 0 0 0 0	0 0 0	0 0 0 0
8	0 0 0 0	0 0 0 0	0 0	1 1	0 0 0 0 0	0 0 0	0 0 0 0
19	0 0 0 0	0 0 0 0	0 0	1 1	0 0 0 0 0	0 0 0	0 0 0 0
21	0 0 0 0	0 0 0 0	0 0	1 1	0 0 0 0 0	0 0 0	0 0 0 0
28	0 0 0 0	0 0 0 0	0 0	1 1	0 0 0 0 0	0 0 0	0 0 0 0
37	0 0 0 0	0 0 0 0	0 0	1 1	0 0 0 0 0	0 0 0	0 0 0 0
38	0 0 0 0	0 0 0 0	0 0	1 1	0 0 0 0 0	0 0 0	0 0 0 0
39	0 0 0 0	0 0 0 0	0 0	1 1	0 0 0 0 0	0 0 0	0 0 0 0
4	0 0 0 0	0 0 0 0	0 0	0 0	1 1 1 1 1	0 0 0	0 0 0 0
5	0 0 0 0	0 0 0 0	0 0	0 0	1 1 1 1 1	0 0 0	0 0 0 0
18	0 0 0 0	0 0 0 0	0 0	0 0	1 1 1 1 1	0 0 0	0 0 0 0
26	0 0 0 0	0 0 0 0	0 0	0 0	1 1 1 1 1	0 0 0	0 0 0 0
27	0 0 0 0	0 0 0 0	0 0	0 0	1 1 1 1 1	0 0 0	0 0 0 0
30	0 0 0 0	0 0 0 0	0 0	0 0	1 1 1 1 1	0 0 0	0 0 0 0
6	0 0 0 0	0 0 0 0	0 0	0 0	0 0 0 0 0	1 1 1	0 0 0 0
7	0 0 0 0	0 0 0 0	0 0	0 0	0 0 0 0 0	1 1 1	0 0 0 0
20	0 0 0 0	0 0 0 0	0 0	0 0	0 0 0 0 0	1 1 1	0 0 0 0
29	0 0 0 0	0 0 0 0	0 0	0 0	0 0 0 0 0	1 1 1	0 0 0 0
40	0 0 0 0	0 0 0 0	0 0	0 0	0 0 0 0 0	1 1 1	0 0 0 0
3	0 0 0 0	0 0 0 0	0 0	0 0	0 0 0 0 0	0 0 0	1 1 1 1
25	0 0 0 0	0 0 0 0	0 0	0 0	0 0 0 0 0	0 0 0	1 1 1 1
32	0 0 0 0	0 0 0 0	0 0	0 0	0 0 0 0 0	0 0 0	1 1 1 1

	2		1 1		2	1		1	1 1	1 1 1	1 2	2 2
	1 1	2 5 1 9	3 0	4 6	6 8	7 4	9 0 7	5 8 2	3 2	3 4		
9	1 1	0 0 0 0	0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0		
33	1 1	1 0 0 0	0 0	0 0	0 0	0 0	1 0 0	0 0 0	0 0	0 0		
10	0 1	0 1 1 0	0 0	0 0	1 0	0 0	0 0 0	0 0 0	0 0	0 0		
13	0 0	1 0 1 1	0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	1 0		
14	1 0	1 1 0 1	0 0	0 0	0 0	0 1	0 0 0	1 0 0	0 0	0 0		
22	0 0	1 1 1 0	0 0	0 0	0 0	0 0	0 0 0	0 0 1	0 0	0 0		
35	0 0	0 1 1 1	0 0	0 1	0 0	0 0	0 0 0	0 0 0	1 0	0 0		
36	0 0	1 1 0 1	0 0	0 1	0 0	0 0	0 0 0	0 0 0	0 0	0 0		
2	0 0	0 0 0 0	1 1	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 1		
11	0 0	0 0 0 0	0 1	1 0	0 0	0 0	0 0 0	0 1 0	0 0	0 0		
12	0 0	0 1 0 0	1 1	0 0	0 0	0 0	0 0 0	0 0 0	1 0	0 0		
15	0 0	0 0 1 0	1 1	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0		
23	0 0	0 0 1 0	1 1	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0		
24	0 0	0 0 0 0	1 0	0 0	0 0	0 0	0 1 0	0 0 1	0 0	0 0		
31	0 0	0 0 0 0	0 1	0 0	0 0	0 0	0 0 1	0 0 0	0 0	0 0		
34	0 0	0 0 0 0	1 1	0 0	0 1	0 0	0 0 0	0 0 0	0 0	0 0		
8	0 0	0 0 0 0	0 0	1 1	0 0	0 0	0 0 0	0 0 0	0 0	0 0		
19	0 0	0 0 0 0	0 1	1 1	0 0	0 0	0 0 0	0 0 0	0 0	0 0		
21	0 0	0 0 0 0	0 0	1 1	0 0	0 0	0 0 0	0 0 0	0 0	0 0		
28	0 0	0 0 0 0	0 0	1 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0		
37	0 0	0 0 0 0	0 0	0 1	0 0	0 0	0 0 0	1 0 0	0 0	0 0		
26	0 0	0 0 0 0	0 0	0 1	1 1	0 0	0 1 0	1 0 0	0 0	0 0		
30	0 0	0 0 0 0	0 0	0 0	1 1	0 0	0 0 0	0 1 0	0 1	0 0		
38	0 0	0 0 0 0	0 0	1 0	1 0	0 0	0 0 0	0 0 0	0 0	0 0		
39	0 0	0 0 0 0	0 0	0 0	0 1	0 0	1 0 0	0 0 0	0 0	0 0		
25	0 0	0 0 0 0	0 1	0 0	0 0	1 1	0 0 0	0 0 0	0 0	0 0		
40	0 0	0 0 0 1	0 0	0 0	0 0	1 0	1 0 0	0 0 0	0 0	0 0		
6	0 0	0 0 0 0	0 0	0 0	0 0	0 0	1 1 0	0 0 0	0 0	0 0		
7	0 0	0 0 0 0	0 0	0 0	0 0	0 0	0 0 1	0 0 0	0 0	0 0		
20	0 0	0 0 0 0	0 0	0 0	0 0	0 0	0 1 1	0 0 1	0 0	0 0		
29	0 0	0 0 0 0	0 0	0 0	0 0	0 0	1 0 1	0 0 0	0 0	0 0		
4	0 0	0 0 0 0	0 0	0 0	0 1	0 0	0 0 0	1 1 1	0 0	0 0		
5	0 0	0 1 0 0	0 0	0 0	0 1	0 0	0 0 0	1 1 1	0 0	0 0		
18	0 0	0 0 0 0	0 0	0 0	1 0	0 0	0 0 0	0 1 1	0 0	0 0		
27	0 0	0 0 0 1	0 0	0 0	0 0	0 0	0 0 0	1 1 1	0 0	0 0		
1	0 1	0 0 0 0	0 0	0 0	0 0	0 0	0 0 0	1 0 0	1 1	0 0		
16	0 1	0 0 0 0	0 0	0 0	0 0	0 1	0 0 0	0 0 0	1 1	0 0		
17	1 0	0 0 0 0	0 0	0 0	0 0	0 0	0 0 1	0 0 0	1 1	0 0		
3	0 1	0 0 0 0	0 0	0 0	0 0	0 1	0 0 0	0 0 0	0 0	1 1		
32	1 0	0 0 0 0	0 0	0 1	0 0	1 0	0 0 0	0 0 0	0 0	1 1		

	1 1 1	1 2 2	2 2 3 1 2	2 4 7	1 5 5	1 6 6	1 1 2 7 7 8 6	2 8 8	1 2 2 3 9 9 0 9 0	1 2 0 3	1 2 3 4	1 2 4 5
10	1 1	0 0	0 1 1	0 0	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	0 0	0 0	0 0
33	1 1	1 0	0 1 0	0 0	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	0 0	0 0	0 0
41	1 1	1 0	0 0 0	0 0	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	0 0	0 0	0 0
11	0 0	1 1	0 0 0	0 0	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	0 0	0 0	0 0
18	0 0	1 1	0 0 0	0 0	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	0 0	1 0	1 0
19	0 0	0 1	0 0 0	0 0	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	0 0	0 0	0 0
24	0 0	0 1	0 0 0	1 0	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	0 0	0 0	0 0
39	1 1	1 1	1 0 1	0 0	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	1 1	0 0	0 0
12	0 0	1 1	1 1 1	0 0	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	1 1	0 0	0 0
31	1 0	0 0	1 1 1	0 0	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	0 1	0 0	0 0
32	1 0	1 0	1 1 1	0 0	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	0 0	0 0	0 0
40	0 0	0 1	1 1 1	0 0	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	0 0	0 0	0 0
7	0 0	0 0	0 0 0	1 0	0 0	0 0	0 1 0 0	0 0	0 0 0 0 0	0 0	0 0	0 0
15	0 0	0 0	0 0 0	1 1	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	0 0	0 0	0 0
35	0 0	0 0	0 0 0	1 0	0 0	0 0	0 0 0 0	0 1	0 0 0 0 0	0 0	0 0	0 0
5	0 0	0 0	0 0 0	0 0	1 0	0 0	0 0 0 0	0 0	0 0 0 0 0	0 0	0 0	1 0
17	0 0	0 0	0 0 0	0 0	1 1	0 0	0 0 0 0	0 0	0 0 0 0 0	0 0	0 0	0 0
6	0 0	0 0	0 0 0	0 0	0 0	1 1	0 0 0 0	0 0	0 0 0 0 0	0 0	0 0	0 0
28	0 0	0 0	0 0 0	1 0	0 0	0 1	0 0 0 0	0 0	0 0 0 0 0	0 0	0 0	0 0
16	0 0	0 0	0 0 0	0 0	0 0	0 0	1 1 1 1	0 0	0 0 0 0 0	0 0	0 0	0 0
27	0 0	0 0	0 0 0	0 0	0 0	0 0	1 1 1 1	0 0	0 0 0 0 0	0 0	0 0	0 0
34	0 0	0 0	0 0 0	0 0	1 0	0 0	1 0 1 1	0 0	0 0 0 0 0	0 0	0 0	0 0
36	0 0	0 0	0 0 0	0 0	1 0	0 0	1 1 0 1	0 0	0 0 0 0 0	0 0	0 0	0 0
8	0 0	0 0	0 0 0	0 1	0 0	0 0	0 0 0 0	1 1	0 0 0 0 0	0 0	0 0	0 0
9	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0 0	1 0	0 0 0 1 0	0 0	0 0	0 0
14	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 1 0	1 0	0 0 0 0 0	0 0	0 0	0 0
29	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0 0	1 1	0 0 0 0 0	0 0	0 0	0 0
1	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0 0	1 0	0 1 1 1 1	0 0	0 0	0 0
3	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0 0	0 0	1 1 1 1 1	0 0	0 0	0 0
13	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0 0	0 0	1 0 0 1 0	0 0	0 0	0 0
21	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0 0	0 0	1 1 1 1 1	0 0	0 0	0 0
22	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0 0	0 0	0 1 1 0 1	0 0	0 0	0 0
30	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0 0	0 0	0 1 0 1 0	0 0	0 0	0 0
2	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	1 1	0 0	0 0
20	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	1 0	0 0	0 0
23	0 0	0 1	1 0 1	0 0	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	1 1	0 0	0 0
25	0 0	0 0	0 0 0	1 0	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	0 0	1 0	0 0
38	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	0 0	1 1	0 0
4	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0 0	0 0	0 0 0 0 0	0 0	0 0	1 1
26	0 0	0 0	0 0 0	0 0	0 0	1 0	0 0 0 0	0 0	0 0 0 0 0	0 0	0 0	1 0
37	0 0	0 0	0 0 0	0 0	0 1	0 0	0 0 0 0	0 0	0 0 0 0 0	0 0	0 0	1 0

		1 1 4 5 1	2 9	1 3 0	1 6 8 2	1 7 3	1 1 4 6	1 1 5 7	1 2 2 8 0 2	1 2 2 9 1 3	2 3 4 0	2 2 2 5 7 9	2 2 6 8
1	1	1 1 1 1	0 1	0 0	0 0 0	0 1	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
7	7	0 1 1 1	0 0	0 1	0 1 1	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
18	18	1 1 0 1	0 0	0 1	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
2	2	0 0 1 0	1 1	1 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
3	3	0 0 1 0	1 1	0 0	0 0 1	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
17	17	0 0 0 0	1 1	0 0	0 0 0	0 1	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
4	4	0 0 0 1	0 0	1 1	0 0 0	1 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
6	6	1 0 0 0	0 1	1 1	0 0 0	0 1	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
8	8	1 0 0 0	0 0	1 0	0 0 1	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
10	10	0 0 1 0	0 0	0 1	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
11	11	0 0 1 0	0 0	1 1	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
5	5	1 0 0 0	0 0	0 0	1 1 0	0 1	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
13	13	0 0 0 0	0 0	0 0	1 1 0	1 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
14	14	0 0 0 0	0 0	0 0	1 1 1	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
15	15	0 0 0 0	0 0	0 0	1 1 1	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
16	16	0 0 0 0	0 0	0 0	1 1 1	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
9	9	0 0 0 0	1 0	0 1	1 0 0	1 1	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
12	12	0 0 0 0	0 0	0 0	0 1 0	1 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
19	19	0 0 0 0	0 0	0 0	0 0 0	0 0	1 1	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
21	21	0 0 0 0	0 0	0 0	0 0 0	0 0	1 1	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
23	23	0 0 0 0	0 0	0 0	0 0 0	0 0	1 1	0 1	0 0 0	0 0 0	0 0	0 0 0	0 0
26	26	0 0 0 0	0 0	0 0	0 0 0	0 0	1 1	0 0	0 0 0	0 0 0	0 0	0 0 0	0 0
20	20	0 0 0 0	0 0	0 0	0 0 0	0 0	1 0	1 1	0 0 0	0 0 0	0 0	0 0 0	0 0
22	22	0 0 0 0	0 0	0 0	0 0 0	0 0	0 1	1 0	0 0 0	0 0 0	0 0	0 0 0	0 0
24	24	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	1 1	0 0 0	0 0 0	0 0	0 0 0	0 0
25	25	0 0 0 0	0 0	0 0	0 0 0	0 0	1 0	1 1	0 0 0	0 0 0	0 0	0 0 0	0 0
27	27	0 0 0 0	0 0	0 0	0 0 0	0 0	0 1	1 0	0 0 0	0 0 0	0 0	0 0 0	0 0
28	28	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	1 1 0	1 0 0	0 0	0 0 0	0 0
30	30	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	1 0 1	0 1 0	0 0	0 0 0	0 0
32	32	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	0 1 1	0 1 0	0 0	0 0 0	0 0
34	34	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	1 0 1	1 0 0	0 0	0 0 0	0 0
35	35	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	0 1 1	0 0 1	0 0	0 0 0	0 0
36	36	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	1 1 0	0 1 0	0 0	0 0 0	0 0
37	37	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	1 1 1	0 0 0	0 0	0 0 0	0 0
29	29	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	0 1 0	1 0 1	0 0	0 0 0	0 0
31	31	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	1 1 1	0 0	0 0 0	0 0
33	33	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	1 1 1	0 0	0 0 0	0 0
38	38	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	1 1 1	0 0	0 0 0	0 0
39	39	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	1 1	0 0 1	0 1
42	42	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	1 1	1 0 0	0 0
44	44	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	1 1	0 1 0	0 0
46	46	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 1	0 0 0	1 0
48	48	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	1 1	0 0 1	0 0
40	40	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 1	0 1 1	0 0
41	41	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	1 1 1	0 0
45	45	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	1 0 1	0 1
49	49	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	1 1 0	1 0
50	50	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	1 0 1	0 0
43	43	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 0 0	1 1
47	47	0 0 0 0	0 0	0 0	0 0 0	0 0	0 0	0 0	0 0 0	0 0 0	0 0	0 1 0	0 1

	3	1 1 2 3	2	2 2 3 3	1 2 3 4	1 1	1 1 3	1 1 3 3 3	1 2 2 3	2 2 2
	1 3 7 2	2 0 6 1 1	4 9 0	5 8 2 3 7 9	6 2 6 8 0	1 3	4 7 5	5 8 3 4 6	9 5 8 0	4 7 9
10	1 0 1 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
18	1 1 1 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
29	1 1 0 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
33	1 1 1 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
34	1 1 1 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
37	1 1 1 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
41	0 1 1 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 1 0
44	1 1 1 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
49	1 1 1 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
50	1 1 0 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 1 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
54	1 1 1 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
55	1 1 1 1	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
35	0 0 0 0	1 1 1 1 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
47	0 0 0 0	1 1 1 0 1	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
53	0 0 0 0	1 1 0 1 1	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
77	0 0 0 0	0 1 1 1 1	0 0 0	0 0 0 1 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
78	0 0 0 0	1 1 1 1 0	0 0 0	0 0 0 0 0 0	0 0 0 0 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
79	0 0 0 0	1 1 1 1 1	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0
83	0 0 0 0	1 1 1 1 1	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 1	0 0 0
88	0 0 0 0	1 1 1 1 1	0 0 0	0 0 0 0 0 0	0 0 1 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
93	0 0 0 0	1 1 1 1 1	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 1	0 0 0 0	0 0 0
8	0 0 0 0	0 0 0 0 0	1 1 1	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
11	0 0 0 0	0 0 0 0 0	1 1 1	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
13	0 0 0 0	0 0 0 0 0	1 1 1	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
14	0 0 0 0	0 0 0 0 0	1 1 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 1	0 0 0
20	0 0 0 0	0 0 0 0 0	1 1 1	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
22	0 0 0 0	0 0 0 0 0	1 1 1	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
23	0 0 0 0	0 0 0 0 0	1 1 1	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
32	0 0 0 0	0 0 0 0 0	1 1 1	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
63	0 0 0 0	0 0 0 0 0	0 0 0	1 1 0 0 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
66	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 0 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
68	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
69	0 0 0 0	0 0 0 0 0	0 0 0	1 0 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
82	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
84	0 0 0 0	0 0 0 0 0	0 1 0	1 1 1 1 1 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
85	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
89	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
90	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
92	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
94	0 0 0 0	0 0 0 0 0	0 0 0	1 0 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 1	0 0 0 0	0 0 0
95	0 0 0 0	0 0 0 0 0	0 0 0	1 1 0 1 1 1	0 0 0 0 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
96	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
97	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 0	0 1 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
98	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
99	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	1 0 0 0	0 0 0
100	0 0 0 0	0 0 0 0 0	0 0 0	1 1 1 1 1 1	0 0 0 0 0	0 0	0 0 0	0 1 0 0 0	0 0 0 0	0 0 0
36	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 1 1 1 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
38	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 1 1 1 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
42	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 0 1 1 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
51	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 1 1 1 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
52	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 1 1 1 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
64	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 1 1 1 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
65	0 0 0 0	0 0 1 0 0	0 0 0	0 0 0 0 0 0	1 0 1 1 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
70	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 1 1 1 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
72	0 0 0 0	0 0 0 1 0	0 0 0	0 0 0 0 0 0	1 1 1 0 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
74	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 1 1 1 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
75	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 1 1 0 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
76	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 1 1 1 1	0 0	0 0 0	0 0 0 0 1	0 0 0 0	0 0 0
80	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 1 1 1 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
81	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	1 1 0 1 1	0 0	0 1 0	0 0 0 0 0	0 0 1 0	0 0 0
86	0 0 0 0	0 0 1 0 0	0 0 0	0 1 0 0 0 0	1 1 1 1 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
87	0 0 0 0	0 0 0 0 0	0 0 0	1 0 0 0 0 0	1 1 1 1 1	0 0	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
1	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
2	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
3	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
4	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
5	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
7	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
21	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	0 1 0	0 0 0 0 0	0 0 0 0	0 0 0
25	0 0 1 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
28	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	1 0 0	0 0 0 0 0	0 0 0 0	0 0 0
48	0 1 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 1	0 0 0	1 0 0 0 0	0 0 0 0	0 0 0
6	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	1 0 1	0 0 0 0 0	0 0 0 0	0 0 0
15	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	1 1 1	0 0 0 0 0	0 0 0 0	0 0 0
16	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	1 1 1	0 0 0 0 0	0 0 0 0	0 0 0
24	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	1 1 1	0 0 0 0 0	0 0 0 0	0 0 0
27	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	1 0	1 0 1	0 0 0 0 0	0 0 0 0	0 0 0
60	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 1	1 1 1	0 0 0 0 0	0 0 0 0	0 0 0
39	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	1 0 0	0 1 0 1 1	0 0 0 0	0 0 0
45	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	1 0 1 1 1	0 0 0 0	0 0 0
67	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	1 1 1 1 1	0 0 0 0	0 0 0
71	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	1 1 1 1 0	0 1 0 0	0 0 0
73	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 1 1 1 1	1 0 0 0	0 0 0
91	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	1 1 1 1 1	0 0 0 1	0 0 0
56	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	1 1 1 1	0 0 0
57	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	1 1 1 1	0 0 0
58	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	1 1 1 1	0 0 0
59	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	1 1 1 1	0 0 0
61	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	1 1 1 1	0 0 0
62	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	1 1 1 1	0 0 0
9	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	1 1 1
12	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0	0 0 0	0 0 0 0 0	0 0 0 0	1 1 1
17	0 0 0 0	0 0 0 0 0	0 0 0	0 0 0 0 0 0	0					

Apêndice D

Erros encontrados na literatura do PFCM

D.1 Instâncias Erradas

James et al. [31]:

- King1982: 0 em M5P5 (Máquina 5, Parte 5) e M6P6;
- Kumar1987: 0 em M5P5 M28P29 M29P9 M(28ou27?)P8, 1 em M19P9 M19P29 M19P8.

Wu et al. [65]:

- Stanfel1985_1: 0 em M15P27.

D.2 Resultados Incoerentes

Artigos com resultados suspeitos (sem restrição de *singleton*):

Wu et al. [64]: Stanfel1985_1

Mesmo não visível, o erro se torna visível no próximo artigo do mesmo autor.

Wu et al. [65]: Stanfel1985_1

O erro está visível neste artigo, na resposta com restrição de *singleton*.

Mahdavi et al. [43]: Stanfel1985_1

Referencia Wu et al. [65], onde o erro está visível.

James et al. [31]: King1982 e Kumar1987

O erro está visível neste artigo.

Elbenani et al. [21]: King1982, Kumar1987 e Stanfel1985_1

Referencia James et al. [31] (que explica o erro na King1982 e Kumar1987) e o Mahdavi et al. [43] (que referencia o Wu et al. [65], explicando o erro na Stanfel1985_1).

Artigos com resultados suspeitos (com restrição de singleton):

Wu et al. [65]: Stanfel1985_1

O erro está visível neste artigo, na resposta com restrição de singleton.

Wu et al. [66]: Stanfel1985_1

Referencia Wu et al. [65], onde o erro está visível.

Bibliografia

- [1] ABDULLAH, A. ; HUSSAIN, A. A new biclustering technique based on crossing minimization. *Neurocomputing* 69, 16–18 (2006), 1882–1896.
- [2] ALPERT, C. J. ; KAHNG, A. B. Recent directions in netlist partitioning: a survey. *Integration, the VLSI Journal* 19 (1995), 1–81.
- [3] AMIT, N. *The Bicluster Graph Editing Problem*, Tel Aviv University, Dissertação de Mestrado, 2004.
- [4] APPLEGATE, D. L. ; BIXBY, R. E. ; CHVÁTAL, V. ; COOK, W. J. *The Traveling Salesman Problem: A Computational Study*. Princeton University Press, 2006.
- [5] ASKIN, R. G. ; SUBRAMANIAN, S. P. A cost-based heuristic for group technology configuration. *International Journal of Production Research* 25 (1987), 101.
- [6] BISSON, G. ; HUSSAIN, F. Chi-Sim: A New Similarity Measure for the Co-clustering Task. In *Proceedings of the 2008 Seventh International Conference on Machine Learning and Applications*, (2008), pp. 211–217.
- [7] BOCTOR, F. A linear formulation of the machine-part cell formation problem. *International Journal of Production Research* 29, 2 (1991), 343–356.
- [8] BOE, W. J. ; CHENG, C. H. A close neighbour algorithm for designing cellular manufacturing systems. *International Journal of Production Research* 29, 10 (1991), 2097–2116.
- [9] BURBIDGE, J. L. Production Flow Analysis. *Production Engineer* 50, 4/5 (1971), 139–152.
- [10] CARRIE, A. S. Numerical taxonomy applied to group technology and plant layout. *International Journal of Production Research* 11, 4 (1973), 399–416.
- [11] CHAN, H. M. ; MILNER, D. A. Direct clustering algorithm for group formation in cellular manufacture. *Journal of Manufacturing Systems* 1 (1982), 63.

-
- [12] CHANDRASEKHARAN, M. P. ; RAJAGOPALAN, R. An Ideal seed non-hierarchical clustering algorithm for cellular manufacturing. *International Journal of Production Reseach* 24, 2 (1986), 451–464.
 - [13] CHANDRASEKHARAN, M. P. ; RAJAGOPALAN, R. MODROC: An extension of rank order clustering for group technology. *International Journal of Production Reseach* 24, 5 (1986), 1221–1233.
 - [14] CHANDRASEKHARAN, M. P. ; RAJAGOPALAN, R. ZODIAC - an algorithm for concurrent formation of part-families and machine-cells. *International Journal of Production Research* 25, 6 (1987), 835–850.
 - [15] CHANDRASEKHARAN, M. P. ; RAJAGOPALAN, R. Groupability an analysis of the properties of binary data matrices for group technology. *International Journal of Production Research* 27 (1989), 1035.
 - [16] CHENG, Y. ; CHURCH, G. M. Biclustering of Expression Data. In *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology*, (2000), pp. 93–103.
 - [17] CHVÁTAL, V. Edmonds polytopes and a hierarchy of combinatorial problems. *Discrete Mathematics* 4, 4 (1973), 305–337.
 - [18] CORMEN, T. H. ; LEISERSON, C. E. ; RIVEST, R. L. ; STEIN, C. *Introduction to Algorithms*. The MIT Press, 2001.
 - [19] DANTZIG, G. ; FULKERSON, R. ; JOHNSON, S. Solution of a large-scale traveling-salesman problem. *Operations Research* 2 (1954), 393–410.
 - [20] DANTZIG, G. B. *Linear Programming and Extensions*. RAND Corporation, 1963.
 - [21] ELBENANI, B. ; FERLAND, J. A. ; BELLEMARE, J. Genetic algorithm and large neighbourhood search to solve the cell formation problem. *Expert Systems with Applications* 39, 3 (2012), 2408–2414.
 - [22] FAURE, N. ; CHRETIENNE, P. ; GOURDIN, E. ; SOURD, F. Biclique completion problems for multicast network design. *Discrete Optimization* 4, 3-4 (2007), 360–377.
 - [23] FLANDERS, R. E. Design, manufacture and production control of a standard machine. *Transactions of the American Society of Mechanical Enginneers* 46 (1924), 691–738.
 - [24] GILBERT, E. Random graphs. *Annals of Mathematical Statistics* 30 (1959), 1141–1144.

- [25] GOMORY, R. E. Outline of an algorithm for integer solutions to linear programs. *Bulletin of the American Mathematical Society* 64 (1958), 275–278.
- [26] GOMORY, R. E. An algorithm for integer solutions to linear programs. In GRAVES, R. L. (Hrsg.) ; WOLFE, P. (Hrsg.): *Recent Advances in Mathematical Programming*, (1963), pp. 269–302.
- [27] GONÇALVES, J. ; RESENDE, M. An evolutionary algorithm for manufacturing cell formation. *Computers & Industrial Engineering* 47, 2-3 (2004), 247–273.
- [28] GUO, J. ; HÜFFNER, F. ; KOMUSIEWICZ, C. ; ZHANG, Y. Improved Algorithms for Biclustering. In *5th International Conference on Theory and Applications of Models of Computation* 4978, (2008), pp. 445–456.
- [29] GUPTA, A. ; PALIT, A. Clique Generation Using Boolean Equations. *Proceedings of The IEEE* 67, 1 (1979), 178–180.
- [30] HARTIGAN, J. A. *Clustering Algorithms*. John Wiley & Sons, 1975.
- [31] JAMES, T. L. ; BROWN, E. C. ; KEELING, K. B. A hybrid grouping genetic algorithm for the cell formation problem. *Comput. Oper. Res.* 34, 7 (2007), 2059–2079.
- [32] KARP, R. Reducibility among combinatorial problems of computer computations. In *Proceedings of a Symposium on the Complexity of Computer Computations*, (1972), pp. 88–104.
- [33] KING, J. R. ; NAKORNCHAI, V. Machine-component group formation in group technology: review and extension. *International Journal of Production Research* 20, 2 (1982), 117–133.
- [34] KING, J. Machine-component grouping in production flow analysis: An approach using a rank order clustering algorithm. *International Journal of Production Research* 18, 2 (1980), 213–232.
- [35] KUMAR, C. S. ; CHANDRASEKHARAN, M. P. Group efficacy: a quantitative criterion for goodness of block diagonal forms of binary matrices in group technology. *International Journal of Production Research* 28, 2 (1990), 233–243.
- [36] KUMAR, K. R. ; KUSIAK, A. ; VANNELLI, A. Grouping of parts and components in flexible manufacturing systems. *European Journal of Operational Research* 24, 3 (1986), 387–397.
- [37] KUMAR, K. R. ; VANNELLI, A. Strategic subcontracting for efficient disaggregated manufacturing. *International Journal of Production Research* 25, 12 (1987), 1715–1728.

- [38] KUSIAK, A. ; CHO, M. Similarity coefficient algorithms for solving the group technology problem. *International Journal of Production Research* 30, 11 (1992), 2633–2646.
- [39] KUSIAK, A. ; CHOW, W. Efficient solving of the group technology problem. *Journal of Manufacturing Systems* 6, 2 (1987), 117–124.
- [40] LAND, A. H. ; DOIG, A. G. An Automatic Method of Solving Discrete Programming Problems. *Econometrica* 28, 3 (1960), 497–520.
- [41] LITTLE, J. D. C. ; MURTY, K. G. ; SWEENEY, D. W. ; KAREL, C. An Algorithm for the Traveling Salesman Problem. *Operations Research* 11, 6 (1963), 972–989.
- [42] MADEIRA, S. ; OLIVEIRA, A. Biclustering algorithms for biological data analysis: A survey. *IEEE-ACM Transactions on Computational Biology and Bioinformatics* 1, 1 (2004), 24–45.
- [43] MAHDAVI, I. ; PAYDAR, M. M. ; SOLIMANPUR, M. ; HEIDARZADE, A. Genetic algorithm approach for solving a cell formation problem in cellular manufacturing. *Expert Syst. Appl.* 36, 3 (2009), 6598–6604.
- [44] MCCORMICK, W. T. ; SCHWEITZER, P. J. ; WHITE, T. W. Problem Decomposition and Data Reorganization by a Clustering Technique. *Operations Research* 20, 5 (1972), 993–1009.
- [45] MITROFANOV, S. P. *The Scientific Principles of Group Technology*. National Lending Library for Science and Technology, 1966.
- [46] MOSIER, C. ; TAUBE, L. The facets of group technology and their impacts on implementation — A state-of-the-art survey. *Omega* 13, 5 (1985), 381–391.
- [47] MOSIER, C. ; TAUBE, L. Weighted similarity measure heuristics for the group technology machine clustering problem. *Omega* 13, 6 (1985), 577–579.
- [48] PADBERG, M. ; RINALDI, G. Optimization of a 532-city symmetric traveling salesman problem by branch and cut. *Operations Research Letters* 6, 1 (1987), 1–7.
- [49] PAILLA, A. ; TRINDADE, A. R. ; PARADA, V. ; OCHI, L. S. A numerical comparison between simulated annealing and evolutionary approaches to the cell formation problem. *Expert Syst. Appl.* 37 (2010), 5476–5483.
- [50] PAPAIOANNOU, G. ; WILSON, J. M. The evolution of cell formation problem methodologies based on recent studies (1997-2008): Review and direction for future search. *European Journal of Operational Research* 206 (2010), 509–521.

-
- [51] PINHEIRO, R. G. S. ; MARTINS, I. C. ; PROTTI, F. ; OCHI, L. S. ; SIMONETTI, L. G. Métodos Exatos e Heurísticos para Biclusterização em Grafos. In *XVI CLAIO/XLIV SBPO*, (2012), pp. 1–12.
- [52] POTHEN, A. Graph partitioning algorithms with applications to scientific computing. *ICASE LaRC Interdisciplinary Series in Science and Engineering* 4 (1997), 323–368.
- [53] PROTTI, F. ; DA SILVA, M. ; SZWARCFITER, J. Applying Modular Decomposition to Parameterized Bicluster Editing. In *Parameterized and Exact Computation* 4169, (2006), pp. 1–12.
- [54] SEIFODDINI, H. ; WOLFE, P. M. Application of the Similarity Coefficient Method in Group Technology. *IIE Transactions* 18, 3 (1986), 271–277.
- [55] SEIFODDINI, H. Single linkage versus average linkage clustering in machine cells formation applications. *Computers and Industrial Engineering* 16, 3 (1989), 419–426.
- [56] SELIM, H. M. ; ASKIN, R. G. ; VAKHARIA, A. J. Cell formation in group technology: Review, evaluation and directions for future research. *Computers & Industrial Engineering* 34, 1 (1998), 3 – 20. – Cellular manufacturing systems: Design, Analysis and Implementation.
- [57] SOUSA FILHO, G. F. ; CABRAL, L. A. F. ; OCHI, L. S. ; PROTTI, F. Metaheuristic GRASP for Bicluster Editing Problem. In *XVI CLAIO/XLIV SBPO*, (2012), pp. 1–12.
- [58] SRINIVASAN, G. ; NARENDHAN, T. T. GRAFICS - a nonhierarchical clustering-algorithm for group technology. *International Journal of Production Research* 29, 3 (1991), 463–478.
- [59] SRINIVASAN, G. ; NARENDHAN, T. T. ; MAHADEVAN, B. Assignment model for the part-families problem in group technology. *International Journal of Production Research* 28, 1 (1990), 145–152.
- [60] STANFEL, L. E. Machine clustering for economic production. *Engineering Costs and Production Economics* 9, 1–3 (1985), 73–81.
- [61] TANAY, A. ; SHARAN, R. ; SHAMIR, R. Biclustering Algorithms: A Survey. In *Handbook of Computational Molecular Biology*, (2005), pp.
- [62] WAGHODEKAR, P. H. ; SAHU, S. Machine-component cell formation in group technology: MACE. *International Journal of Production Research* 22, 6 (1984), 937–948.

-
- [63] WOLSEY, L. A. *Integer Programming*. Wiley-Interscience, Series in Discrete Mathematics Optimization, 1998.
- [64] WU, T. H. ; CHANG, C. C. ; CHUNG, S. H. A simulated annealing algorithm for manufacturing cell formation problems. *Expert Systems with Applications* 34, 3 (2008), 1609–1617.
- [65] WU, T. H. ; CHANG, C. C. ; Y., Y. J. A hybrid heuristic algorithm adopting both Boltzmann function and mutation operator for manufacturing cell formation problems. *International Journal of Production Economics* 120, 2 (2009), 669–688.
- [66] WU, T. H. ; CHUNG, S. H. ; CHANG, C. C. A water flow-like algorithm for manufacturing cell formation problems. *European Journal of Operational Research* 205, 2 (2010), 346–360.