

Universidade Federal de Juiz de Fora
Instituto de Ciências Exatas/Faculdade de Engenharia
Programa de Pós-Graduação em Modelagem Computacional

Alexandre Cançado Cardoso

**Decomposição baseada em modelo de problemas de otimização de projeto
utilizando redução de dimensionalidade e redes complexas**

Juiz de Fora

2016

Alexandre Cançado Cardoso

**Decomposição baseada em modelo de problemas de otimização de projeto
utilizando redução de dimensionalidade e redes complexas**

Dissertação apresentada ao Programa de Pós-graduação em Modelagem Computacional, da Universidade Federal de Juiz de Fora como requisito parcial à obtenção do grau de Mestre em Modelagem Computacional.

Orientador: Prof. D.Sc. Carlos Cristiano Hasenclever
Borges

Coorientador: Prof. D.Sc. Raul Fonseca Neto

Juiz de Fora

2016

Ficha catalográfica elaborada através do programa de geração automática da Biblioteca Universitária da UFJF, com os dados fornecidos pelo(a) autor(a)

Cardoso, Alexandre Cançado.

Decomposição baseada em modelo de problemas de otimização de projeto utilizando redução de dimensionalidade e redes complexas / Alexandre Cançado Cardoso. -- 2016.

149 f. : il.

Orientador: Carlos Cristiano Hasenclever Borges

Coorientador: Raul Fonseca Neto

Dissertação (mestrado acadêmico) - Universidade Federal de Juiz de Fora, ICE/Engenharia. Programa de Pós-Graduação em Modelagem Computacional, 2016.

1. Otimização de Projeto. 2. Decomposição baseada em modelo. 3. Redução de dimensionalidade. 4. Detecção de comunidades. I. Borges, Carlos Cristiano Hasenclever, orient. II. Neto, Raul Fonseca, coorient. III. Título.

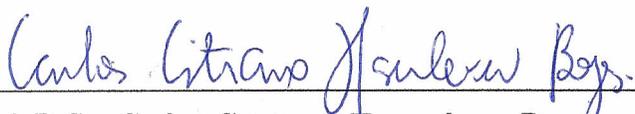
Alexandre Caçado Cardoso

Decomposição baseada em modelo de problemas de otimização de projeto
utilizando redução de dimensionalidade e redes complexas

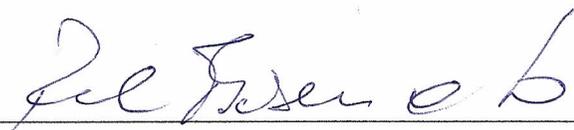
Dissertação apresentada ao Programa de Pós-graduação em Modelagem Computacional, da Universidade Federal de Juiz de Fora como requisito parcial à obtenção do grau de Mestre em Modelagem Computacional.

Aprovada em 16 de Setembro de 2016.

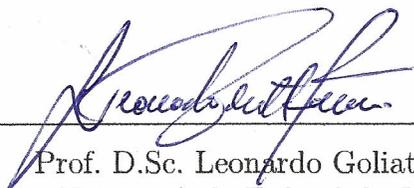
BANCA EXAMINADORA



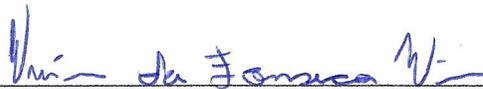
Prof. D.Sc. Carlos Cristiano Hasenclever Borges - Orientador
Universidade Federal de Juiz de Fora



Prof. D.Sc. Raul Fonseca Neto - Coorientador
Universidade Federal de Juiz de Fora



Prof. D.Sc. Leonardo Goliatt da Fonseca
Universidade Federal de Juiz de Fora



Prof. D.Sc. Vinícius da Fonseca Vieira
Universidade Federal de São João del-Rei



Prof. D.Sc. Marcelo Costa Pinto e Santos
Instituto Federal do Sudeste de Minas Gerais

*À minha esposa e
aos meus pais.*

AGRADECIMENTOS

Agradeço ao Programa de Pós-graduação em Modelagem Computacional, a Universidade Federal de Juiz de Fora, aos professores, servidores e colegas, por este período de aprendizado, compartilhamento de conhecimento e experiências, desenvolvimento acadêmico e pessoal, auxílio e companheirismo. A Guiando Telecom e a Intrinsic Soluções em Informática, assim como aos colegas de trabalho, pela compreensão e apoio em conciliar as disciplinas, a pesquisa e as atividades profissionais.

Em especial, agradeço ao Professor Carlos Cristiano Hasenclever Borges e ao Professor Raul Fonseca Neto pela orientação e ensinamentos que me auxiliaram a trilhar os caminhos do conhecimento desde os tempos do bacharelado. Aos membros da banca, Professor Leonardo Goliatt da Fonseca, Professor Vinícius da Fonseca Vieira e Professor Marcelo Costa Pinto e Santos, por aceitarem o convite e pelas valiosas contribuições que enriquecerão este trabalho.

Também, aos familiares e amigos, que me impulsionaram a enfrentar os desafios com torcida e orações. Aos meus pais, pelo apoio incondicional, dedicação e exemplo que sempre me guiaram e incentivaram. Agradeço a minha esposa, que percorreu comigo cada passo desta trajetória, auxiliando, animando, e demonstrando seu amor em todos os momentos.

Acima de tudo agradeço ao Deus todo poderoso, criador do céu, da terra, dos mares e de tudo que houve, há e haverá. Por ter estado comigo durante esta caminhada, e mais importante pela reconciliação por meio de Jesus. Que minha vida e obra sejam para Sua glória.

Soli Deo Gloria!

*”E, assim, se alguém está em
Cristo, é nova criatura; as coisas
antigas já passaram; eis que se
fizeram novas.” 2 Coríntios 5:17*

RESUMO

A estratégia de dividir para conquistar é comum a diversos ramos de atuação, indo do projeto de algoritmos à política e sociologia. Em engenharia, é utilizada, dentre outras aplicações, para auxiliar na resolução de problemas de criação de um projeto (*general design problems*) ou de um projeto ótimo (*optimal design problems*) de sistemas grandes, complexos ou multidisciplinares. O presente trabalho apresenta um método para divisão, decomposição destes problemas em sub-problemas menores a partir de informação apenas do seu modelo (*model-based decomposition*). Onde a extração dos padrões de relação entre as variáveis, funções, simulações e demais elementos do modelo é realizada através de algoritmos de aprendizado não supervisionado em duas etapas. Primeiramente, o espaço dimensional é reduzido a fim de ressaltar as relações mais significativas, e em seguida utiliza-se a técnica de detecção de comunidade oriunda da área de redes complexas ou técnicas de agrupamento para identificação dos sub-problemas. Por fim, o método é aplicado a problemas de otimização de projeto encontrados na literatura de engenharia estrutural e mecânica. Os sub-problemas obtidos são avaliados segundo critérios comparativos e qualitativos.

Palavras-chave: Otimização de Projeto. Decomposição baseada em modelo. Redução de dimensionalidade. Detecção de comunidades.

ABSTRACT

The divide and conquer strategy is common to many fields of activity, ranging from the algorithms design to politics and sociology. In engineering, it is used, among other applications, to assist in solving general design problems or optimal design problems of large, complex or multidisciplinary systems. The present work presents a method for splitting, decomposition of these problems into smaller sub-problems using only information from its model (model-based decomposition). Where the pattern extraction of relationships between variables, functions, simulations and other model elements is performed using unsupervised learning algorithms in two steps. First, the dimensional space is reduced in order to highlight the most significant relationships, and then we use the community detection technique coming from complex networks area and clustering techniques to identify the sub-problems. Finally, the method is applied to design optimization problems encountered in structural and mechanical engineering literature. The obtained sub-problems are evaluated against comparative and qualitative criteria.

Keywords: Design optimization. Model-based decomposition. Dimensionality reduction. Community detection.

SUMÁRIO

1	Introdução	18
1.1	Engenharia e Projeto	18
1.2	Otimização de Projetos em Engenharia	19
1.3	Projetos de Sistemas Complexos e Multidisciplinares	20
1.4	Modelagem	22
1.5	Objetivos	24
1.5.1	<i>Objetivos Gerais</i>	25
1.5.2	<i>Objetivos Específicos</i>	25
1.6	Estrutura do Trabalho	26
2	Decomposição de projeto.....	27
2.1	Estratégia de Divisão e Conquista	27
2.2	Decomposição de sistemas	29
2.2.1	<i>Tipos de Decomposição</i>	30
2.2.2	<i>Decomposição Baseada em Modelo</i>	34
2.2.2.1	<i>Objetivos da Decomposição Baseado em Modelo</i>	37
2.2.3	<i>Integração do Sistema Decomposto</i>	38
2.2.3.1	<i>Estratégias de Integração</i>	40
2.2.3.2	<i>Categorias dos Métodos de Integração</i>	43
2.2.3.3	<i>Ordenação Decomposição e Integração</i>	43
3	Redução de dimensionalidade	45
3.1	Variáveis Aleatórias, Vetores Aleatórios e Matriz de Dados	45
3.2	Dimensão do Espaço de Características	46
3.3	Redução de Dimensionalidade	46
3.4	Métodos	47
3.4.1	<i>Análise de Componentes Principais</i>	47
3.4.1.1	<i>Definição Matemática</i>	51
3.4.1.2	<i>Obtenção das Componentes Principais</i>	54
3.4.1.3	<i>Obtenção das Componentes Principais para Dados Padronizados</i>	58

3.4.1.4	<i>Obtenção das Componentes Principais através da Decomposição em Valor Singular</i>	60
3.4.1.5	<i>Redução de Dimensionalidade a partir das Componentes Principais</i>	63
3.4.2	<i>Decomposição em Valor Singular</i>	65
4	Análise de Agrupamento e Detecção de Comunidades	68
4.1	Análise de Agrupamento	68
4.1.1	<i>Kmeans</i>	69
4.1.1.1	<i>Variância e Distância Euclidiana ao Quadrado</i>	69
4.1.1.2	<i>Similaridade de Cosseno</i>	70
4.1.1.3	<i>Complexidade Computacional</i>	72
4.2	Redes Complexas	72
4.2.1	<i>Detecção de Comunidades</i>	73
4.2.1.1	<i>Particionamento de Grafos</i>	76
4.2.1.2	<i>Complexidade Computacional</i>	78
4.2.1.3	<i>Modularidade</i>	80
4.2.1.4	<i>Método de Louvain</i>	81
5	Metodologia	83
5.1	Representação	83
5.1.0.1	<i>Tabela de Dependência Funcional</i>	83
5.1.0.2	<i>Matriz de Ocorrência</i>	86
5.1.0.3	<i>Grafo Bipartido</i>	89
5.2	Métodos de Decomposição Utilizados	93
5.2.1	<i>Método de Referência</i>	93
5.2.2	<i>Métodos Propostos</i>	94
6	Aplicação e Experimentos	97
6.1	Projeto 1 - Modelo de Hock-Schittkowski Modificado	97
6.1.1	<i>Escolha das Dimensões do Espaço Reduzido</i>	99
6.1.2	<i>Resultados</i>	100
6.1.2.1	<i>Método Referência com Dois Agrupamentos</i>	100
6.1.2.2	<i>Método Referência com Três Agrupamentos</i>	102
6.1.2.3	<i>Método Proposto com SVD e Louvain</i>	104

6.1.2.4	<i>Método Proposto com Louvain e sem RD</i>	105
6.1.2.5	<i>Medidas de Qualidade</i>	108
6.2	Projeto 2 - Cilindro Hidráulico Mono-objetivo	109
6.2.1	<i>Escolha das Dimensões do Espaço Reduzido</i>	111
6.2.2	<i>Resultados</i>	111
6.2.2.1	<i>Método de Referência</i>	111
6.2.2.2	<i>Método Proposto com SVD e Louvain</i>	112
6.2.2.3	<i>Método Proposto com Louvain e sem RD</i>	113
6.2.2.4	<i>Medidas de Qualidade</i>	113
6.3	Projeto 3 - Cilindro Hidráulico Multi-objetivo	114
6.3.1	<i>Escolha das Dimensões do Espaço Reduzido</i>	115
6.3.2	<i>Resultados</i>	115
6.3.2.1	<i>Método de Referência</i>	115
6.3.2.2	<i>Método Proposto com SVD e Louvain</i>	117
6.3.2.3	<i>Método Proposto com Louvain e sem RD</i>	117
6.3.2.4	<i>Medidas de Qualidade</i>	118
6.4	Projeto 4 - Mola sob Tração e Compressão	119
6.4.1	<i>Escolha das Dimensões do Espaço Reduzido</i>	119
6.4.2	<i>Resultados</i>	122
6.4.2.1	<i>Método de Referência</i>	122
6.4.2.2	<i>Método Proposto com SVD e Louvain</i>	122
6.4.2.3	<i>Método Proposto com Louvain e sem RD</i>	122
6.4.2.4	<i>Medidas de Qualidade</i>	123
6.5	Projeto 5 - Redutor de Velocidade	123
6.5.1	<i>Escolha das Dimensões do Espaço Reduzido</i>	124
6.5.2	<i>Resultados</i>	125
6.5.2.1	<i>Método de Referência</i>	125
6.5.2.2	<i>Método Proposto com SVD e Louvain</i>	125
6.5.2.3	<i>Método Proposto com Louvain e sem RD</i>	127
6.5.2.4	<i>Medidas de Qualidade</i>	127
6.6	Projeto 6 - Viga Soldada	128
6.6.1	<i>Escolha das Dimensões do Espaço Reduzido</i>	129

6.6.2	<i>Resultados</i>	129
6.6.2.1	<i>Método de Referência</i>	129
6.6.2.2	<i>Método Proposto com SVD e Louvain</i>	131
6.6.2.3	<i>Método Proposto com Louvain e sem RD</i>	131
6.6.2.4	<i>Medidas de Qualidade</i>	132
6.7	Discussão	133
7	Considerações Finais	137
7.1	Trabalhos Futuros	139
	REFERÊNCIAS	141

LISTA DE ILUSTRAÇÕES

2.1	Representação de algoritmo desenvolvido pela técnica de divisão e conquista (Retirado de [1]).	28
2.2	Representação de algoritmo desenvolvido pela técnica de divisão e conquista com 3 níveis (Retirado de [1]).	29
2.3	Sistemas de tração 4x4 (Adaptado de [2]).	31
2.4	Fluxo para projeto de um sistema ótimo (Adaptado de [3]).	36
2.5	Conflito entre projeto elétrico e hidráulico residencial (Adaptado de [4]).	40
2.6	Estratégia genérica de integração hierárquica (Retirado de [5]).	41
2.7	Estratégia genérica de integração não hierárquica (Retirado de [6]).	41
3.1	<i>Scree plot</i> do PCA para base de dados Iris	64
4.1	Ângulo entre dois vetores (Retirado de [7])	71
4.2	Citações anuais dos principais artigos de Redes Complexas (Retirado de [8])	74
4.3	Exemplos de comunidades segundo definição em (a) clique, (b) forte, e (c) fraca (Retirado de [8]).	75
4.4	Quantidade de possíveis partições para um rede com N nós (Retirado de [8]).	79
4.5	Exemplo de execução do método de Louvain (Retirado de [8]).	82
5.1	Grafo bipartido para o modelo da Eq.5.1	91
6.1	Grafo bipartido para o Projeto 1	99
6.2	Projeto 1 - Projeção dos elementos do modelo no espaço reduzido bidimensional obtido pelo SVD sobre a matriz de ocorrência	101
6.3	Projeto 1 - Projeção do resultado do agrupamento com $k = 2$ no espaço reduzido bidimensional obtido pelo SVD	103
6.4	Projeto 1 - Projeção do resultado do agrupamento com $k = 3$ no espaço reduzido bidimensional obtido pelo SVD	104
6.5	Projeto 1 - Comunidades detectadas pelo método proposto com SVD(2) + Louvain	106

6.6	Projeto 1 - Projeção do resultado da detecção de comunidades no espaço reduzido bidimensional obtido pelo SVD	107
6.7	Projeto 1 - Comunidades detectadas pelo método proposto sem RD + Louvain	108
6.8	Projeto do cilindro hidráulico mono-objetivo (Retirado de [9]).	110
6.9	Projeto 2 - <i>Scree plot</i> das proporções de aproximação	112
6.10	Projeto 3 - Proporção de aproximação das p reduções de dimensionalidade através do SVD	116
6.11	Mola sob tração e compressão (Retirado de [10])	120
6.12	Projeto 4 - Proporção de aproximação das p reduções de dimensionalidade através do SVD	121
6.13	Redutor de velocidade (Retirado de [10])	125
6.14	Projeto 5 - Proporção de aproximação das p reduções de dimensionalidade através do SVD	126
6.15	Viga soldada (Retirado de [10])	129
6.16	Projeto 6 - Proporção de aproximação das p reduções de dimensionalidade através do SVD	130

LISTA DE TABELAS

5.1	FDT para o modelo da Eq.5.1	84
5.2	FDT conforme [11] para o modelo da Eq.5.1	85
5.3	Tabla de ocorrências para o modelo da Eq.5.1	88
6.1	Projeto 1 - Tabela de ocorrências	98
6.2	Projeto 1 - Proporção acumulada de aproximação das p reduções de dimensionalidade através do SVD	100
6.3	Projeto 1 - Sub-problemas decompostos pelo método de referência com SVD(2) e K-means(2)	100
6.4	Projeto 1 - Tabela de ocorrências reordenada ressaltando os agrupamentos (colorido) e as ligações (cinza), para método SVD(2) + 2-means	102
6.5	Projeto 1 - Sub-problemas decompostos pelo método de referência com SVD(3) e K-means(3)	103
6.6	Projeto 1 - Tabela de ocorrências reordenada ressaltando os agrupamentos (coloridas) e as ligações (cinza), para método SVD(2) + 3-means	103
6.7	Projeto 1 - Sub-problemas decompostos pelo método proposto com SVD(2) e Louvain	105
6.8	Projeto 1 - Tabela de ocorrências reordenada ressaltando as comunidades (coloridas) e as ligações (cinza), para método SVD(2) + Louvain	105
6.9	Projeto 1 - Sub-problemas decompostos pelo método proposto sem RD e com Louvain	106
6.10	Projeto 1 - Quantidade de variáveis de ligação e sub-problemas obtidos	109
6.11	Projeto 1 - Medidas de qualidade das comunidades/agrupamentos (a)	109
6.12	Projeto 1 - Medidas de qualidade das comunidades/agrupamentos (b)	109
6.13	Projeto 2 - Proporção acumulada de aproximação das p reduções de dimensionalidade através do SVD	111
6.14	Projeto 2 - Sub-problemas decompostos pelo método de referência com SVD(2) e K-means(2)	111
6.15	Projeto 2 - Sub-problemas decompostos pelo método proposto com SVD(2) e Louvain	112

6.16	Projeto 2 - Sub-problemas decompostos pelo método proposto sem RD e com Louvain	113
6.17	Projeto 2 - Quantidade de sub-problemas obtidos	113
6.18	Projeto 2 - Medidas de qualidade das comunidades/agrupamentos (a)	113
6.19	Projeto 2 - Medidas de qualidade das comunidades/agrupamentos (b)	114
6.20	Projeto 3 - Sub-problemas decompostos pelo método de referência com SVD(3) e K-means(3)	115
6.21	Projeto 3 - Sub-problemas decompostos pelo método proposto com SVD(3) e Louvain	117
6.22	Projeto 3 - Sub-problemas decompostos pelo método proposto sem RD e com Louvain	117
6.23	Projeto 3 - Quantidade de sub-problemas obtidos	118
6.24	Projeto 3 - Medidas de qualidade das comunidades/agrupamentos (a)	118
6.25	Projeto 3 - Medidas de qualidade das comunidades/agrupamentos (b)	119
6.26	Projeto 4 - Sub-problemas decompostos pelo método de referência com SVD(2) e K-means(2)	122
6.27	Projeto 4 - Sub-problemas decompostos pelo método proposto com SVD(2) e Louvain	122
6.28	Projeto 4 - Sub-problemas decompostos pelo método proposto sem RD e com Louvain	123
6.29	Projeto 4 - Quantidade de sub-problemas obtidos	123
6.30	Projeto 4 - Medidas de qualidade das comunidades/agrupamentos (a)	123
6.31	Projeto 4 - Medidas de qualidade das comunidades/agrupamentos (b)	124
6.32	Projeto 5 - Sub-problemas decompostos pelo método de referência com SVD(2) e K-means(2)	125
6.33	Projeto 5 - Sub-problemas decompostos pelo método proposto com SVD(2) e Louvain	126
6.34	Projeto 5 - Sub-problemas decompostos pelo método proposto sem RD e com Louvain	127
6.35	Projeto 5 - Quantidade de sub-problemas obtidos	127
6.36	Projeto 5 - Medidas de qualidade das comunidades/agrupamentos (a)	127
6.37	Projeto 5 - Medidas de qualidade das comunidades/agrupamentos (b)	128

6.38 Projeto 6 - Sub-problemas decompostos pelo método de referência com SVD(3) e K-means(3)	129
6.39 Projeto 6 - Sub-problemas decompostos pelo método proposto com SVD(3) e Louvain	131
6.40 Projeto 6 - Sub-problemas decompostos pelo método proposto sem RD e com Louvain	132
6.41 Projeto 6 - Quantidade de sub-problemas obtidos	132
6.42 Projeto 6 - Medidas de qualidade das comunidades/agrupamentos (a)	133
6.43 Projeto 6 - Medidas de qualidade das comunidades/agrupamentos (b)	133

1 Introdução

1.1 Engenharia e Projeto

A engenharia consiste no campo de estudos ”em que princípios da natureza são aplicados para construir objetos úteis” [12]. Composta, segundo [13], por atividades que incluem: análise, projeto (*design*), fabricação/construção, pesquisa e desenvolvimento de sistemas.

Por sua vez, *Design* (Projeto) é definido em [14] como o estudo da criação¹ de artefatos e da interação deles no nosso ambiente físico, psicológico, econômico, social e digital, podendo ser visto tanto como uma ciência quanto como uma arte. Seu processo construtivo envolve técnicas de análise, metodologias de desenvolvimentos e o formalismo matemático típicos das ciências, bem como ações baseadas na experiência e intuição prática do projetista.

Portanto, a atividade de projetar, ou seja, criação ou definição de um projeto (*design*), é um dos principais campos de atuação da engenharia [15]. Independentemente da especialização dentre as suas diversas disciplinas, a qualidade do projeto está intimamente ligada a qualidade do produto final obtido, seja ele um motor, prédio, *software* ou até mesmo processos, apenas para citar alguns dos mais distintos artefatos que a engenharia produz.

”Por muitas razões, não apenas o mercado competitivo, um engenheiro pode não estar interessado somente em um projeto que funcione em certo nível nominal, mas no melhor projeto de alguma forma” [12]. Assim, um projeto pode ser pensado visando obter um resultado final que satisfaça a expectativas maiores que somente um artefato pronto, atendendo expectativas que podem ir desde um produto com melhor qualidade até um de menor custo de fabricação, utilização ou impacto ambiental.

¹O termo criação é utilizado no sentido de concepção, e não como confecção, fabricação ou construção.

1.2 Otimização de Projetos em Engenharia

A otimização visa obter o melhor resultado para determinado problema de acordo com certas circunstâncias. Mais especificamente, em [16] define-se otimização em engenharia como o conjunto de decisões a serem tomadas com o objetivo final de minimizar o esforço necessário ou maximizar o benefício desejado, podendo ser aplicada nas várias etapas de qualquer sistema, seja no projeto, na construção ou na manutenção. Destaca-se que decisões assumidas nas etapas iniciais tem consequências para todas as seguintes.

Desta forma, decisões de projetos ótimas, ou pelo menos boas, afetam o resultado final do sistema construído e até mesmo sua manutenção. Caso um projeto, ou uma análise inerente, tenha sido mal feito as consequências podem até mesmo ser a inviabilidade de confecção ou manutenção do sistema.

”Frequentemente, a otimização em engenharia é feita implicitamente. Usando uma combinação de julgamento, a experiência, modelagem, opiniões dos outros, e etc, o engenheiro toma decisões de projeto que levam, ou que ele espera que levem, a um projeto ideal. Alguns engenheiros são muito bons nisso. No entanto, se houver muitas variáveis a serem ajustadas com vários objetivos conflitantes e/ou restrições, este tipo de otimização baseada na experiência pode ficar aquém de identificar o projeto ideal. As interações são muito complexas e as variáveis demasiadamente numerosas para determinar intuitivamente o projeto ideal.” [12]

Métodos formais para a realização de otimizações foram desenvolvidos já nos primórdios do cálculo diferencial na época de Newton, Lagrange, Cauchy, Bernoulli, Euler, Lagrange, Weirstrass, dentre outros grandes cientistas. Tanto que diversos métodos clássicos em otimização são conhecidos pelo nomes destes seus ilustres criadores. Após um período de arrefecimento voltaram a tona durante a Segunda Guerra Mundial com a utilização de técnicas de programação matemática (*mathematical programming techniques*) e pesquisa operacional (*operational research*).

Nos dias atuais novos métodos, estratégias e técnicas continuam sendo desenvolvidas, assim como suas aplicações. Indo além dos limites do cálculo tradicional, utilizando por exemplo, de métodos simbólicos e dos mais diversos algoritmos e meta-heurísticas (*metaheuristics*) com inspirações em fenômenos físicos, químicos, biológicas e sociais. Tamanha evolução, seja nas técnicas ou na aplicação em problemas reais só tem sido possível por conta do avanço da computação, dando origem assim a chamada otimização

de projeto assistida por computador (*computer-aided design optimization* - CADO)[13].

1.3 Projetos de Sistemas Complexos e Multidisciplinares

Nas últimas décadas, os projetos de engenharia também foram se tornando cada vez mais complexos, confirmando a previsão de Stephen Hawking de “que o próximo século será o século da complexidade” [8]. Este cenário, constantemente, ocasiona o desenvolvimento de novos métodos e tecnologias, bem como propulsionadas por novas condições, as quais permitem ou requerem que sistemas cada vez maiores e mais complexos sejam projetados.

Segundo [17] uma das características mais aceitas para o mundo de hoje é a sua “complexidade”. Esta complexidade não é nova, mas há uma ênfase atualmente em lidar com seus aspectos diretamente, motivada pela crescente capacidade de entender aspectos relativos a complexidade do mundo físico, em grande parte devido ao rápido crescimento da tecnologia de computação e informação. Impulsionados tanto por fatores de negócios quanto tecnológicos os problemas de projeto são cada vez mais visto como sistemas complexos, ao invés de componentes isolados.

Estes novos desafios para a criação de projetos em engenharia geraram, conseqüente, uma área de pesquisas chamada de projeto de sistemas complexos em engenharia (*design of complex engineering systems*) [18] [16]. Papalambros e Michelena, em [17], definem Sistemas como “uma coleção de entidades que podem ser propriamente estruturadas tal que alguma forma de decomposição é possível (...) em contraste com o termo componente”. Ao invés de outra definição baseada em objeto que forneça a uma certa entrada uma saída determinada.²

A complexidade destes sistemas é ampliada pela necessidade de integrar as múltiplas disciplinas envolvidas no projeto a ser criado. Neste contexto se fala em otimização de projetos multidisciplinares (*multi-disciplinary design optimization* - MDO) ou otimização de projetos de sistemas multi-disciplinares (*multidisciplinary system design optimization* - MSDO) [14] [19] [13]. Neste caso, não apenas a natureza multidisciplinar do problema se torna um obstáculo, mas inclusive a integração das diferentes disciplinas e equipes

²Este trabalho utiliza a primeira definição.

interdisciplinares, muitas vezes envolvendo áreas que vão além das ciências de tecnologia e engenharias, chegando a agregar membros oriundos de campos das humanas, saúde e dos mais diversos ramos do conhecimento formal ou prático.

Portanto, como pontuado em [20]: "a dificuldade de solucionar problemas de otimização em larga escala e problemas de otimização multidisciplinar (MDO), (...) tem motivado várias abordagens de decomposição." Os dois principais atributos que tornam esta estratégia atrativa, conforme [21], são:

1. Simplificação: os problemas menores são geralmente mais simples de solucionar que problemas de maior dimensão;
2. Velocidade: soluções para os sub-problemas podem ser derivadas em paralelo e, além disto, o problema completo nem sempre é escalável.

Estas vantagens podem ser obtidas tanto para processos computacionais quanto humanos. No caso da simplificação, problemas menores geralmente demandam um custo computacional menor, conseguem ser solucionados por métodos e algoritmos mais simples (ou conseguem ser solucionados enquanto o problema original complexo nem consegue ser resolvido), apresenta, ainda, a possibilidade dos sub-problemas serem resolvidos por métodos distintos. Além disto, para os projetistas, a simplificação torna os sub-problemas mais compreensíveis, permitindo que o especialista entenda o problema de definição do projeto ou do projeto ótimo e consiga contribuir validando, inserindo seu conhecimento prático, tomando melhores decisões quanto ao processo computacional ou, até mesmo, solucionando o sub-problema.

A velocidade ou agilidade na obtenção dos resultados pode ser uma consequência da simplificação, tanto do ponto de vista humano quanto computacional, mas também pode ser obtida devido a divisão, possibilitando que os sub-problemas sejam solucionados em paralelo por equipes distintas e utilizando técnicas de programação distribuída. Permite-se, assim, a execução do projeto dentro de prazos e custos viáveis enquanto considerando-se o sistema completo, até mesmo uma inviabilização de sua realização pode ocorrer.

Complementando, em [5] afirma-se que: "decomposição de problemas de projeto de sistemas em larga-escala permite que se obtenha: simplificação conceitual do sistema, redução na dimensão do problema, procedimentos computacionais mais eficientes, diferentes técnicas de resolução para sub-problemas individuais, projeto simultâneo,

modularização, análise multi-objetiva, e eficiência na comunicação e gestão entre os diversos grupos envolvidos no processo de desenho do projeto.”

Entretanto, também é ressaltado em [21] os dois principais desafios associados a esta estratégia de criação projetos:

1. Decomposição: descobrir o conjunto mais adequado de sub-problemas;
2. Integração: combinar os sub-sistemas separados em uma solução global.

Tanto a decomposição do projeto (*design decomposition*) quanto a integração (*integration*), ou coordenação (*coordination*), podem se apresentar como tarefas difíceis. Portanto, o estudo de métodos para decomposição e para integração são bastante relevantes, visando que o processo de divisão e unificação de um problema em sub-problemas seja interessante tanto na definição do projeto ou do projeto ótimo em uma aplicação real em engenharia.

A criação de métodos para estas duas tarefas são tradicionalmente estudadas separadamente na literatura, conforme explica-se na Seção 2.2.3. Da mesma forma o presente trabalho tem como foco exclusivamente a decomposição do projeto, também chamada de particionamento de projeto (*design partitioning*).

1.4 Modelagem

A criação de um projeto em engenharia consiste em encontrar os valores para variáveis de projeto (*design variables*) que atenda as restrições de projeto (*design constraints*). Caso se deseje um projeto ótimo, os valores definidos para as variáveis de projeto ainda devem buscar que o objetivo ou objetivos definidos sejam otimizados.

Por exemplo, deseja-se projetar uma embalagem onde deve-se definir as dimensões da largura, altura, comprimento e espessura (variáveis de projeto), de forma que a embalagem resista a um peso de no mínimo 5% maior que o peso do produto, que é 20kg (restrição de desigualdade). Além disto, deve ter um volume exato de 10,5 litros (restrição de igualdade). Considerando que o problema seja de otimização do custo, o projeto obtido pela definição da dimensões da embalagem, ainda, deve consumir a menor quantidade possível de material (objetivo).

Entretanto, definições como a do exemplo descrito não são adequadas para a utilização de métodos matemáticos ou computacionais. Para isto, são necessários os problemas sejam modelados em uma estrutura formal. Michelena e Papalambros [11] apresentam as formulações clássicas tanto para o problema de projeto geral (*general design problem* - GDP), Eq. 1.1, quanto para o problema de projeto ótimo (*optimal design problem* - ODP), Eq. 1.2.

$$\begin{aligned} \text{encontre } \mathbf{x} &\in X^m \\ \text{tal que } \mathbf{h}(\mathbf{x}) &= \mathbf{0} \\ \mathbf{g}(\mathbf{x}) &\leq \mathbf{0} \end{aligned} \quad (1.1)$$

onde, x é o vetor com variáveis de projeto, h é a forma vetorial das restrições de igualdade do projeto (*equality design constraints*), g é a forma vetorial das restrições de desigualdade do projeto (*inequality design constraints*), e X^m representa as restrições de limite das variáveis de projeto no espaço de dimensão m .

A diferença do problema geral para o ótimo é que, não basta apenas definir um projeto factível, devendo, ainda, otimizar o objetivo de projeto (*design objective*), definido pela função objetivo do projeto, f . Na Eq. 1.2 apresenta-se a formulação do ODP:

$$\begin{aligned} \text{encontre } \mathbf{x} &\in X^m \\ \text{tal que } \mathbf{h}(\mathbf{x}) &= \mathbf{0} \\ \mathbf{g}(\mathbf{x}) &\leq \mathbf{0} \\ e \quad f(\mathbf{x}) &\text{ seja mínimo} \end{aligned} \quad (1.2)$$

Reescrevendo a Eq. 1.2 do ODP no formato padrão de um problema de otimização obtém-se a Eq. 1.3, abaixo:

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}) \\ \text{tal que } \quad & \mathbf{h}(\mathbf{x}) = \mathbf{0} \\ & \mathbf{g}(\mathbf{x}) \leq \mathbf{0} \\ & \mathbf{x} \in X^m \end{aligned} \quad (1.3)$$

Para o exemplo da embalagem, o modelo de GDP seria:

$$\begin{aligned} \text{encontre } \quad & x_1, x_2, x_3, x_4 \in \Re \\ \text{tal que } \quad & h_1(x_1, x_2, x_3, x_4) = Volume(x_1, x_2, x_3, x_4) - 10,5l = 0 \\ & g_1(x_1, x_2, x_3, x_4) = 1,05 * 20kg - Resistência(x_1, x_2, x_3, x_4) \leq 0 \end{aligned} \quad (1.4)$$

O modelo de otimização para o ODP fica:

$$\begin{aligned}
 \min_{x_1, x_2, x_3, x_4} \quad & f(x_1, x_2, x_3, x_4) = \text{Material}(x_1, x_2, x_3, x_4) \\
 \text{tal que} \quad & h_1(x_1, x_2, x_3, x_4) = \text{Volume}(x_1, x_2, x_3, x_4) - 10,5l = 0 \\
 & g_1(x_1, x_2, x_3, x_4) = 1,05 * 20kg - \text{Resistência}(x_1, x_2, x_3, x_4) \leq 0 \\
 & x_1, x_2, x_3, x_4 \in \mathfrak{R}
 \end{aligned} \tag{1.5}$$

No presente trabalho, apesar do formalismo utilizado para expressar os modelos de GDP e ODP provirem da programação matemática, não é exigido que o modelo seja composto apenas por elementos analíticos, ou seja, matemáticos.

Isto se deve à decomposição dos problemas considerar o modelo de forma simbólica [22]. Portanto, não há qualquer obstáculo para decomposição caso o modelo do GDP ou do ODP sejam compostos por variáveis e constantes categóricas, booleanas, etc; ou por restrições e objetivos que sejam simulações, sentenças lógicas, superfícies de resposta (*response surface*) [11][23], algoritmos, podendo até mesmo serem avaliações realizadas por humanos, tanto de forma objetiva quanto subjetivamente. Também, permite-se que elementos de múltiplas natureza estejam presentes no mesmo modelo.

Inclusive, uma das grandes vantagens da decomposição é exatamente ser aplicada a modelos de sistemas multidisciplinares, permitindo assim que se modularize os componentes em grupos similares o que viabiliza que cada sub-problema seja resolvido pela técnica mais adequada, enquanto que para o problema original pode, inclusive, ser impossível encontrar um método de solução que se aplique a problemas compostos por naturezas distintas.

1.5 Objetivos

O principal objetivo deste trabalho é apresentar um modelo de decomposição de sistemas formulado através estratégias de redução de dimensionalidade em conjunto com técnicas advindas de redes complexas, utilizando características intrínsecas destes métodos com a finalidade de se obter um modelo eficiente e competitivo.

1.5.1 Objetivos Gerais

Para que este objetivo seja alcançado, deve-se realizar a decomposição de sistemas de engenharia de modo que o problema de se obter um projeto factível ou um projeto ótimo para o artefato sejam sub-divididos em sub-problemas menores e mais simples, através das relações explícitas e implícitas expressas exclusivamente a partir do modelo do sistema, de forma que elementos com forte interação sejam alocados para os mesmos sub-problemas e os elementos com pouca ou nenhuma interação sejam tratados como sub-projetos separados. Utilizando uma abordagem semântica de interpretação do modelo, algoritmos de redução de dimensionalidade para evidenciar as relações relevantes e eliminar as que não contribuem para o processo, e identificar os elementos de cada sub-problema de projeto por um método de detecção de comunidades provindo da área de Redes Complexas, tal que os sub-sistemas resultantes sejam considerados adequados perante a medidas de qualidade e comparativamente com um método de referência e resultados da literatura, quando disponíveis.

1.5.2 Objetivos Específicos

- Extrair por uma estratégia simbólica os relacionamentos das entidades do modelo do problema de projeto;
- Obter uma representação para o modelo que permita a aplicação de redução de dimensionalidade e da detecção de comunidades;
- Validar a possibilidade de uso da tarefa de detecção de comunidade ao invés de análise de agrupamento;
- Construir um *workflow* eficiente para a tarefa de decomposição de sistemas complexos;
- Validar este novo paradigma construtivo para decomposição de sistemas de engenharia

1.6 Estrutura do Trabalho

Neste primeiro capítulo, Introdução, foi apresentada a área de projeto (*design*) em engenharia, tanto gerais quanto ótimos, assim como as de sistemas complexos e multidisciplinares. Também se abordou o cenário atual de sistemas cada vez mais complexos, grandes e multidisciplinares ressaltando a necessidade da decomposição destes para a realização da tarefa de definição do projeto em sub-problemas menores e mais simples, de forma a justificar o presente trabalho. Também, ainda neste capítulo, apresentou-se a modelagem clássica dos GDPs e dos ODPs, e por último os objetivos gerais e específicos foram definidos.

O segundo capítulo, Decomposição de projeto, trata sobre a decomposição de projeto, abordando a inspiração na estratégia de divisão e conquista, a decomposição de sistemas complexos, seus tipos, com ênfase especial no utilizado neste trabalho, a decomposição de projeto baseada em modelo. Também trata sobre a área de integração de problemas de projeto decompostos e como estes dois campos de estudo e pesquisa interagem entre si. Por sua vez, no capítulo terceiro, Redução de dimensionalidade, é realizada a apresentação dos fundamentos estatísticos da área de redução de dimensionalidade assim como dos métodos utilizados neste presente trabalho. Para finalizar os capítulos de contextualizações, o capítulo quarto, Análise de agrupamento e Detecção de comunidades, explica as tarefas de análise de agrupamento e de detecção de comunidades em redes complexas, bem como uma introdução sobre o que são redes complexas. Este capítulo, também contém seções sobre o algoritmo K-means e método de Louvain utilizados para identificação dos sub-sistemas.

No quinto capítulo, Metodologia, são expostas tanto as representações para o modelo quanto os métodos de decomposição utilizados, tanto os propostos quanto o de referência proveniente da literatura. Estes métodos são aplicados e testados em problemas retirados da literatura no sexto capítulo, Aplicação e Experimentos, e os resultados apresentados, que incluem além dos problemas decompostos, também medidas de avaliação qualitativas e comparativas. Por fim o capítulo sétimo, Considerações finais, contém a discussão dos resultados do trabalho e a sugestão de trabalhos futuros.

2 Decomposição de projeto

Conforme descrito em [17], o projeto de artefatos e coleções de artefatos complexos em engenharia geralmente requer o uso de simulações acopladas de técnicas de otimização. Como a complexidade aumenta, a capacidade de empregar a intuição (mesmo que para compreender os *trade-offs* computacionais) diminui rapidamente. Além disto, a capacidade humana de realmente resolver estes problemas de otimização do sistema torna-se limitada com o aumento da dimensionalidade.

Pimmler e Eppinger [21] afirmam ser uma prática comum entre os engenheiros a de resolver problemas complexos inicialmente subdividindo-os em um conjunto de problemas menores tratados mais facilmente. Seguindo, desta forma, os conselhos de Henry Ford: "Nada é particularmente difícil se você dividir em pequenos trabalhos"[24] [25]. E de René Descartes: "Divida cada uma das dificuldades (...) tão quanto necessário para sua solução adequada." [26]

2.1 Estratégia de Divisão e Conquista

A estratégia de dividir para conquistar, ou dividir e conquistar (originalmente em latim: *divide et impera*), é utilizada desde a antiguidade com o objetivo de controlar um território ou nação fragmentando as concentrações de poder político, militar, e a identidade cultural. Por mais, que se tenha registros de sua utilização por diversos impérios, o primeiro relato desta estratégia como um método formal de guerra e dominação é atribuído a Júlio César no livro *Commentarii de Bello Gallico* ("Comentários sobre a Guerra da Gália") [27] e, posteriormente, difundida por manuais como *Dell'arte della guerra* ("A arte da guerra") de Maquiavel [28]. Atualmente a política continua a encontrar novas utilidades além do campo militar como, por exemplo, em aplicações no *marketing* político [29] [30] [31].

A mesma estratégia, também é aplicada em outros ramos do conhecimento, com destaque em projetos de algoritmos em ciência da computação, onde é nomeada de "divisão e conquista" (*divide and conquer*). Consiste em fracionar o problemas em sub-problemas menores que serão resolvidos recursivamente (se necessário um sub-problema

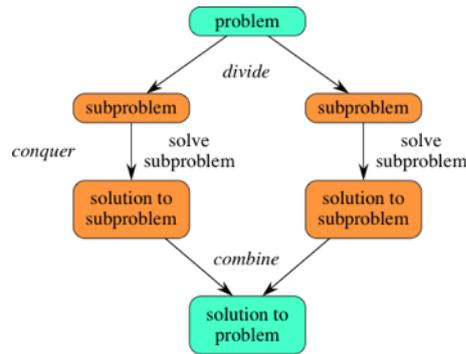


Figura 2.1: Representação de algoritmo desenvolvido pela técnica de divisão e conquista (Retirado de [1]).

deve ser fragmentado em problemas menores e assim sucessivamente), com as soluções dos sub-problemas combinadas de maneira adequada se obtém a solução do problema original. Os algoritmos desenvolvidos por esta técnica consistem em três passos descritos a seguir, conforme apresentado em [1]:

1. Dividir o problema em um número de sub-problemas que são instâncias menores do mesmo problema;
2. Conquistar os sub-problemas dividindo-os recursivamente ou, se forem pequenos o suficiente, resolvendo o caso base;
3. Combinar as soluções dos sub-problemas na solução para o problema original.

Segundo [32], o segredo do sucesso da estratégia é o processo de divisão (Passo 1) e o processo da combinação, ou conquista (Passo 2), serem de baixo custo, considerando que a resolução do caso básico seja significativamente menos custosa que a do problema completo. Permite-se, assim, obter a solução de forma mais fácil que a resolução direta do problema completo. Ressalta-se, ainda, que a resolução de alguns problemas completos só são possíveis ou viáveis por meio da divisão em sub-problemas mais simples.

Assumindo um caso em que cada divisão inicial produza dois sub-problemas tem-se o diagrama da Fig. 2.1. Sendo necessário, cada sub-problema é novamente dividido, obtendo, por exemplo, o caso representado na Fig. 2.2. Vale ressaltar, que não é obrigatório que todos os sub-problemas se dividam a mesma quantidade de vezes, tanto quanto a profundidade quanto a largura.

A técnica da divisão e conquista produz algoritmos eficientes para muitos problemas [32], como, por exemplo: Busca binária, *Merge sort*, *Quick sort*, algoritmo de Karatsuba e

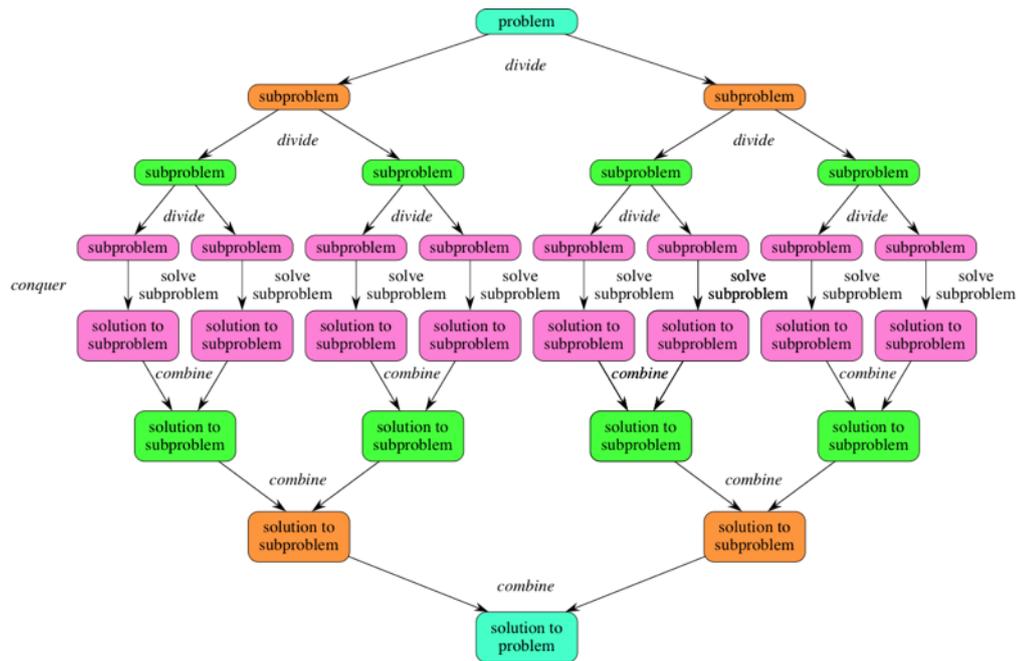


Figura 2.2: Representação de algoritmo desenvolvido pela técnica de divisão e conquista com 3 níveis (Retirado de [1]).

Ofman (para multiplicação), diversos *parsers* para linguagens de programação ou naturais, e a tradicional solução para o problema Torres de Hanói.

Além da política e computação, esta estratégia também tem sido aplicada atualmente a outras áreas como ao *marketing* de produtos [33], mercado imobiliário [34], direito e economia [31]. Na engenharia, além da área de desenvolvimento de *softwares*, alguns exemplos são nas áreas de cadeia de suprimento químicos [35], sistemas químicos quânticos em nanomateriais e bio-estruturas [26] e na decomposição de projeto (*decomposed-based general design*) ou projeto ótimo (*decomposed-based design optimization*), que é o caso deste trabalho.¹

2.2 Decomposição de sistemas

Como apresentado no Cap. 1, a cada dia os sistemas se tornam mais complexos, multidisciplinares e maiores. Consequentemente projetá-los se torna um mais desafiador. Portanto, como colocado em [17]: "uma abordagem para resolver estes problemas difíceis é tentar quebrar o problema em menores, tal que possam ser resolvidos mais facilmente, e

¹Os métodos de decomposição e integração (Seção 2.2.3) de projeto não são algoritmos projetados pelo paradigma de divisão e conquista, mas técnicas para o projeto de sistemas que também utilizam-se da mesma inspiração, a estratégia *divide et impera*.

compor a solução global do sistema a partir da solução de de suas partes”. Isto é, aplicar a estratégia de dividir para conquistar na resolução de um GDP ou um ODP.

A ”quebra” ou divisão em problemas menores é chamada de decomposição ou partição², enquanto a composição ou conquista da solução para o problema completo original a partir das soluções dos problemas decompostos, sub-problemas, é chamada de integração (*integration*) ou coordenação (*coordination*)³.

2.2.1 Tipos de Decomposição

De acordo com [17], os métodos de decomposição são classificados em:

- Decomposição por objeto (*object decomposition*) - decompõe pelos componentes físicos;
- Decomposição por aspecto (*aspect decomposition*) - decompõe pelos domínios de conhecimento, motivação original da MDO;
- Decomposição sequencial (*sequential decomposition*) - decompõe pela direção do fluxo de elementos ou informação;
- Decomposição baseada no modelo (*model-based decomposition*) - decompõe a partir do informações extraídas exclusivamente do modelo.

A decomposição por objeto e a decomposição por aspecto, também chamadas respectivamente de decomposição baseada em objeto (*object-based decomposition*) e decomposição baseada em aspecto (*aspect-based decomposition*) [14], assumem uma decomposição ”natural” do problema. Entretanto, desenhar fronteiras entre os componentes físicos e suas interações é subjetivo, enquanto as divisões pelos domínios de conhecimento podem ser ditadas por considerações administrativas que falham em mensurar o acoplamento das disciplinas.

²Os trabalhos iniciais utilizam principalmente o termo partição [36] relacionando ao particionamento de grafo. Mas também apresentam o termo decomposição hora como sinônimo de partição, hora incluindo tanto a partição quanto a integração. Entretanto, alguns trabalhos mais recentes como [9] não utilizam o termo particionamento, e consideram decomposição apenas à tarefa de divisão. Ainda, o fato deste trabalho envolver detecção de comunidades em redes complexas (intimamente relacionada, mas diferente, a área de particionamento de grafos [8]) contribuiu para a preferência de decomposição para nomear exclusivamente a divisão, objetivando evitar ambiguidades.

³Na literatura o termo mais frequente é *coordination*, entretanto na tradução para o português considerou-se preferível o termo integração.

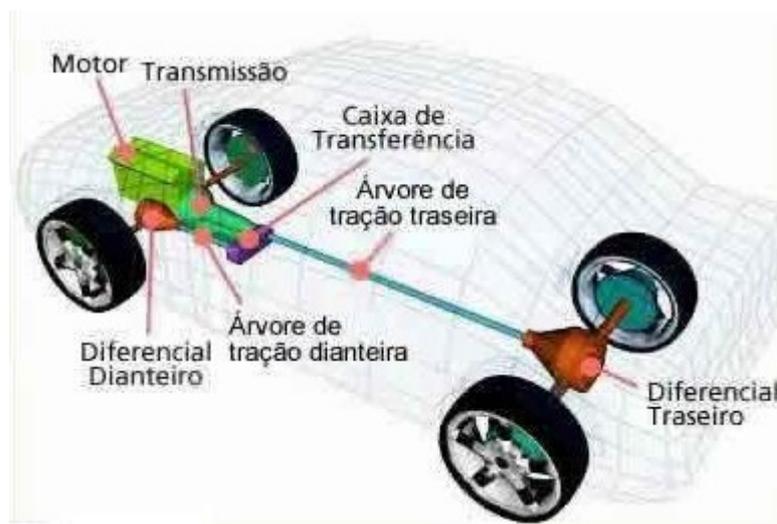


Figura 2.3: Sistemas de tração 4x4 (Adaptado de [2]).

Por sua vez, a decomposição sequencial presume uma direção única no projeto do fluxo de informação que pode contradizer o comportamento cooperativo desejado em um projeto de engenharia simultâneo. Finalmente, a decomposição baseada em modelo utiliza a representação funcional matemática, os objetivos e restrições de projeto expressa(s) no modelo(s) para identificar estruturas implícitas que sejam desconexas (não relacionadas) ou fracamente conexas dentro do modelo. Mas esta limitada informações de projeto incluídas ao modelo, seja de forma implícita, explícita, intencional ou não.

Para ilustrar os tipos de decomposição, considere que se deseja dividir o projeto do sistema de tração quatro por quatro de um carro elétrico da Fig. 2.3.⁴

Realizando uma decomposição por objeto, o sistema complexo será dividir em cada um de seus componentes físicos, isto é objetos. Portanto, um possível solução seria solucionar os sub-problemas de projetar o motor, a transmissão, cada um dos diferenciais, cada uma das árvore de tração, a caixa de transferência, cada um dos eixos, as rodas dianteiras e as rodas traseiras. Observe que no caso, foi decidido não separar as rodas esquerdas das direitas, ou seja, foram considerados objetos similares e que não seria necessário uma outra subdivisão. Por fim, as tarefas de projetar cada sub-problema é entregue a equipes de engenheiros especialistas em cada componente, podendo mais de um projeto ser alocado para uma mesma equipe, mas serem abordados como problemas distintos. No exemplo, seria o cenário onde ambos os eixos fossem ser projetados pela mesma equipe, mas que

⁴O exemplo é apenas para ilustrar os tipos de decomposição, não foi baseado em decomposições reais, e considera os componentes de um ponto de vista intuitivo e generalista, externo as peculiaridades de projeto mecânico especializado de um automóvel.

cada um fosse analisado como um projeto a parte.

Se a mesma decomposição fosse realizada por aspecto, o sistema seria segmentado conforme os domínios de conhecimento. Uma possível solução seria decompor o sistemas em dois sub-problemas, o projeto do motor elétrico e/ou projeto das demais partes mecânicas. Alocando o sub-projeto do motor elétrico para um equipe composta por engenheiros eletricitas e mecatrônicos, enquanto as partes restantes são consideradas como um sistema único a ser alocado para uma equipe de engenheiros mecânicos.

Entretanto, considerou-se que o sistema envolvendo todas as partes mecânicas ainda seria muito complexo para ser projetado diretamente. Então, decidiu-se decompor este sub-sistema novamente baseando-se nos aspectos. Observa-se que também seria possível que esta segunda decomposição fosse realizada por objeto, segundo [14] esta forma de divisão, chamada de decomposição baseada em equipe (*team-based decomposition*), também é comum. Sabendo que os engenheiros mecânicos da empresa formavam cinco grupos de especialização: em diferenciais, em árvores de tração, em eixos, em rodas e um último em transmissão. Decidiu-se que cada uma destas equipes seria responsável pelo sub-projeto mecânico de sua especialidade.

Por fim, a equipe responsável pelos diferenciais ainda considerou que o aspecto de projeto do diferencial dianteiro e traseiro eram significativamente distintos, e portanto, internamente decompôs novamente esta parte do projeto em dois sub-problemas. Apesar do resultado ter sido similar ao obtido pela decomposição baseada em objeto, percebe-se que o processo construtivo foi diferente. Vale ressaltar que não é obrigatório que os sub-projetos sejam solucionados em paralelo, todos os sub-projetos poderiam ser desenhados pela mesma equipe multidisciplinar de engenheiros, mas a decomposição dos problemas (independente do tipo) ditariam como estes projetos seriam abordados. Ou seja, cada sub-problema seria desenvolvido como um projeto independente (mesmo que pela mesma equipe) e posteriormente conquistados em um projeto único na integração.

A terceira forma de decomposição é a sequencial. Neste caso, o sistema de tração seria projetado seguindo o fluxo dos elementos ou da informação. Considerando-se o fluxo da produção da força mecânica no motor até a aplicação desta nas rodas provocando o deslocamento do carro. Assim, uma possível decomposição seria primeiramente projetar o motor, depois a transmissão, seguida das árvores de tração, então os diferenciais, chegando aos eixos e por fim as rodas. Sendo que cada projeto poderia ser executado por uma

equipe distinta, e entregue a próxima equipe para o projeto da parte subsequente, ou ser desenvolvido pela mesma equipe.

Por fim, o último tipo de decomposição é a baseada em modelo. Inicialmente os engenheiros constroem o modelo para o sistema de tração como um todo, definindo quais seriam as variáveis de projeto que se deseja descobrir, as restrições e, caso seja um ODP, a função objetivo ou as funções objetivo (por exemplo, custo do veículo e potência final do automóvel). Então, métodos de decomposição como o do presente trabalho são aplicados através de informações explícitas e implícitas nos elementos do modelo visando identificar os relacionamentos destes elementos e definir sub-modelos a serem tratados separadamente. Em seguida se define a estratégia de integração dos sub-problemas. Finalmente, eles são resolvidos e as soluções combinadas de acordo com a estratégia de integração escolhida.

Evidentemente, a tarefa de criação do modelo para o sistema completo também é um problema complexo. Entretanto, considerando-se o modelo de um sistema como um artefato a ser projetado, pode-se aplicar as mesmas técnicas aqui descritas para a decomposição deste problema em sub-problemas de definição de sub-modelos, os quais, por sua vez, seriam integrados no modelo completo para o sistema de tração. Este modelo completo seria a entrada para o segundo processo de decomposição e integração que visa definir não o modelo, mas os valores das variáveis de projeto deste modelo, permitindo inclusive que o modelo seja validado, redundâncias e conflitos identificados, e até mesmo surgimento de ideias novas provindas de uma melhor compreensão do problema.

É importante, também, lembrar que para a resolução dos componentes, provavelmente modelos formais serão definidos, principalmente se for desejado a obtenção de um projeto ótimo para o componente, como já é comum no estudo de otimização de projeto (*design optimization*). Sendo assim, a decomposição baseada em modelo não implicaria na tarefa extra de criar o modelo, utilizando apenas estes modelos que já serão criados para tentar obter melhores decomposições.

Nota-se que nas decomposições os sub-projetos foram alocados para diferentes equipes mas nada foi apresentado sobre o método de resolução de cada projeto. Isto foi intencional, pois para os métodos de decomposição independe a estratégia de resolução e até mesmo como será feita a conquista da solução combinada. A forma de resolução de cada sub-problema é definida para cada um deles separadamente, podendo em cada um se utilizar

a estratégia mais adequada, ao invés de uma forma única, que seria o caso do problema completo. Por sua vez, a unificação dos sub-problemas é uma tarefa estudada pelos trabalhos em integração de problemas de projeto decompostos, conforme será delineado na Seção 2.2.3.

Para o problema de projeto do modelo do sistema de tração completo, é natural pensar em utilizar as decomposições baseadas em objeto e aspecto. Também, é possível perceber que a sequencial se aplicaria. Entretanto, pode se concluir que a baseada em modelo só seria aplicável ao problema matematicamente enunciado. Mas isto não é verdade, a maioria dos métodos de decomposição baseada em modelo, inclusive o apresentado por este trabalho, não obrigam que o modelo utilizado seja um modelo analítico de programação matemática (por mais que este seja o caso mais comum). Também, é possível aplicar a mesma metodologia a modelos em alto nível que descrevam apenas a existência ou não de relação entre os componentes do sistema, não sendo necessário de expressar analiticamente estes relacionamentos.

Um grande diferencial da decomposição baseada em modelo decorre de não se fundamentar nos atributos naturais do problema (seja de composição, área de conhecimento ou fluxo), mas na interação entre os elementos modelados (seja esta interação expressa de forma matemática ou não). Isto possibilita a identificação de arranjos inovadores, que não seriam obtidos nos demais tipos de decomposição. Por mais, que outros tipos tenham um apelo mais intuitivo, tornam-se limitados e influenciados por divisões já estabelecidas. Contrapondo a isto, ao se considerar apenas informações do modelo, o processo de decomposição pode conseguir identificar características inovadoras no projeto [37], levando a melhores decomposições tanto para resolução dos sub-problemas quanto para a auxiliar na compreensão que os projetistas tem sobre o sistema, o que é especialmente interessante em sistemas grandes, complexos e multidisciplinares.

2.2.2 Decomposição Baseada em Modelo

Segundo descrito em [6] e [38], a decomposição baseada em modelo (*model-based decomposition*) permite identificar estruturas fracamente conectadas que estejam implícitas no modelo matemático, ou não, do projeto que possam satisfazer requisitos de projeto de alto nível, tais como concorrência, modularização⁵ robustez e simplificação do

⁵De acordo com Baldwin e Clark [18] a modularização é um dos principais motivos para o grande avanço na área de computação tanto no âmbito de *softwares* quanto de *hardwares*, citando como exemplo só

mesmo. Também, permite melhor aproveitamento de recursos computacionais, através de procedimentos algorítmicos mais simples e eficientes, computação paralela e distribuída, e a resolução de cada sub-sistemas por técnicas individuais mais adequada.

Evidentemente, para se dividir um projeto baseado em seu modelo, é necessário primeiramente definir o modelo. Independente do formalismo utilizado o modelo do sistema pode ser compreendido em problemas de duas naturezas: otimizar o projeto sistemas, ou simplesmente projetar o sistema (alguns sistemas são tão complexos, grandes e envolvem tantas disciplinas que projetá-los, mesmo que de forma não ótima, já não é uma tarefa simples e vale ser decomposta). No caso do problema de otimização tem-se um ODP e no caso geral um GDP, conforme definido na Seção 1.4.

Com o o problema de projeto do sistema em mãos, pode-se iniciar o processo de decomposição. O resultado da decomposição são os problemas de projeto dos sub-sistemas. Feito a divisão chega o momento de conquistar, realizado pela integração. E, finalmente, o projeto ou projeto ótimo do sistema completo é obtido.

A Fig. 2.4 apresenta um fluxo tradicional para decomposição (ou particionamento, *partitioning*) e integração (ou coordenação, *coordination*) com o objetivo de projetar um sistema ótimo.

Note que a Fig. 2.4 apresenta dois caminhos distintos para obtenção de um sistema ótimo. O fluxo da esquerda, chamado de análise, proposto por Wagner [39] *apud* [3] [40] sob orientação de Papalambros, consiste em definir o objetivo ou os objetivos da otimização antes da decomposição, que será realizada através de um ODP. Por sua vez, o fluxo da direita, chamado de síntese, proposto por Krishnamachari e Papalambros [41] [42], decompõe o sistema a partir do GDP, ou seja sem que o objetivo ou os objetivos tenham sido selecionados, o que só é realizado após a decomposição e antes da integração.

Como na análise a decomposição também tem acesso à função objetivo, esta poderá ser subdividida⁶ entre os diversos sub-sistemas e solucionada separadamente, ou seja avaliada pela decomposição (*decomposition analysis*). Por sua vez, na síntese⁷, o objetivo do

circuito integrados. Pois torna a complexidade administrável, permite trabalho em paralelo e acomodar incertezas futuras. Ainda, afirmam que outras indústrias também podem necessitar, tendo que se adaptar a modularização, visto ser uma suas implicações de valor e custo serem uma grande força financeira para o contexto presente.

⁶Análise do grego *análusis, eōs* significa dissolução ou separação de um todo em seus elementos ou partes componentes.

⁷Síntese do latim *synthēsis, is* significa coleção ou reunir elementos diferentes e fundi-los num todo coerente.

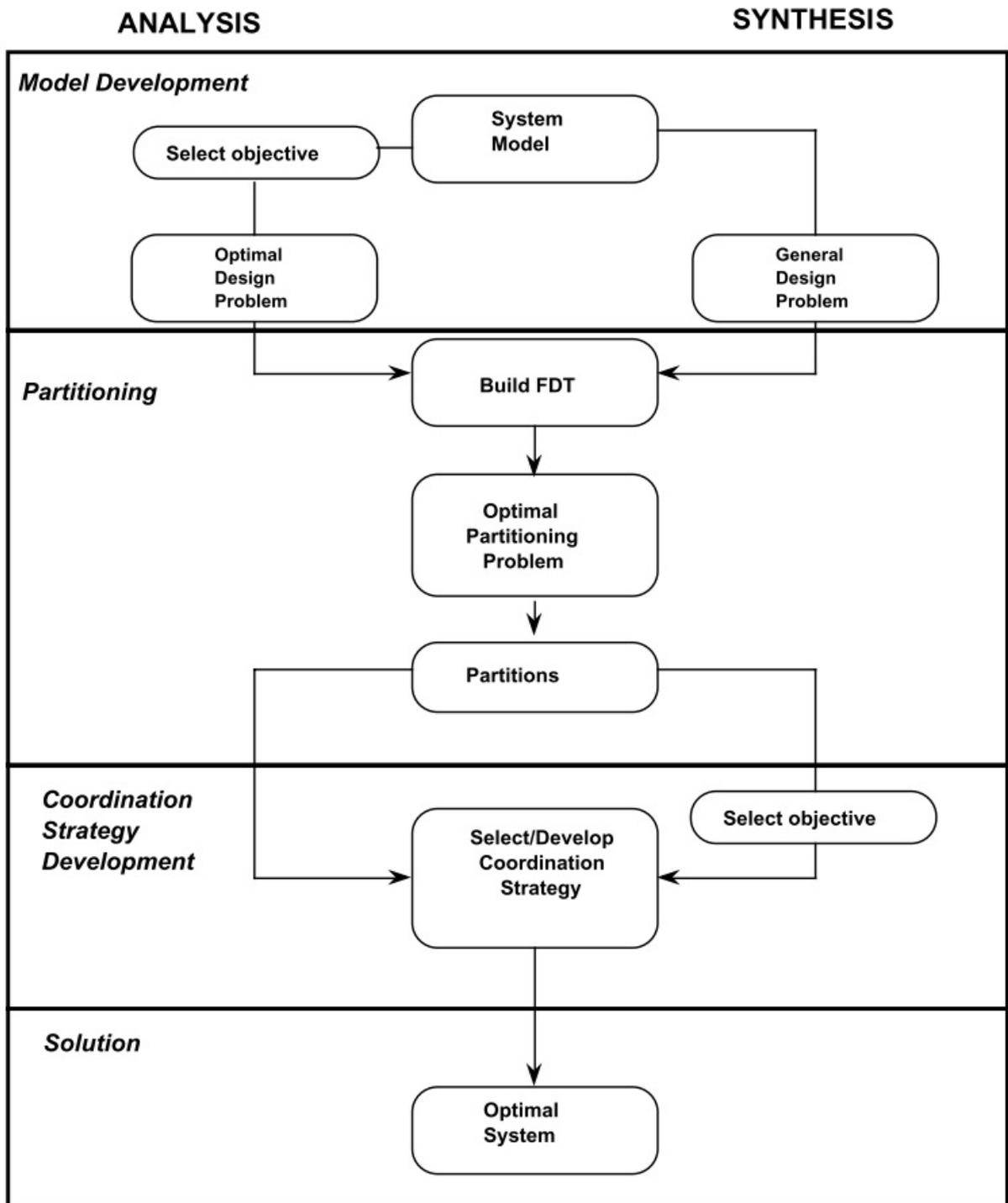


Figura 2.4: Fluxo para projeto de um sistema ótimo (Adaptado de [3]).

sistema completo é construído utilizando critérios de cada sub-sistema, ou seja sintetizado a partir da decomposição (*decomposition synthesis*). Para maiores detalhes sugere-se [40], [3], [17] e [43], além dos trabalhos onde foram propostos.

Comumente os métodos de decomposição, inclusive o deste trabalho, são independentes se a escolha da função objetivo foi realizada antes, análise, ou será realizada depois, síntese. Da mesma forma, não há interferência no método mesmo que se deseje um projeto ótimo para o sistema ou uma solução geral factível. Neste último caso o fluxo será similar ao fluxo descrito na Fig. 2.4 para a síntese, sendo a decomposição realizada sobre o GDP. A única diferença é que não haverá a escolha dos objetivos antes da integração. Ou, diante do problema decomposto, os projetistas podem perceber que é possível obter um projeto ótimo, apesar de inicialmente não ser possível identificar os objetivos para o sistema completo.

2.2.2.1 Objetivos da Decomposição Baseado em Modelo

Alguns trabalho abortaram a decomposição baseada em modelo como um problemas de otimização multi-objetivo, chamando de decomposição/partição ótima baseada em modelo (*optimal model-based decomposition/partition* - OMBD ou OMBP). Em [5] o problema é formulado com os dois seguintes objetivos conflitantes:

- Minimizar a quantidade de variáveis de ligação⁸;
- Minimizar o tamanho dos sub-problemas por maximização do número de partições, tal que sejam de tamanhos similares.

Em outro trabalho, [11], os mesmos autores definiram o OMBD com os seguintes objetivos conflitantes:

⁸Este objetivo é definido originalmente por Michelena e Papalambros em [5], como: "minimizar o tamanho do problema mestre por minimização das variáveis de ligação". Em um *framework* hierárquico de integração, o problemas mestre é um sub-problema que se comunica através das variáveis de ligação com seus sub-problemas (existindo interação apenas entre os sub-problemas e seus problemas mestres, e não entre sub-problemas de mesmo nível). Entretanto, em [6], os mesmos autores aplicando decomposição e integração ao projeto de um sistema de *powertrain* (trem de força ou sistema de tração) utilizam tanto a estratégia hierárquica quanto a distribuída (onde a não existe problemas mestres e a comunicação é feita diretamente entre os sub-problemas através das variáveis de ligação). Desta forma, a definição "minimizar o tamanho do problema mestre" perde seu sentido. Entretanto, no segundo trabalho não é feita uma nova definição do OMBD, apenas uma referência ao primeiro trabalho. Apesar disto, se observa que estes objetivos são tratados diretamente como "minimizar a quantidade de variáveis de ligação" tanto para a integração hierárquica, quanto a distribuída. As estratégias gerais hierárquicas e distribuídas são apresentadas na Seção 2.2.3.1.

- Minimizar as interconexões entre sub-problemas;
- Balancear o tamanho dos sub-problemas.

Onde as interconexões entre sub-problemas são dadas pelas quantidades de variáveis de ligação (*linking variables* ou *coordination variables* ou). Desta forma, podemos extrair quatro objetivos para um OMBD:

- Minimizar as interconexões entre os sub-problemas, ou seja a quantidade de variáveis de ligação;
- Minimizar o tamanho dos sub-problemas;
- Maximizar o número de sub-problemas;
- Maximizar o balanceamento do tamanho dos sub-problemas, ou seja, da similaridade dos tamanhos dos sub-problemas.

O tamanho dos sub-problemas tradicionalmente é definido em relação a quantidade de elementos (variáveis de projeto, funções objetivo, restrições de igualdade e desigualdade). Entretanto, existem aplicações onde pode ser mais interessante mensurar um sub-problema por outros critérios, como: a estrutura organizacional do time de projetistas, e a estimativa do custo computacional de cálculo das equação e funções, a execução das simulação envolvidas e acessos aos dados das superfícies de resposta (*response surface data*) [11]. No caso deste trabalho serão utilizadas as definições tradicionais, lembrando que estas informações também podem ser adicionadas ao modelo antes da decomposição.

Vale ainda ressaltar, que existem abordagens onde deseja-se que a ligação entre os sub-problemas seja realizada através de funções/equações de ligação (*linking functions*) ao invés das variáveis de projeto. Segundo [11], isto não é uma prática comum sendo preferível a utilização de variáveis de ligação. De qualquer, forma o método deste trabalho permite que ambas as abordagens sejam utilizadas com alterações somente na interpretação.

2.2.3 Integração do Sistema Decomposto

Pode-se imaginar que após a decomposição do modelo de um projeto em modelos de sub-projetos bastaria executar um algoritmo para definição das variáveis de projeto sobre cada sub-modelo e teria-se, assim, a solução para o problema original. Isto é o mesmo

que considerar que em um processo de dividir para conquistar, após a realização da etapa de divisão, a decomposição do modelo, a etapa de conquista seria trivial. Entretanto, o problemas de integração, etapa de conquista, é tão complexo quanto o de divisão.

Como ilustração, considera-se novamente, o projeto do sistema de tração da Fig. 2.3. Suponha que foi realizada uma decomposição baseada em objeto entre diversas equipes de engenheiros. Cada equipe realizou o melhor sub-projeto que conseguiu para o componente físico, sub-problema que lhe ficou designado. Entretanto, ao se reunirem para montar o sistema de tração eles perceberam que simplesmente os componente não se encaixavam devidamente.

Pode-se pensar que uma solução seria decompor o projeto do sistema de tração de forma sequencial, seguindo o fluxo de energia mecânica do motor até as rodas. Neste caso primeiramente os engenheiros desenvolveram o projeto de um motor econômico e potente que atendia todos os requisitos do projeto original. Sequencialmente todas as partes foram projetadas, até que nos últimos estágios percebeu-se, no projeto da roda, que devido as variáveis de projeto definidas para os componentes anteriores os únicos posicionamento ou dimensões possíveis não atendiam aos requisitos do projeto completo original.

O mesmo cenário indesejado também poderia ocorrer para qualquer outro tipo de decomposição. No caso da decomposição baseada no modelo, a resolução de um sub-problema pode gerar variáveis de projeto que sejam incompatíveis com a solução obtida pela resolução do outro sub-problema. Desta forma, podem levar o problema original para uma solução ruim ou até mesmo não factível.

Também, no caso de projetos residenciais, comumente há uma decomposição intuitiva por aspectos entre os projetos estruturais, hidráulicos, elétricos e arquitetônicos, não sendo raro observar conflitos entre eles na hora da construção [44]. Por exemplo, na residência da Fig. 2.5 o projeto elétrico e o hidráulico foram feitos separadamente e ambos projetaram utilizar um mesmo local para as instalações.

O problema de como realizar a conquista do projeto original a partir do sub-projetos obtidos pela decomposição é estudado e solucionado por métodos de integração, sendo uma área tão importante quanto a de decomposição.

Para o caso de residências simples uma estratégia comum de integração informal é deixar para descobrir *in loco*, ou seja, o chamado tradicional algoritmo do avestruz [45]. O conflito não desaparecerá só por que não foi identificado ou tratado. O projeto estará

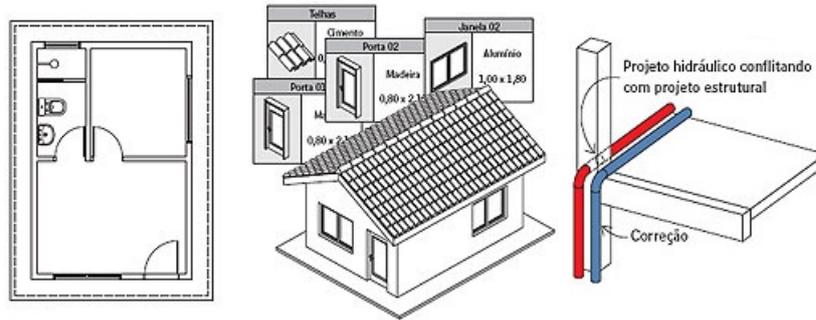


Figura 2.5: Conflito entre projeto elétrico e hidráulico residencial (Adaptado de [4]).

errado e deverá ser corrigido durante a construção, possivelmente de uma forma que não seria a ideal, e talvez nem aceitável segundo restrições de segurança ou custo. No pior caso, será necessário corrigir a casa após estar pronta, ou seja, por conta de uma integração mal feita dos sub-projetos, o projeto completo apresentará um erro, que não sendo corrigido durante a fabricação do artefato (a residência) gerou um defeito no produto e conseqüentemente uma manutenção corretiva precoce, junto com custos adicionais e limitações.

A própria Fig. 2.5 apresenta um método de integração melhor, onde um projetista recebe todos os projetos e os unifica em um único projeto tridimensional, que possibilita a visualização de conflitos e sua correção ainda antes da entrega do projeto final para a residência. Logo, é perceptível que para um bom projeto é necessário tanto um bom método de decomposição quanto uma estratégia adequada de integração.

2.2.3.1 Estratégias de Integração

Sistemas decompostos a partir do modelo devem ser integrados para que se obtenha a solução final para o sistema original completo. Relativo ao *lay out* (disposição das interações entre os sub-problemas) a integração pode ser classificada em duas estratégias gerais básicas: hierárquica (Fig. 2.6) e não hierárquica, também chamada de distribuída (Fig. 2.7).

Em qualquer estratégia a única comunicação, ligação (*link*), entre os sub-problemas é realizada através das variáveis de ligação (*linking variables*). Ou seja, toda a troca de informação, interação, entre os sub-problemas acontece exclusivamente pelo intercâmbio destas variáveis de projetos identificadas como comum a múltiplos sub-problemas. Portanto, pode-se dizer que elas coordenam (integram) os problemas, devido

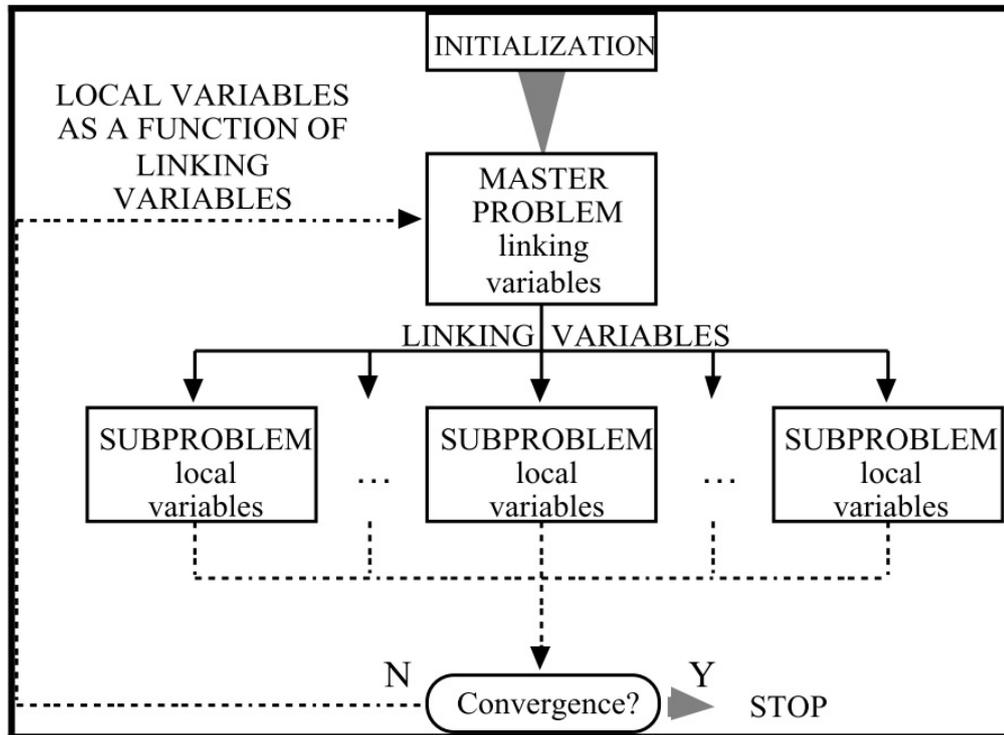


Figura 2.6: Estratégia genérica de integração hierárquica (Retirado de [5]).

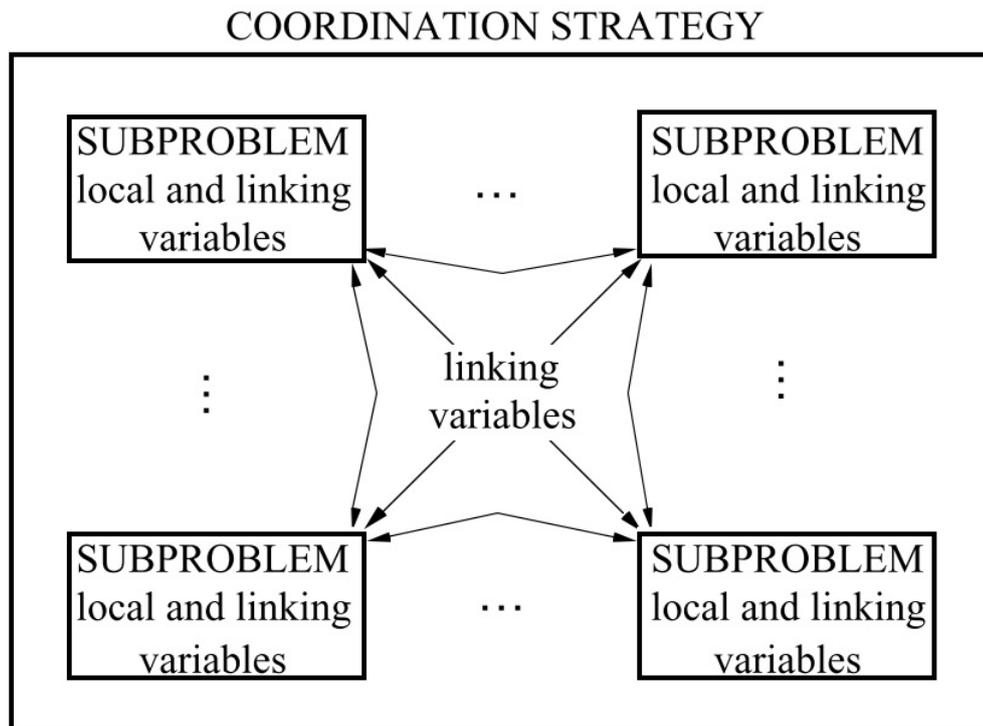


Figura 2.7: Estratégia genérica de integração não hierárquica (Retirado de [6]).

a isto também são chamadas de variáveis de coordenação (*coordinating variables*).

A estratégia hierárquica é utilizada quando se identifica que um dos sub-problemas (normalmente o que contém a função objetivo) compartilha variáveis com múltiplos sub-problemas, mas os demais sub-problemas não compartilham variáveis entre si. Neste caso, o primeiro será nomeado de problema mestre e os demais serão sub-problemas filhos do problema mestre.

Já a estratégia não hierárquica é adequada para quando cada par de sub-problemas compartilhem algumas poucas variáveis entre si, sendo que as mesmas não são compartilhadas por estes sub-problemas com os demais. Desta forma, cada sub-problema interagirá exclusivamente com os sub-problemas que compartilha variáveis de ligação, sendo que com cada um destes "vizinhos" há a comunicação apenas da variável de interesse mútuo.

Estas estratégias não são restritas apenas a um único nível. Os sub-problemas podem ser novamente decompostos e integrados até que seja suficientemente simples solucioná-los. Entretanto, é importante que se um sub-problema foi decomposto novamente, seus sub-problemas filhos devem ser integrados antes da sua própria integração com os sub-problemas irmãos. Isto produz estruturas de divisão e conquistas análogas a da Fig. 2.2 no desenvolvimento de um algoritmo recursivo.

Também não é raro configurações de integração híbridas tanto em níveis distintos quanto no mesmo nível. E ainda o cenário mais desejado em que todos os sub-problemas não compartilham qualquer variável de ligação, ou seja, não são interdependentes. Neste caso, basta solucionar cada sub-problemas separadamente para a suas variáveis locais e retorna-las como solução do sistema completo.

Por mais que o caso de não haver interdependência pareça utópico, não são raros os sistemas complexos que tenham ao menos um pequeno grupo de componentes disjuntos dos demais, sendo que pelos sistemas serem tão grandes e complexos os projetistas não conseguem identificá-los. Então, um processo de decomposição destes sistemas revelaria os componentes independentes permitindo que estas variáveis de projeto sejam obtidas diretamente.

Existem, ainda, outras estratégias gerais, que podem ser entendidas como especificações ou derivações das descritas anteriormente. As principais são: *sequential decomposed programming* e *hierarchical overlapping coordination* (HOC). Indica-se [17]

como uma fonte para um conhecimento inicial.

2.2.3.2 Categorias dos Métodos de Integração

Definida a estratégia de integração é necessário definir como será realizada a transferência de informação de cada variável de ligação entre os respectivos sub-problemas ou com o problema mestre. Sendo este a principal dificuldade na integração de componentes decompostos de um sistema.

Segundo [40], os diversos métodos encontrados na literatura pertencem a três categorias: factível (*feasible*), dual, e factível-dual (*dual-feasible*). Nos métodos factíveis as variáveis de ligação eram variáveis primais. Já, nos métodos duais elas são variáveis duais associadas com as respectivas restrições onde a variável de ligação originalmente ocorria. Por fim, os métodos factíveis-duais usam tanto variáveis primais quanto duais e suas respectivas funções para comunicação dos sub-problemas.

Tradicionalmente estes métodos eram projetados sobre estruturas específicas de cada problema. Entretanto, por volta da década de 1990 iniciou-se a identificar de forma rigorosa estruturas nas interações dos sub-problemas que relacionassem a específicas técnicas candidatas para resolução do problema de integração. Wagner [46] apresenta um compêndio dos métodos clássicos e algumas relações com as estruturas de interação das variáveis de ligação.

2.2.3.3 Ordenação Decomposição e Integração

Na Seção 2.2.2.1 foi apresentado que a decomposição de um projeto pode ser vista como um problema de otimização que visa a dividir o GDP ou ODP de forma ótima. Por sua vez, na Seção 2.2.3, ao estudar a integração também se percebe que quanto menos decomposto um sistema mais fácil é combina-lo de volta. Desta forma, a partir de um ponto de vista de otimização multiobjetivo, pode-se compreender estes dois problemas como tendo objetivos conflitantes.

A forma, mais natural e tradicional de tratá-los, é por meio do fluxo apresentado na Seção 2.2.2 e Fig. 2.4. Sendo a principal forma de implementação do fluxo é uma opção de implementação da estratégia chamada de decomposição para integração ($C \rightarrow D$), a qual todas as técnicas apresentadas anteriormente utilizam. Pois o presente trabalho se dedica exclusivamente a decomposição realizada sem qualquer restrições impostas para a

forma de combinação dos resultados dos sub-problemas.

Entretanto, existem outras estratégias de ordenar a decomposição e a integração. Allison *et al.* [47] apresentam quatro estratégias de realização das tarefas de decomposição e integração:

- Independente (D,C)
- Decomposição para Integração ($D \rightarrow C$)
- Coordenação para Decomposição ($C \rightarrow D$)
- Simultânea (D—C)

O presente trabalho se dedica apenas ao problema de decomposição independente de coordenação, portanto também apresentou a forma principal e tradicional de ordenação do problema foco e da integração. No entanto, está além do escopo proposto aprofundar mais no assunto. Para maiores informações sobre as demais estratégias se sugere o próprio trabalho de Allison *et al.*. Também para o caso da estratégia simultânea, recomenda-se [48] e [49], onde a tarefa é realizada através de algoritmos genéticos multi-objetivos.

3 Redução de dimensionalidade

Neste trabalho as relações entre os elementos do modelo do sistema complexo a ser decomposto serão expressas através de representações matriciais conforme será apresentado na Seção 5.1. Tal que todas estas formas matriciais representativas dos modelos podem ser compreendidas como uma matriz de dados perante a ótica de um processo de descoberta de conhecimento em base de dados (*knowledge discovery in databases* - KDD), no qual objetiva-se que as relações com maior valor semântico sejam realçadas enquanto as de menor significado sejam removidas.

Sarkar *et al* [37] propõe a utilização da técnica de redução de dimensionalidade conhecida como decomposição em valor singular para evidenciar os padrões semânticos pela obtenção de um espaço reduzido onde o mapeamento dos elementos com relações fortes sejam mais similares que os elementos não relacionados ou com relações fracas. Seguindo na mesma linha, os métodos propostos por esta trabalho para resolução da decomposição baseada em modelo de problemas de projeto e de problemas de projeto ótimo também faram uso da redução de dimensionalidade.

Portanto, o presente capítulo visa apresentar os fundamentos de redução de dimensionalidade necessários para o desenvolvimento dos métodos propostos e para compreensão do método de referência utilizado na avaliação comparativa. Para isto, na seção seguinte, se iniciará descrevendo as bases estatísticas teóricas da redução de dimensionalidade.

3.1 Variáveis Aleatórias, Vetores Aleatórios e Matriz de Dados

Um conjunto de dados (*Dataset*) multivariados, \mathbf{X} , é definido por diversas observações/itens/medições de um vetor aleatório, X . Este, por sua vez, é um vetor de variáveis/atributos/características aleatórias/randômicas (V.A.).

Seja o vetor aleatório X (Eq. 3.1) abaixo:

$$X^T = X_1, X_2, \dots, X_k, \dots, X_p \quad (3.1)$$

Onde $X_1, X_2, \dots, X_k, \dots, X_p$ são variáveis aleatórias.

Os dados são ditos multivariados, quando o vetor aleatório é composto por múltiplas variáveis aleatórias ($p > 1$). Normalmente quando X é composto por uma única V.A., é chamado de monovariado e quando composto por duas V.A. é bivariado.

Dados bivariados são multivariados, pois p maior que um. Os métodos estatísticos (e computacionais) de análise e manipulação de dados multivariados são igualmente válidos para bivariados. Estes também podem ser estudados por uma abordagem específica e simplificada, não sendo esta utilizada neste trabalho, visto que o mesmo visa um estudo de redução de dimensionalidade de dados multivariados para qualquer valor de p .

Fazendo n observações das p variáveis aleatórias, ou seja, n medições do vetor aleatório X , obtemos a matriz de dados \mathbf{X} (Eq. 3.2) abaixo:

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{j1} & x_{j2} & \dots & x_{jp} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{bmatrix} \quad (3.2)$$

3.2 Dimensão do Espaço de Características

A dimensão do espaço de características dos dados é o número de variáveis aleatórias observadas, ou seja, a dimensão do vetor aleatório. Portanto, para a base de dados \mathbf{X} tem-se que a dimensão do espaço de característica de seus dados (ou a dimensão da base de dados) é p .

3.3 Redução de Dimensionalidade

“Redução de dimensionalidade de dados refere-se ao processo de derivação de um conjunto de graus de liberdade que pode ser ajustado para reproduzir grande parte da variabilidade observada” [50]. Ou seja, segundo Roweis [51], redução de dimensionalidade é o processo de reduzir o número de variáveis aleatórias consideradas.

A redução de dimensionalidade pode ser realizada por duas abordagens: seleção de

características e extração de características.

A primeira consiste em reduzir as V.A. para um subconjunto das V.A. originais. Ou seja, o novo espaço será um subespaço das características original retirando algumas dimensões e perdendo por completo as informações expressas por elas.

Normalmente, por esta abordagem, o problema de reduzir a dimensionalidade é tratado como um problema de otimização combinatória, que visa obter a melhor combinação das dimensões a serem mantidas, orientado por uma estratégia específica. Essas estratégias podem ser: *filter* (orientado pelo ganho de informação – *information gain*), ou *wrapper* (orientado pela precisão – *accuracy*).

A outra abordagem consiste em transformar o espaço de alta dimensão dos dados em um espaço com menos dimensões. Esta transformação pode ser linear ou não linear.

Um exemplo de transformação linear é o PCA (*Principal Component Analysis* – Análise de Componentes Principais) [52], que é o método de redução de dimensionalidade mais tradicional, conhecido e difundido.

3.4 Métodos

3.4.1 Análise de Componentes Principais

”A técnica dominante de transformação de características que transforma o dado para um espaço reduzido, o qual captura a maior parte da variância destes, é a Análise de Componentes Principais”[53] , do inglês, *Principal Component Analysis* (PCA), ”originalmente conhecida como Transformada de Karhunen-Loève”[54]e, também, chamada em outras áreas por Transformada de Hotelling e método da função ortogonal empírica (*Empirical Orthogonal Function*), segundo Fodor [55]

Ainda, de acordo com Shlens [56] , por ser um método simples e não paramétrico de extração de informação relevante de conjuntos confusos de dados, com mínimo pós-processamento pode ser utilizado para redução da dimensionalidade e ser aplicado em diversas áreas, da neurociência a computação gráfica, o PCA é o mais valioso resultado da álgebra linear aplicada. Portanto, será o primeiro método abordado por este trabalho.

Uma análise de componentes principais é um método de estatística multivariada com o objetivo de ”explicar a estrutura de variância-covariância de um conjunto de variáveis aleatórias através de poucas combinações lineares destas.”[52]

Do ponto de vista da álgebra linear, o PCA encontra uma nova base para o espaço de características, a qual é composta pelos vetores de coeficientes das combinações lineares das variáveis originais, sendo estes vetores ortogonais entre si. Estas combinações lineares são chamadas de componentes principais (*Principal Components* - PC). E são definidas de forma que a primeira componente principal (PC_1) maximize a variância (o seu vetor de coeficientes - a_1 - será o primeiro vetor da base), a segunda componente (PC_2) maximize a variabilidade restante (o seu vetor de coeficientes - a_2 - será o segundo vetor da base), e assim por diante.

”Geometricamente, estas combinações lineares $[(PC_1, PC_2, \dots, PC_p)]$ representam a seleção de um novo sistema de coordenadas obtido por rotação do sistema com [as V.A.] X_1, X_2, \dots, X_p como eixos coordenados [(original)]” [52]. Sendo os vetores de coeficientes de cada componente (a_i) o vetor direcional do respectivo eixo coordenado.

Por exemplo, seja o conjunto de dados, \mathbf{X} , com dez amostras e duas variáveis aleatórias ($X^T = [X_1 X_2]$) definido abaixo:

$$\mathbf{X} = \begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}$$

Sabemos, da álgebra linear, que a base canônica para o espaço cartesiano de duas dimensões, e logo para o espaço das características em questão ($X^T = [X_1 X_2]$), é $B = (0; 1), (1; 0) = \hat{i}, \hat{j}$. Aplicando o PCA uma nova base para o mesmo espaço é obtida: $B_{PCA} = a_1, a_2 = (-0,70711; 0,70711), (-0,70711; -0,70711)$.¹ Onde os vetores são ortogonais entre si ($a_1^T \cdot a_2 = (-0,70711; 0,70711)_T \cdot (-0,70711; -0,70711) = 0$).

¹Dependendo da forma como os cálculos do PCA foram realizados pode-se obter uma base equivalente, onde o vetor de coeficiente seja o oposto do obtido, no exemplo. Mas, da geometria, sabemos que a direção do eixo coordenado definida tanto pelo a_1 quanto $-a_1$ são as mesmas.

O PCA, também, obtém a quantidade de variabilidade representada por cada componente. Que para o exemplo é 100% para a primeira e 0% para a segunda. Nota-se que de fato a primeira componente maximizou a representação da variância (e que a segunda maximiza a variância restante, no caso 0%).

Entretanto, apesar de obtermos um novo espaço de características $X_{PCA}^T = [PC_1 PC_2]$, que é muito útil para interpretação da estrutura de variância-covariância e análise dos dados, a dimensão continua a mesma ($Dim(X) = 2 = Dim(X_{PCA})$). Observando a variabilidade explicada por cada componente é perceptível que não há necessidade de representar os dados de \mathbf{X} por duas V.A. (características). E que pode-se reduzir a dimensão de X com uma perda mínima de informação representando-os em um novo espaço formado apenas pela(s) componente(s) que captura(m) a maior parte da variância.

Para tanto, definimos um novo espaço de característica (X_{red}) de menor dimensão (no caso, $Dim(X_{red}) = 1 = k < p = 2 = Dim(X)$), sendo as k primeiras componente principais os eixos deste novo espaço. Desta forma, para o exemplo, a base do novo espaço será: $B_{red} = v_1 = a_1 = (-0,70711; 0,70711)$. Ou seja, o novo espaço (vetor aleatório, vetor de características) é composto por apenas a primeira componente: $X_{red}^T = [PC_1]$.

Nota-se que como foi utilizada a componente que captura a maior parte da variabilidade dos dados, esta continuará a ser representada no novo espaço. No caso, a variância representada por PC_1 é de 100%, logo o novo espaço continuará a representar toda a variabilidade dos dados (ou seja, não perde, praticamente, qualquer informação).

Com o novo espaço determinado é hora de obter a matriz de Escore (*Score*) dos dados, isto é, transformar a matriz de dados original (\mathbf{X}) para a matriz dos dados no novo espaço ($\mathbf{X}_{red} = Score$).² Segue a matriz obtida para o exemplo:

²O mesmo procedimento pode ser utilizado para gerar os escores, a matriz de dados, para o espaço definido utilizando todas as componente principais (X_{PCA}), antes da redução. Isto é muito utilizado na estatística multivariada para interpretação destes.

$$\mathbf{X}_{\text{red}} = \text{Score}_{X_{\text{red}}}(\mathbf{X}) = \begin{bmatrix} 0.70711 \\ 0.70711 \\ -0.70711 \\ -0.70711 \\ -0.70711 \\ 0.70711 \\ -0.70711 \\ -0.70711 \\ 0.70711 \\ 0.70711 \end{bmatrix}$$

Percebe-se que ficou mais simples identificar os dois agrupamentos das amostras, o dos positivos (onde nos dados originais, primeira coluna era 0 e segunda 1) e o dos negativos (onde nos dados originais, primeira coluna era 1 e segunda 0). Isto ilustra o uso da redução de dimensionalidade como ferramenta de pré-processamento de dados em um processo de KDD, pois, da mesma forma que facilitou a tarefa de agrupar os dados manualmente, provavelmente, também, tornaria mais simples uma mineração destes dados.

Além da representação na dimensão permitir uma melhora na acurácia e na identificação de padrões/informações mais interessantes, ao ressaltar as relações mais importantes entre os dados. Ela ainda pode melhorar o desempenho computacional da técnica de mineração por diminuir a quantidade de dados a serem manipulados e, também, na economia de espaço de armazenamento.

Este foi um exemplo didático com uma relação muito simples entre os dados. Mas ele serve para mostrar os objetivos, o funcionamento e os resultados obtidos com o uso do método PCA e conseqüentemente o motivo de seu uso (tal qual de outros métodos de redução de dimensionalidade). Pois em bases de dados enormes, até mesmo esta simples relação de negação entre as V.A. pode ser difícil, nem que em custo de tempo de processamento, de ser identificada.

Imagine fazer este agrupamento manualmente de uma matriz de dados com mil amostras e cem atributos com a mesma relação de negação entre si. Com certeza este padrão não nos apareceria simplesmente e demandaria muito mais tempo de análise dos dados para extrairmos alguma informação relevante.

Diferentemente do exemplo, onde consegue-se reduzir a dimensão com perda de 0% da representação da variabilidade dos dados, em casos reais, normalmente, isto não é possível. Sendo necessário utilizar todas as componentes para se reproduzir a variabilidade do sistema por completo.

Entretanto, "frequentemente a maior parte da variabilidade pode ser explicada [(representada, capturada)] por um pequeno número k de componentes principais" [52]. Logo, frequentemente é possível reduzir a dimensão do espaço de características de p para k com perdas mínimas de informação.

3.4.1.1 Definição Matemática

Tradicionalmente na estatística a análise de componentes principais é definida populacionalmente para posteriormente se derivar sua definição amostral. Entretanto, será abordado diretamente do ponto de vista amostral. Pois, este trabalho trata da aplicação na extração de padrões sobre uma matriz amostral obtida a partir do modelo, onde se pretende aprender de forma não supervisionada as relações relevantes entre os elementos do modelo, e não inferir quais seriam as propriedades estatísticas de uma população dentro da qual os elementos do modelo seria uma amostra. Para as definições segundo a abordagem populacional e a derivação da amostral sugere-se consultar Johnson [52].

Dada uma matriz de dados \mathbf{X} com n amostras, observações $(x_i; i = 1, 2, n)$, com dimensão p , ou seja, $x_i^T = [x_{i1} x_{i2} \dots x_{ip}]$; $i = 1, 2, n$. É X , seu respectivo espaço de características, vetor de variáveis aleatórias (ou ainda, apenas, vetor aleatório) com p variáveis aleatórias, V.A., X_1, X_2, \dots, X_p (Eq. 3.3),

$$X^T = \begin{bmatrix} X_1 & X_2 & \dots & X_p \end{bmatrix} \quad (3.3)$$

E S a sua matriz de variância-covariância amostral não viciada³.

³Escolheu-se a matriz de variância-covariância amostral não viciada ($S = (n - 1)^{-1} \mathbf{X}_{\text{centralizado}}^T \mathbf{X}_{\text{centralizado}}$) ao invés da viciada ($S_n = n^{-1} \mathbf{X}_{\text{centralizado}}^T \mathbf{X}_{\text{centralizado}}$) conforme sugerido em Johnson [52] e utilizado na bibliografia de área de redução de dimensionalidade, mineração de dados e KDD [57] [58] e implementações dos métodos [59][56][60]

Considera-se as combinações lineares em 3.4.

$$\begin{aligned}
 Y_1 &= a_1^T X = a_{11}X_1 + a_{12}X_2 + \cdots + a_{1p}X_p \\
 Y_2 &= a_2^T X = a_{21}X_1 + a_{22}X_2 + \cdots + a_{2p}X_p \\
 &\vdots \\
 Y_p &= a_p^T X = a_{p1}X_1 + a_{p2}X_2 + \cdots + a_{pp}X_p
 \end{aligned} \tag{3.4}$$

Onde, $a_i^T = [a_{i1} \ a_{i2} \ \cdots \ a_{ip}]$ é o vetor de coeficientes da combinação linear Y_i .

Segundo [52], obtemos a variância (Eq. 3.5) e a covariância (Eq. 3.6) das combinações lineares de (Eq. 3.4).

$$Var(Y_i) = a_i^T S a_i \quad i = 1, 2, \dots, p \tag{3.5}$$

$$Cov(Y_i, Y_j) = a_i^T S a_j \quad i, j = 1, 2, \dots, p \tag{3.6}$$

As componente principais são, então, definidas como as combinações lineares 3.4 ortogonais, tal que a variância (Eq. 3.5) seja tão grande possível. Onde a primeira componente principal será a com maior variância.

Entretanto, percebe-se, que $Var(Y_i) = a_i^T S a_i$ pode ser aumentado indefinidamente multiplicando-se o vetor de coeficientes a_i por uma constante. A fim de eliminar esta indeterminação, adicionamos a restrição de que o vetor de coeficientes deve ser unitário.

Portanto, definindo o método PCA com uma otimização, temos Eq. 3.7 [52]

$$\begin{array}{ll}
\text{Primeira componente} & \text{combinação linear } Y_1 = a_1^T X \\
\text{principal } (PC_1) & = \text{que maximiza } Var(Y_1) = Var(a_1^T X) \\
\text{s.a.} & a_1^T a_1 = 1 \\
\text{Segunda componente} & \text{combinação linear } Y_2 = a_2^T X \\
\text{principal } (PC_2) & = \text{que maximiza } Var(Y_2) = Var(a_2^T X) \\
\text{s.a.} & a_2^T a_2 = 1 \\
& Cov(Y_1, Y_2) = Cov(a_1^T X, a_2^T X) = 0 \\
& \vdots \\
\text{i-ésima componente} & \text{combinação linear } Y_i = a_i^T X \\
\text{principal } (PC_i) & = \text{que maximiza } Var(Y_i) = Var(a_i^T X) \\
\text{s.a.} & a_i^T a_i = 1 \\
& Cov(Y_j, Y_i) = Cov(a_j^T X, a_i^T X) = 0 ; \\
& \forall j < i \\
& \vdots \\
\text{Última (p-ésima) componente} & \text{combinação linear } Y_p = a_p^T X \\
\text{principal } (PC_p) & = \text{que maximiza } Var(Y_p) = Var(a_p^T X) \\
\text{s.a.} & a_p^T a_p = 1 \\
& Cov(Y_j, Y_p) = Cov(a_j^T X, a_p^T X) = 0 ; \\
& \forall j < i
\end{array}
\tag{3.7}$$

E conseqüentemente as notações PC_i e Y_i são equivalentes para denotar a combinação linear que é a i -ésima componente principal.

Nota-se que o PCA pode ser realizado por diferentes técnicas que encontrem a matriz de vetores de coeficientes ($A = [a_1 a_2 \cdots a_p]$) resolvam o problema de otimização em Eq. 3.7 desde as mais simples até outras mais complexas (mas menos utilizadas, e portanto, além do escopo deste trabalho), como uso de redes neurais artificiais. Também existe uma formulação não linear para o PCA, sendo o Kernel PCA o seu método de solução mais conhecido [54] (apresenta mais informações sobre esta técnica).

3.4.1.2 Obtenção das Componentes Principais

O problema da obtenção das componentes principais é tradicionalmente solucionado através de ferramentas da álgebra linear e propriedades da estatística multivariada. Apesar de ser definido como uma maximização, podemos utilizar de certos resultados analíticos a partir das áreas citadas, para sua resolução.

A seguir será derivada esta forma de obter as componentes (conforme [52]).

Sabe-se que "as matrizes de variâncias-covariâncias (ou de correlações) são sempre matrizes semi-definidas positivas, ou seja, matrizes simétricas ($M_{p \times p}$) cujas formas quadráticas ($x^T M x$) são sempre não-negativas." (CADIMA, [20_]). E que maximização em um espaço unitário da forma quadrática de uma matriz M semi-definida qualquer, isto é, com autovalores não negativos ($\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_p \geq 0$) é igual ao seu maior autovalor (λ_1):

$$\max_{x \neq 0} \frac{x^T M x}{x^T x} = \lambda_1$$

Sendo que o valor de x que soluciona esta maximização é o autovetor unitário associado a este maior autovalor, ou seja, e_1 . Como os autovetores são perpendiculares (ortogonais) entre si, podemos definir genericamente que

$$\max_{x \neq 0, x \perp e_1, \dots, e_i} \frac{x^T M x}{x^T x} = \lambda_{i+1}; i = 0, 1, 2, \dots, p-1 \quad (3.8)$$

É atingido quando $x = e_{i+1}; i = 1, 2, \dots, p-1$ respectivamente, onde p é a dimensão da matriz M .⁴ Ou seja, $x = e_1$ soluciona:

$$\max_{x \neq 0} \frac{x^T M x}{x^T x} = \lambda_1$$

Em seguida, a solução é $x = e_2$ (e o valor máximo λ_2):

$$\max_{x \neq 0, x \perp e_1} \frac{x^T M x}{x^T x} = \lambda_2$$

E assim por diante até:

$$\max_{x \neq 0, x \perp e_1, \dots, e_{p-1}} \frac{x^T M x}{x^T x} = \lambda_p$$

⁴A prova pode ser obtida em [52]

Onde $x = e_p$ e o valor máximo é λ_p , o menor autovalor associado a M .

Portanto, seja S a matriz de variância-covariância associada com o vetor aleatório X . E $(\lambda_1, e_1), (\lambda_2, e_2), \dots, (\lambda_p, e_p)$ os pares de autovalores-autovetores de S . Onde e_i são autovetores normalizados e $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_p \geq 0$. Então, da Eq.3.8, com $M = S$ e $x = a$, temos:

$$\max_{a \neq 0, a \perp e_1, \dots, e_i} \frac{a^T S a}{a^T a} = \lambda_{i+1}; i = 0, 1, 2, \dots, p-1 \quad (3.9)$$

Onde, o a que soluciona a maximização é $a = e_{i+1}$. Ou seja:

$$\max_{a \neq 0, a \perp e_1, \dots, e_i} \frac{a^T S a}{a^T a} = \lambda_{i+1} = \frac{e_{i+1}^T S e_{i+1}}{e_{i+1}^T e_{i+1}}; i = 0, 1, 2, \dots, p-1$$

Entretanto, os autovetores são vetores unitários, logo, $e_j^T e_j = 1; k = 1, 2, \dots, p$.

Conseqüentemente:

$$\max_{a \neq 0, a \perp e_1, \dots, e_i} \frac{a^T S a}{a^T a} = \lambda_{i+1} = \frac{e_{i+1}^T S e_{i+1}}{e_{i+1}^T e_{i+1}} = e_{i+1}^T S e_{i+1}; i = 0, 1, 2, \dots, p-1$$

Da Eq. 3.5 temos que a variância de uma componente principal (lembrando $Y_j = PC_j$) é exatamente $Var(PC_j) = Var(Y_j) = e_j^T S e_j$. Portanto:

$$\begin{aligned} \max_{a \neq 0, a \perp e_1, \dots, e_i} \frac{a^T S a}{a^T a} &= \lambda_{i+1} = \frac{e_{i+1}^T S e_{i+1}}{e_{i+1}^T e_{i+1}} = e_{i+1}^T S e_{i+1} = Var(Y_{i+1}) = \\ &= Var(PC_{i+1}); i = 0, 1, 2, \dots, p-1 \end{aligned}$$

Ou seja,

$$Var(PC_j) = Var(Y_j) = e_j^T S e_j = \frac{e_j^T S e_j}{e_j^T e_j} = \lambda_j; j = 1, 2, \dots, p \quad (3.10)$$

Ainda, como os autovetores são perpendiculares entre si ($e_k^T \perp e_j = e_k^T e_j = 0; k \neq j$) e $S e_j = \lambda e_j$, temos:

$$Var(PC_k, PC_j) = Var(Y_k, Y_j) = e_k^T S e_j = e_k^T \lambda_j e_j = \lambda_j e_k^T e_j = 0; j = 1, 2, \dots, p \quad (3.11)$$

E como o a_i que soluciona a i -ésima maximização (Eq.3.9) é igual ao respectivo

autovetor ($a_i = e_i$), temos Eq.3.12:

$$\begin{aligned}
 PC_1 = Y_1 &= a_1^T X = e_1^T X = e_{11}X_1 + \cdots + e_{1p}X_p \\
 PC_2 = Y_2 &= a_2^T X = e_2^T X = e_{21}X_1 + \cdots + e_{2p}X_p \\
 &\vdots \\
 PC_p = Y_p &= a_p^T X = e_p^T X = e_{p1}X_1 + \cdots + e_{pp}X_p
 \end{aligned} \tag{3.12}$$

Onde o novo espaço de características $PC_T = [PC_1 PC_2 \cdots PC_p]$ tem a mesma variância total que o espaço original. Conforme demonstrado a seguir:

Cada elemento da matriz de variância-covariância associado a um vetor aleatório $M_T = [M_1 M_2 \cdots M_p]$ qualquer é definido por $Cov(M_i, M_j)$, logo os elementos da sua diagonal são: $Cov(M_i, M_i) = Var(M_i)$. Portanto tem-se que a variância total de M , ou seja, o somatório das variâncias de suas p V.A., é o traço da sua respectiva matriz de variância-covariância.

Para o caso do espaço de características original (X com matriz de variância-covariância S) temos:

$$Var_{tot}(X) = \sum_{i=1}^p Var(X_i) = tr(S)$$

Como visto, realizando a decomposição espectral de S obteve-se os autovalores $\lambda_1, \lambda_2, \cdots, \lambda_p$ e os autovetores $e_1 e_2 \cdots e_p$. Esta decomposição também pode ser descrita da forma matricial: $S = P \Lambda P_T$, onde Λ é a matriz diagonal dos autovalores e $P = [e_1 e_2 \cdots e_p]$ (observe que como os autovetores são os coeficientes das componentes lineares $A = [a_1 a_2 \cdots a_p] = [e_1 e_2 \cdots e_p] = P$). Desta forma, $PP_T = P_T P = I$, sendo I a matriz identidade. Portanto:

$$tr(S) = tr(P \Lambda P_T) = tr(\Lambda P_T P) = tr(\Lambda I) = tr(\Lambda) = \lambda_1 + \lambda_2 + \cdots + \lambda_p$$

Da Eq. 3.10 tem-se que $Var_i = \lambda_i$, logo

$$\begin{aligned}
 \sum_{i=1}^p Var(X_i) &= tr(S) = tr(\Lambda) = \lambda_1 + \lambda_2 + \cdots + \lambda_p = \\
 &= Var(Y_1) + Var(Y_2) + \cdots + Var(Y_p) = \sum_{i=1}^p Var(Y_i)
 \end{aligned}$$

Ou seja,

$$\sum_{i=1}^p \text{Var}(X_i) = \sum_{i=1}^p \text{Var}(Y_i) = \sum_{i=1}^p \text{Var}(PC_i) \quad (3.13)$$

A partir desta demonstração pode-se derivar uma fórmula para o cálculo da proporção da variabilidade (variância total) explicada por cada componente principal (Eq.3.14), informação essencial para a redução da dimensionalidade.

$$\left(\begin{array}{l} \text{Proporção da} \\ \text{variância total} \\ \text{segundo } PC_i \end{array} \right) = \frac{\text{Var}(PC_i)}{\text{Var}_{tot}(PC)} = \frac{\lambda_i}{\lambda_1 + \lambda_2 + \dots + \lambda_p} \quad (3.14)$$

Conseqüentemente a proporção acumulada da variabilidade representada pelas k primeiras componentes (tal que $k \leq p$) é dado pela Eq.3.15 abaixo:

$$\left(\begin{array}{l} \text{Proporção acumulada da} \\ \text{variância total segundo} \\ \text{as } k \text{ primeiras PC} \end{array} \right) = \sum_{i=1}^k \frac{\text{Var}(PC_i)}{\text{Var}_{tot}(PC)} = \sum_{i=1}^k \frac{\lambda_i}{\lambda_1 + \lambda_2 + \dots + \lambda_p} ; k \leq p \quad (3.15)$$

Desta forma, tem-se um algoritmo (Alg. 1) para obtenção das componentes principais associadas a um espaço de características X a partir de uma matriz de dados \mathbf{X} com n amostras neste espaço.

Algoritmo 1: Obtenção das componentes principais pela decomposição espectral da matriz de variância-covariância.

Entrada: matriz de dados (\mathbf{X}).

Saída: matriz de coeficientes das componentes principais (A); vetor das variâncias explicadas por cada componente principal (λ).

início

Obtenha a matriz de variância-covariância S associada a X ;
 Realize a decomposição espectral da matriz S ;
 Ordene os pares autovalor-autovetor (λ_i, e_i) , tal que, $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_p$;
 $A \leftarrow [e_1 \ e_2 \ \dots \ e_p]$;
 $\lambda \leftarrow [\lambda_1 \ \lambda_2 \ \dots \ \lambda_p]^T$;

fim

Onde a matriz de variância-covariância amostral não viciada, S , pode ser obtida

segundo o algoritmo Alg. 2 (ou pelo uso de bibliotecas de estatística⁵).

Algoritmo 2: Calcula a matriz de variância-covariância amostral não viciada.

Entrada: matriz de dados (\mathbf{X}).

Saída: matriz de variância-covariância amostral não viciada (S).

início

 | Obtenha a matriz de dados centralizada \mathbf{C} ;
 | $S \leftarrow \frac{1}{n-1} \mathbf{C}^T \mathbf{C}$;

fim

Por sua vez a matriz de dados centralizada ($\mathbf{C} = [c_1 c_2 \cdots c_n]^T$) é obtida subtraindo de cada amostra (x_i) o vetor de médias amostrais ($\bar{x} = [\bar{x}_1 \bar{x}_2 \cdots \bar{x}_p]^T$; \bar{x}_i é a média amostral da i -ésima V.A., X_i). Conforme ilustrada no Alg. 3.

Algoritmo 3: Centraliza a matriz de dados.

Entrada: matriz de dados (\mathbf{X}).

Saída: matriz de dados centralizada (\mathbf{C}).

início

 | Calcule o vetor de médias amostrais $\bar{x} = [\bar{x}_1 \bar{x}_2 \cdots \bar{x}_p]^T$;
 | $\mathbf{C} \leftarrow [x_1 - \bar{x} \ x_2 - \bar{x} \ \cdots \ x_n - \bar{x}]^T$;

fim

3.4.1.3 Obtenção das Componentes Principais para Dados Padronizados

Entretanto a solução apresentada está sujeita a influência da diferenças das unidades e escalas das características. Podendo ocorrer a dominância de algumas sobre as outras, não por suas relevância para o sistema, mas pela discrepância entre as respectivas escalas. Sendo este um cenário comum e problemático em um processo de KDD.

A fim de solucionar esta dependência as ações mais tradicionais utilizadas no processo são realizar a normalização ou a padronização dos dados, conforme [57] e [58]. Especificamente a padronização tem uma relação com os cálculos do PCA que é bastante útil.

Inicialmente será aplicado o método tradicional (utilizando a decomposição espectral da matriz de variância-covariância) ao espaço padronizado e posteriormente utilizaremos de suas propriedades para obter uma outra forma de solucionar o PCA para este cenário.

Seja $Z_T = [Z_1 Z_2 \cdots Z_p]$ o vetor característico obtido da padronização do espaço original ($X_T = [X_1 X_2 \cdots X_p]$) e $\mathbf{Z} = [z_1 z_2 \cdots z_n]^T$ a matriz de dados padronizada, tal que:

⁵Como a função $cov(X)$ da biblioteca *statistics* da linguagem de programação MatLab [59]

$$z_{ij} = \frac{(x_{ij} - \bar{x}_j)}{\sqrt{s_{jj}}}; i = 1, 2, \dots, n; j = 1, 2, \dots, p$$

Onde, z_{ij} é a coordenada na dimensão da j -ésima V.A. do espaço padronizado (Z_j) para a i -ésima amostra da matriz de dados padronizada (z_i). Da mesma forma, x_{ij} é a coordenada na dimensão da j -ésima V.A. do espaço original (X_j) para a i -ésima amostra da matriz de dados original (x_i). E \bar{x}_j é a média amostral da j -ésima variável aleatória do espaço original (X_j), a qual, também, tem variância amostral $s_{jj} = Cov(X_j, X_j) = Var(X_j) = s_j^2$.

O Alg. 4 realiza a padronização dos dados de uma matriz de dados de entrada:

Algoritmo 4: Padroniza os dados

Entrada: matriz de dados (\mathbf{X}).

Saída: matriz de dados padronizada (\mathbf{Z}).

início

 Obtenha a matriz de dados centralizada \mathbf{C} ;

 Calcule o vetor de desvios padrões amostrais $s^{1/2} = [s_1^{1/2} \ s_2^{1/2} \ \dots \ s_p^{1/2}]^T$;

$\mathbf{Z} \leftarrow [c_1 - s^{1/2} \ c_2 - s^{1/2} \ \dots \ c_n - s^{1/2}]^T$;

fim

Sendo que a centralização dos dados pode ser feita utilizando-se do Alg. 3.

Após a padronização, pode-se, então, utilizar a técnica desenvolvida na Seção 3.4.1.2 para se obter as componentes principais em relação ao espaço padronizado. Para tanto, basta informar a matriz de dados padronizada, \mathbf{Z} ao invés da matriz de dados original, \mathbf{X} , como entrada para o Alg. 2.

Entretanto, este processo pode ser realizado de uma outra forma, aproveitando-se da uma relação entre a matriz de variância-covariância do espaço padronizado e da matriz de correlação dos dados originais.

Johnson [52] demonstra que a matriz de variância-covariância amostral do espaço padronizado Z é igual a matriz de correlação amostral (R) do espaço original X , isto é, $S_Z = R_X$. Portanto, a partir deste resultado, podemos definir uma nova forma de solucionar o método do PCA.

Ao invés de se padronizar o espaço para então calcularmos sua matriz de variância-covariância e então realizar a decomposição espectral desta para definição das respectivas componente principais. Pode-se obter as componentes principais associadas ao espaço padronizado diretamente da matriz de dados original, conforme descrito no Alg. 5.

Algoritmo 5: Obtenção das componentes principais associadas ao espaço padronizado pela decomposição espectral da matriz de correlação dos dados originais.

Entrada: matriz de dados original (\mathbf{X}).

Saída: matriz de coeficientes das componentes principais associadas ao espaço padronizado (A); vetor das variâncias explicadas por cada componente principal associada ao espaço padronizado (λ).

início

Obtenha a matriz de correlação R associada a X ;

Realize a decomposição espectral da matriz R ;

Ordene os pares autovalor-autovetor (λ_i, e_i) , tal que, $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_p$;

$A \leftarrow [e_1 \ e_2 \ \dots \ e_p]$;

$\lambda \leftarrow [\lambda_1 \ \lambda_2 \ \dots \ \lambda_p]^T$;

fim

Onde, a matriz de correlação é definida matricialmente por $R = (D^{1/2})^{-1}S(D^{1/2})^{-1} = D^{-1/2}SD^{-1/2}$, tal que, $D^{1/2}$ é a matriz de desvios padrões amostrais (uma matriz diagonal, composta pelos elementos do vetor de desvios padrões amostrais, $s^{1/2}$).

Da mesma forma que demonstrou-se que a variância explicada pela i -ésima componente principal é o autovalor associado ao autovetor que por sua vez é o vetor de coeficientes da componente, ou seja, $Var(PC_i) = \lambda_1$, pode-se derivar é obtidas pela decomposição espectral de S_X ou R_X ⁶

3.4.1.4 Obtenção das Componentes Principais através da Decomposição em Valor Singular

Entretendo, a multiplicação $\mathbf{X}^T\mathbf{X}$, utilizada no cálculo da matriz de variância-covariância, está sujeita a um erro numérico quando alguns elementos da matriz de dados são muito pequenos.⁷ Então, para evitar-lo, pode-se solucionar o PCA utilizando-se da decomposição em valor singular, do inglês, *singular value decomposition* (SVD), conforme será verificado nesta seção.

A decomposição em valor singular de uma matriz $M_{n \times p}$ qualquer, segundo a álgebra linear é dada pela Eq. 3.16.

$$M = U\Gamma V^T \quad (3.16)$$

Onde, Γ ⁸ é a matriz diagonal retangular ($n \times p$), e os elementos de sua diagonal (s_{ii}) são

⁶Johnson [52]apresenta a demonstração detalhada.

⁷Um exemplo de matriz que ocasiona este erro é a matriz de Läuchli [61].

⁸No contexto de SVD, normalmente se utiliza S para se representar a matriz Γ , entretanto nesta seção

não negativos; U é a matriz ($n \times n$) dos vetores singulares pela esquerda ($U = [u_1 u_2 \cdots u_n]$); e V é a matriz ($p \times p$) dos vetores singulares pela direita ($V = [v_1 v_2 \cdots v_p]$). Tanto as colunas de U (os vetores singulares pela esquerda), quanto as de V (vetores singulares pela direita), são ortonormais entre si, portanto, $U^T U = I$ e $L^T L = I$

Multiplicando 3.16 por M^T a esquerda obtemos a decomposição em valor singular de $M^T M$ (Eq.3.17):

$$M^T M = (U \Gamma V^T)^T (U \Gamma V^T) = (V \Gamma^T U^T) (U \Gamma V^T) = V \Gamma^T U^T U \Gamma V^T = V \Gamma^T \Gamma V^T = V (\Gamma^T \Gamma) V^T \quad (3.17)$$

Note que a decomposição em vetor singular de $M^T M = V W V^T$; $W = \Gamma^T \Gamma$, também é sua decomposição espectral. Onde, V é a matriz de autovetores associada a $M^T M$ e $\Gamma^T \Gamma$ é a matriz diagonal dos autovalores associados.

Isto é, $V = [v_1 v_2 \cdots v_p] = [e_1 e_2 \cdots e_p] = P$ e $W = \Gamma^T \Gamma = \Lambda$, ou seja, o i -ésimo elemento da diagonal de Γ é igual a raiz quadrada do i -ésimo elemento da diagonal de λ ($diagonal(\Gamma^T \Gamma) = [\gamma_1 \gamma_1 \gamma_2 \gamma_2 \cdots \gamma_p \gamma_p] = [\gamma_1^2 \gamma_2^2 \cdots \gamma_p^2] = [\lambda_1 \lambda_2 \cdots \lambda_p] = \Lambda$; $diagonal(\Gamma) = [\gamma_1 \gamma_2 \cdots \gamma_p]$).

Lembrando que por convenção os autovalores são ordenados da forma $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0$ (e P é ordenada para que e_i seja o autovetor associado a λ_i). Da mesma forma, utilizamos esta convenção para os valores singulares: $\gamma_1 \geq \gamma_2 \geq \cdots \geq \gamma_p$ (e, conseqüentemente, V é ordenada para que v_i seja o autovetor associado a γ_i).

Retomando a matriz de dados \mathbf{X} com espaço característico $X^T = [X_1 X_2 \cdots X_p]$ temos que a sua matriz de variância-covariância amostral não viciada (S) é:

$$S = \begin{bmatrix} s_{11} & s_{12} & \cdots & s_{1p} \\ s_{21} & s_{22} & \cdots & s_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ s_{p1} & s_{p2} & \cdots & s_{pp} \end{bmatrix}; s_{ij} = s_{ji}; i, j = 1, 2, \cdots, p \quad (3.18)$$

Onde, a s_{ij} é a covariância não viciada entre X_i e X_j ($Cov(X_i, X_j)$), sendo $s_{ii} = Cov(X_i, X_i) = Var(X_i) = s_i^2$. O qual é determinado conforme segue:

será utilizado esta notação para não ocasionar confusão com a matriz de variância-covariância amostral, S .

$$s_{ij} = \frac{1}{n-1} \sum_{l=0}^n [(x_{li}\bar{x}_i)(x_{lj}\bar{x}_j)]$$

Sendo, \bar{x}_i a média amostral da i -ésima V.A. (X_i). Então pode-se reescrever a Eq.3.18 na forma matricial (Eq. 3.19).

$$S = \frac{1}{n-1} \begin{bmatrix} \sum_{l=0}^n (x_{l1}\bar{x}_1)^2 & \sum_{l=0}^n [(x_{l1}\bar{x}_1)(x_{l2}\bar{x}_2)] & \cdots & \sum_{l=0}^n [(x_{l1}\bar{x}_1)(x_{lp}\bar{x}_p)] \\ \sum_{l=0}^n [(x_{l2}\bar{x}_2)(x_{l1}\bar{x}_1)] & \sum_{l=0}^n (x_{l2}\bar{x}_2)^2 & \cdots & \sum_{l=0}^n [(x_{l2}\bar{x}_2)(x_{lp}\bar{x}_p)] \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{l=0}^n [(x_{lp}\bar{x}_p)(x_{l1}\bar{x}_1)] & \sum_{l=0}^n [(x_{lp}\bar{x}_p)(x_{l2}\bar{x}_2)] & \cdots & \sum_{l=0}^n (x_{lp}\bar{x}_p)^2 \end{bmatrix} =$$

$$= \frac{1}{n-1} \mathbf{C}^T \mathbf{C} \quad (3.19)$$

Tal que, \mathbf{C} é a matriz de dados centralizada (ou seja, $c_{ij} = x_{ij} - \bar{x}_j ; i, j = 1, 2, \dots, p$).

Foi visto na Seção 3.4.1.2 que as componentes principais são obtidas realizando a decomposição espectral da matriz de variância-covariância do espaço. Isto é o mesmo que a decomposição espectral de $Cov(\mathbf{X}) = S = \frac{1}{n-1} \mathbf{C}^T \mathbf{C}$. Ou seja, $P\Lambda P = S = \frac{1}{n-1} \mathbf{C}^T \mathbf{C} = \Gamma^T \Gamma = V(\Gamma^T \Gamma)V^T$.

Anteriormente, derivou-ses a decomposição em valor singular de uma matriz M qualquer (Eq.3.16) de forma a obter a decomposição associada a $M^T M$ (Eq.3.17). Para se obter uma outra técnica de solução, que evite o problema numérico da multiplicação $\mathbf{X}^T \mathbf{X}$ ou $\mathbf{C}^T \mathbf{C}$ ou $\Gamma^T \Gamma$ intrínseca ao cálculo da matriz de variância-covariância, precisa-se definir uma matriz M , tal que $M^T M = S = \frac{1}{n-1} \mathbf{C}^T \mathbf{C}$.

Então, seja, $M = \sqrt{n-1}^{-1} \mathbf{C}$ (onde n é o número de amostras de \mathbf{C}), onde \mathbf{C} é a matriz de dados \mathbf{X} centralizada. Tem-se, então:

$$M^T M = \left(\frac{1}{\sqrt{n-1}} \mathbf{C}^T \right) \left(\frac{1}{\sqrt{n-1}} \mathbf{C} \right) = \frac{1}{n-1} \mathbf{C}^T \mathbf{C} == P\Lambda P = V(\Gamma^T \Gamma)V^T$$

Lembre que a decomposição em vetor singular de M é $M = U\Gamma V^T$. Portanto, ao invés de se calcular a decomposição espectral da matriz de variância-covariância de \mathbf{X} ($Cov(\mathbf{X}) = S = \frac{1}{n-1} \mathbf{C}^T \mathbf{C}$), pode-se obter a matriz de componentes principais (PC) e matriz diagonal com suas respectivas variâncias (Λ), apenas realizando a decomposição

em valor singular de $\sqrt{n-1}^{-1}\mathbf{C}$. Onde a matriz de vetores singulares pela esquerda (V) serão as componentes principais ($e_i = v_i$), ou seja, $PC_i = V_i$) e os valores singulares (γ_i) ao quadrado são as variâncias explicadas por cada componente ($\gamma_i^2 = \lambda_i$, ou seja, $\Gamma^T\Gamma = \Lambda$).

Desta forma é possível obter as componentes principais para os dados não padronizados evitando os problemas numéricos do cálculo da matriz de variância-covariância. Da mesma forma, esta solução utilizando o SVD pode ser aplicada a dados padronizado substituído a matriz de dados \mathbf{X} pela matriz de dados padronizados \mathbf{Z} . Neste caso não há necessidade de centralizar Z , pois isto já é feito durante o processo de padronização (mas a execução desta centralização não geraria qualquer erro).

O Alg. 6 soluciona o PCA, evitando erros numéricos, através do uso do SVD (independente dos dados terem sido ou não previamente centralizados).

Algoritmo 6: Obtém as componentes principais utilizando pela decomposição em valor singular

Entrada: matriz de dados (\mathbf{X}).

Saída: matriz de coeficientes das componentes principais (A); vetor das variâncias explicadas por cada componente principal (λ).

início

Obtenha a matriz de dados centralizada \mathbf{C} /* Se X centralizada, esta linha é opcional. */ ;

$\mathbf{M} \leftarrow \sqrt{n-1}^{-1}\mathbf{C}$ Realize a decomposição em valor singular da matriz \mathbf{M} ;

Ordene os pares valor singular-vetor singular a esquerda (γ_i, v_i) , tal que, $\gamma_1 \geq$

$\gamma_2 \geq \dots \geq \gamma_p$;

$A \leftarrow [v_1 \ v_2 \ \dots \ v_p]$;

$\lambda \leftarrow [\gamma_1^2 \ \gamma_2^2 \ \dots \ \gamma_p^2]^T$;

fim

3.4.1.5 Redução de Dimensionalidade a partir das Componentes Principais

Após obtermos as p componentes principais, independentemente da técnica utilizada, podemos realizar a redução da dimensionalidade. Nesta seção serão abordados os critérios de decisão da dimensão k mais adequada para o novo espaço.

Um importante critério é a proporção da variabilidade explicada por uma componente (Eq. 3.14) e a proporção acumulada da variabilidade explicada pelas k primeiras componentes (Eq. 3.15). Pois, desta forma pode-se definir qual é o menor k , tal qual, a variabilidade capturada por esta componentes seja satisfatória, ou seja, que a perda de informação (variabilidade explicada pelas $p-k$ últimas componentes) seja suficientemente

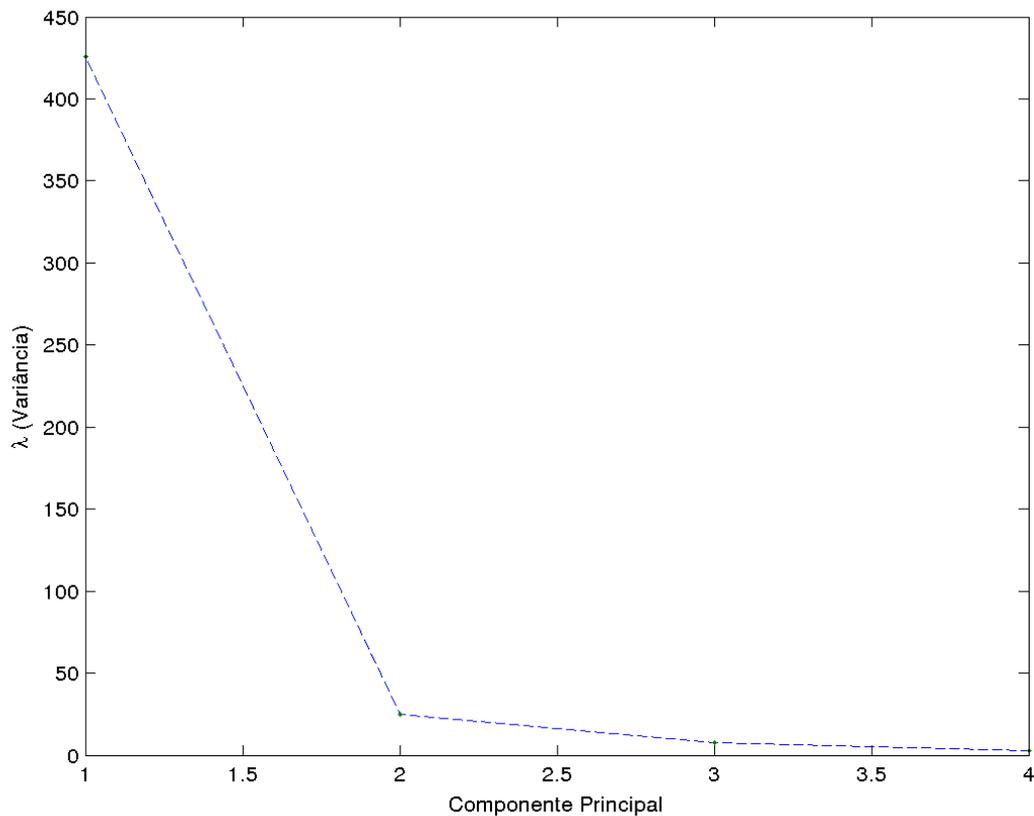


Figura 3.1: *Scree plot* do PCA para base de dados Iris

pequena.

Esta proporção satisfatória varia de aplicação para aplicação. Normalmente, representações que expliquem 85% da variabilidade dos dados são consideradas boas, em alguns casos este limite é aumentado para 95% [52]. Outra técnica é a análise do *scree plot* (gráfico de encosta⁹) do índice da componente pela sua respectiva variância. Na qual, procura-se, principalmente, encontrar o *cotovelo* (*elbow*) da curva, o local onde a uma alteração brusca na curvatura da curva.

A Fig. 3.1 mostra o *scree plot* de análise de componentes principais para a base de dados Iris [62] *apud* [63] a partir dos dados não padronizados. Pela análise deste gráfico temos que uma boa dimensão para o espaço reduzido seria $k = 2$, pois é neste ponto que temos o *cotovelo* da curva.

Entretanto nem sempre o gráfico de declive obtido poderá indicar a dimensão mais adequada, seja pela curva não apresentar um alteração brusca na sua curvatura, ou por

⁹Em referencia ao cascalhos de rocha ao pé de penhascos

apresentar várias alterações. Nesta última situação, os diversos pontos onde há esta variação brusca são bons candidatos a serem a dimensão mais adequada.

Em casos onde a redução seja aplicada para obtenção de uma representação onde a menor quantidade de dados é necessária para se economizar memória ou custo de manipulação destes, pode-se definir um limite ao tamanho dos dados e se utilizar a dimensão k , tal que, forneça a maior variabilidade acumulada dentro do limite de espaço imposto.

Outra estratégia de decisão para escolha do k que, também, não foca na quantidade de variabilidade explicada mas nos limites para k é a visualização de dados. Onde, devido a natureza da aplicação, o número de dimensões esta sujeito a uma, duas, ou três, $k = 1, 2, 3$.

Definido a quantidade de dimensões desejadas ($k \leq p$), temos então que o novo espaço é definido pelo vetor característico $X_{reduzido} = [PC_1 PC_2 \cdots PC_k]^T$, onde as k primeira componentes principais são as variáveis aleatórias. E os respectivos coeficientes, $a_i, i = 1, 2, \cdots, k$ (ou seja, os autovetores, $e_i, i = 1, 2, \cdots, k$), são os vetores unitários que dão a direção (e convencionalmente, também, o sentido) dos eixos deste espaço.

Por fim, com o espaço de características definido, os dados \mathbf{X} podem ser representados no novo espaço. Tradicionalmente esta representação dos dados é chamada de escore (*score*) dos dados, e portanto a nova matriz de dados é a matriz de Escores, **Score**. Mas outros nomes como matriz reduzida, matriz de dados reduzida, também são utilizados (**Score** $\equiv \mathbf{X}_{reduzido} \equiv \mathbf{X}_{red}$, ou ainda, indicando a dimensão do espaço, \mathbf{X}_{kD}).

Independentemente do nome utilizado, a transformação dos dados do espaço original para o reduzido é feito de acordo com a Eq. 3.20.

$$Score = X A_{reduzido} = X [a_1 a_2 \cdots a_k] = X [e_1 e_2 \cdots e_k] ; k \leq p \quad (3.20)$$

3.4.2 Decomposição em Valor Singular

Como visto na Seção 3.4.1.4 a decomposição em valor singular, do inglês, *singular value decomposition* (SVD), também conhecida como teorema de Eckart-Young (*Eckart-Young theorem*), é uma das técnicas mais comuns de resolução do PCA. Entretanto, o PCA via SVD ou pelas demais técnicas exige que os dados sejam centralizados, seja explicitamente (caso do SVD) ou implicitamente no cálculo da matriz de variância-covariância ou dos dados padronizados (Z). E, nem sempre, é desejável realizar este processo de obtenção da

média das V.A. e posterior subtração dela de todos os dados, pois dependendo do número de amostras (n) isto pode ser um processo custoso, ou indesejado se os dados forem muito dispersos.

Nestes casos, pode-se aproveitar de algumas propriedades da decomposição em valor singular para se realizar a redução de dimensionalidade. Como vimos, o SVD de uma matriz qualquer é dada pela Eq. 3.16. Portanto, seja esta a matriz de dados ($M = \mathbf{X}$), temos:

$$\mathbf{X} = USV = \sum_{i=1}^{\text{posto}(\mathbf{X})} s_i u_i v_i^T$$

Onde, $U = [u_1 u_2 \cdots u_n]$ é a matriz de vetores singulares pela direita, S a matriz retangular diagonal dos valores singulares, e $V = [v_1 v_2 \cdots v_p]$ a matriz de vetores singulares pela esquerda.

Tan [58] ressalta algumas propriedades desta decomposição:

- Os vetores singulares pela direita (u_i) capturam padrões entre as amostras (objetos);
- Os vetores singulares pela esquerda (v_i) capturam padrões entre os atributos (V.A.);
- A matriz decomposta (no caso, \mathbf{X}) pode ser aproximada de forma ótima pegando-se em ordem os termos da decomposição. Isto é, os autovalores são naturalmente ordenados ($s_1 \geq s_2 \geq \cdots \geq s_p$), e a matriz original é representada da melhor forma aproximada utilizando-se os k primeiros valores e vetores singulares (evidentemente, que quanto mais valores e vetores forem utilizados melhor será a precisão da aproximação);
- Pode-se obter uma nova matriz de dados da mesma forma que foi obtida a matriz *Score* no PCA (Eq. 3.20), substituindo a matriz dos k primeiros coeficientes ($[a_1 a_2 \cdots a_k]$) pela matriz dos k primeiros vetores singulares pela esquerda, $[v_1 v_2 \cdots v_k]$.

Destas afirmações, percebe-se que é possível se obter um espaço de menor dimensão através da aplicação direta do SVD. Onde, de forma análoga ao PCA, o novo espaço é definido pelo vetor de características $X_{reduzido} = [v_1 X_1 v_2 X_2 \cdots v_k X_k]; k \leq p$. E o eixo de cada dimensão é definido pela direção (e convencionalmente, o sentido) do respectivo vetor singular a esquerda.

Por sua vez, a transformação da matriz de dados original (\mathbf{X}) para a matriz de dados no novo espaço (ou matriz de escores, $\mathbf{X}_{\text{reduzida}} = \mathbf{Score}$) é obtida pela Eq. 3.21.

$$Score = XV_{\text{reduzido}}^T X[v_1 v_2 \cdots v_k] ; k \leq p \quad (3.21)$$

Enquanto os autovalores representam a variabilidade explicada pela componente principal a qual estão associados, os valores singulares representam o quanto a decomposição nesta dimensão se aproxima da matriz original. Portanto, $s_i > s_j$ significa que $s_i u_i v_i^T = Fator_i$ aproxima-se mais de \mathbf{X} do que $s_j u_j v_j^T = Fator_j$.

Então, analogamente, a proporção de variabilidade explicada por uma componente (Eq. 3.14) e a proporção acumulada das k primeira componentes (Eq. 3.15), tem-se, a proporção da aproximação da matriz original (Eq. 3.22).

$$\left(\begin{array}{c} \text{Proporção da aproximação} \\ \text{segundo } Fator_i = s_i u_i v_i^T \end{array} \right) = \frac{s_i}{s_1 + s_2 + \cdots + s_p} \quad (3.22)$$

Conseqüentemente a proporção acumulada da aproximação representada pelos k primeiros fatores (tal que $k \leq p$) é dado pela Eq.3.23 abaixo:

$$\left(\begin{array}{c} \text{Proporção acumulada da aproximação} \\ \text{segundo os } k \text{ primeiros fatores} \end{array} \right) = \sum_{i=1}^k \frac{s_i}{s_1 + s_2 + \cdots + s_p} ; k \leq p \quad (3.23)$$

Para se definir o número de dimensões utiliza-se as mesmas técnicas desenvolvidas para o PCA (Seção 3.4.1.5), adaptadas para o SVD:

- Menor k , tal que, a proporção de aproximação acumulada seja satisfatório (normalmente, maior que 85% ou 95%);
- Análise da *scree plot* dos valores singulares pelo índice do respectivo fator, escolhendo k de acordo com o *cotovelo* da curva;
- Maior k , tal que, o volume de dados não exceda um limite pré-estabelecido;
- $k = 1, 2, 3$ para aplicações de visualização dos dados.

4 Análise de Agrupamento e Detecção de Comunidades

Para realização da tarefa de decomposição, os modelos dos problemas de projeto geral ou ótimo podem ser compreendidos como um conjunto de pontos ou uma rede complexa (formalmente representadas pela estrutura de grafo) conforme descritas no Cap. 5. Da mesma forma que a técnica de redução de dimensionalidade, utilizada para evidenciar as relações semânticas mais relevantes, foi introduzida no Cap. 3, este capítulo aborda: a análise de agrupamento, provinda do aprendizado de máquina e comumente utilizado na etapa de mineração de dados em processos de KDD; assim como a área de Redes Complexas, dentro da qual o trabalho propõe tratar o problema de decomposição de sistemas complexos como uma tarefa de detecção de comunidades. Estas técnicas serão utilizadas para identificação, extração, dos sub-problemas sobre as representações das relações entre os elementos do modelo a ser decomposto, sendo que cada sub-problema será entendido como um agrupamento ou comunidade respectivamente.

4.1 Análise de Agrupamento

”Análise de agrupamento (*cluster analysis*), ou simplesmente agrupamento (*clustering*), é o processo de particionar um conjunto de dados (ou observações) em sub-conjuntos. Cada sub-conjunto é um grupo (*cluster*), tal que cada observação em um grupo são similares uma a outra, e ainda dissimilares as observações nos outros grupos” [57]. Em português o processo também é comumente chamado pelo estrangeirismo: *clusterização*.

Existem muitos algoritmos e técnicas de agrupamento, sendo o K-means [64] considerado como um dos principais algoritmos de agrupamento. Pelleg [65] afirma que isto se deve a sua simplicidade e propriedades de convergência para um mínimo local. Consequentemente a sua popularidade o K-means também se tornou o padrão de comparação e representante da tarefa de agrupamento.

4.1.1 *Kmeans*

O método K-means tem como objetivo particionar as amostras, observações, pontos no espaço característico, em um número pré-definido de k grupos, onde cada ponto pertence ao agrupamento do qual o centro ele mais se aproxima. Este centro, chamado de centroide (*centroid*), consiste no centro de massa do agrupamento, ou seja, na média (*means*) das amostras pertencentes ao respectivo grupo. Por isto, o nome de K-means, do inglês, K-médias.

O algoritmo tradicional do K-means consiste em: (i) iniciar os k centroides com um valor aleatório; (ii) atribuir cada ponto ao agrupamento do centroide mais próximo; (iii) atualizar o valor de cada centroides com a média dos pontos contidos nos respectivos agrupamentos; retorna ao passo (ii) e repete o processo até convergir, isto é, que não haja mais mudança na posição de qualquer centroide, significando que um ótimo local foi encontrado para a solução. Estes passos são representados formalmente no Alg. 7.

Algoritmo 7: K-means

Entrada: matriz de coordenadas dos pontos (\mathbf{X}); número de grupos desejados (k)

Saída: vetor com os índices do grupo de cada ponto (\mathbf{Z}).

início

Centroides \leftarrow Sorteia k pontos aleatórios

repita

para cada ponto em \mathbf{X} faça

 | Grupo _{i} \leftarrow Grupo do Centroide mais próximo;

fim para

 Centroides \leftarrow Médias dos pontos em cada grupo;

até convergir;

fim

4.1.1.1 Variância e Distância Euclidiana ao Quadrado

Considerando uma partição k_j que agrupa n_j observações o centroide pode ser definido pela média amostral (viciada) [52] conforme a Eq. 4.1.

$$\text{Centroide}_j = \bar{x}_j = \frac{1}{n_j} \sum_{i=1}^{n_j} x_{ij} \quad (4.1)$$

Por sua vez, a dissimilaridade da variância amostral (viciada) [52] entre um ponto do

grupo e o centroide é calculado pela Eq. 4.2.

$$s_{ij}^2 = (x_{ij} - \bar{x}_j)^2 \quad (4.2)$$

Sendo a variabilidade total do agrupamento definida pela Eq. 4.3.

$$s_j^2 = \frac{1}{n_j} \sum_{i=1}^{n_j} (x_{ij} - \bar{x}_j)^2 \quad (4.3)$$

Nota-se que a variância entre uma amostra e seu centroide é exatamente a distância euclidiana ao quadrado, conforme explicitado na Eq. 4.4.

$$\text{Distância Euclidiana}_{ij}^2 = \sqrt{(x_{ij} - \bar{x}_j)^2}^2 = (x_{ij} - \bar{x}_j)^2 = s_{ij}^2 \quad (4.4)$$

Por isto, o algoritmo K-means é chamado um método de análise de agrupamento baseado em distância (*distance-based cluster analysis*). Além da distância euclidiana ao quadrado, outras métricas de dissimilaridade também são utilizadas como medida da variabilidade do K-means. Os principais exemplos são: a distância de Manhattan, correlação (1 – correlação), e a dissimilaridade de cosseno [59],¹ sendo esta última especialmente importante na área de Recuperação de Informação [66], conforme abordado na Seção 4.1.1.2.

4.1.1.2 Similaridade de Cosseno

A similaridade de cossenos (*cosine similarity*) e sua respectiva métrica de dissimilaridade é uma das métricas mais difundidas na área de aprendizado de máquina, onde, a dissimilaridade de cossenos (*cosine dissimilarity*) é chamada comumente, mas incorretamente, de distância de cossenos (*cosine distance*). Apesar de ser de grande utilidade na comparação da similaridade ou dissimilaridade de vetores, esta métrica não pode ser considerada uma métrica de distância por violar a desigualdade triangular e permitir valores negativos [52].

Para definição da métrica, considere um par qualquer de vetores \mathbf{x} e \mathbf{y} no espaço característico e o ângulo θ entre eles, como ilustrado pela Fig. 4.1. O ângulo θ pode ser

¹Em implementações que utilizem de outras métricas de dissimilaridade, como a de cosseno, o K-means pode perder a garantia de convergência.

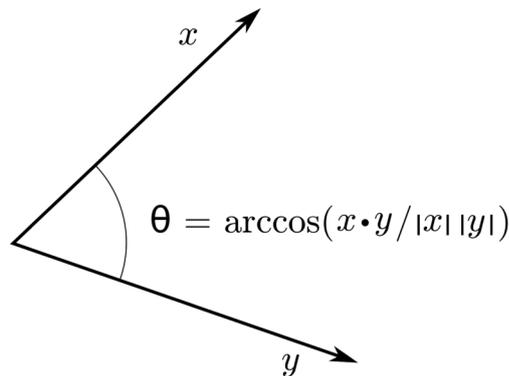


Figura 4.1: Ângulo entre dois vetores (Retirado de [7])

definido pela Eq. 4.5.

$$\theta_{\mathbf{x},\mathbf{y}} = \operatorname{arccos} \left(\frac{\mathbf{x}\mathbf{y}}{\|\mathbf{x}\| \cdot \|\mathbf{y}\|} \right) \quad (4.5)$$

Desta forma, a similaridade entre \mathbf{x} e \mathbf{y} é definida como o cosseno de $\theta_{\mathbf{x},\mathbf{y}}$, dado na Eq. 4.6.

$$\operatorname{Similaridade}(\mathbf{x}, \mathbf{y}) = \cos(\theta_{\mathbf{x},\mathbf{y}}) = \frac{\mathbf{x}\mathbf{y}}{\|\mathbf{x}\| \cdot \|\mathbf{y}\|} \quad (4.6)$$

Sabendo que a função cosseno leva qualquer ângulo para um escalar pertencentes ao intervalo $[-1; +1]$, onde, por exemplo: os valores de similaridade para vetores com mesma direção e sentido ($\theta = 0 \text{ rad} = 0^\circ$) será 1 ($\cos(\theta) = 1$); para os com mesma direção e sentido oposto ($\theta = \pi \text{ rad} = 180^\circ$) será -1 ($\cos(\theta) = -1$), e entre vetores ortogonais ($\theta = \pi/2 \text{ rad} = 90^\circ$) será 0 ($\cos(\theta) = 0$). Devido ao valor ser definido em relação ao ângulo entre os vetores fica consequentemente independente do da magnitude dos mesmos. É importante destacar que esta é uma propriedade de muito valia para a área de Recuperação de Informação, onde, a similaridade entre uma página e uma palavra (ou entre textos) será definida unicamente pela similaridade da representação vetorial de ambos, sem influências da quantidade de ocorrências, o que causaria problemas na recuperação devido ao enviesamento.

Alguns métodos de agrupamento como o próprio K-means utilizam a medida de dissimilaridade ao invés da similaridade, o que, devido as propriedades de intervalo da

função cosseno, pode ser facilmente obtido através da relação apresentada na Eq. 4.7.

$$Dissimilaridade(\mathbf{x}, \mathbf{y}) = 1 - Similaridade(\mathbf{x}, \mathbf{y}) = 1 - \cos(\theta_{\mathbf{x}, \mathbf{y}}) = \frac{\mathbf{x}\mathbf{y}}{\|\mathbf{x}\| \cdot \|\mathbf{y}\|} \quad (4.7)$$

4.1.1.3 Complexidade Computacional

O método K-means, como esperado, visto ser esta uma característica da tarefa de *clusterização*, independentemente de características específicas dos parâmetros, apresenta complexidade computacional NP-difícil (*NP-hard*) [67][68][69]. Ainda, segundo [70], pré-definindo k , sendo m a dimensão do espaço de características e n a quantidade de amostras, pode ser resolvido exatamente em $O(n^{dk+1} \log(n))$.

A principal implementação do K-means, chamada de algoritmo de Lloyd [71], utiliza-se de heurísticas para obter uma complexidade $O(knmi)$ (sendo i o número de interações até a convergência) para o caso médio. Outra heurística utilizada para melhorar a performance do algoritmo é conhecida como *K-means++* [72], e atua na definição dos pontos iniciais para os centroides garantindo que o mínimo local encontrado seja $O(\log k)$ competitivo com o desejado mínimo global. Neste trabalho foi utilizada o método *kmeans* presente na *Statistics and Machine Learning Toolbox* do *MatLab R2015a* [59], o qual implementa o algoritmo de Lloyd com o método do *K-means++* para a inicialização.

Existem outras implementações mais eficientes computacionalmente para o K-means, fazendo, por exemplo a utilização de *kd-trees* [65]. Entretanto, elas podem apresentar algumas limitações. No trabalho citado, a única métrica de dissimilaridade permitida é a distância euclidiana quadrada, existindo adaptações funcionais apenas para outras distâncias de Minkowski². Logo não seria aplicável ao problema de decomposição segundo a metodologia de resolução do presente trabalho.

4.2 Redes Complexas

Estamos rodeados por sistemas complexos que vão desde redes sociais, de comunicação, de infra-estrutura até interconexões neurais, interações genéticas e químicas, etc. Estes

²Distância de Minkowski é definida como $Minkowski_p = (\sum_{i=1}^n |a_i - b_i|^p)^{1/p}$. Sendo considerada uma generalização das distâncias euclidiana de Manhattan, as quais são obtidas fazendo $p = 2$ ou $p = 1$, respectivamente.

sistemas, são chamados de complexos devido a dificuldade de derivar o comportamento do todo a partir de seus elementos, sua compreensão desempenha um importante papel no dia a dia, nas ciências exatas, humanas e sociais, e em áreas de mercado indo da economia a engenharia. Segundo Barabási [8], o entendimento, predição, descrição matemática, e eventuais controle destes sistemas complexos são um dos maiores desafios intelectuais e científicos do século XXI. Barabási, ainda, acrescenta que por de traz de cada sistema complexo existe uma rede intrincada que codifica as interações entre os componentes dos sistemas, e que, "consequentemente, nós nunca compreenderemos sistemas complexos exceto que desenvolvamos um entendimento profundo da rede por de traz deles." [8]

O estudo e desenvolvimento de métodos para compreensão ou utilização destas redes de interações entre os elementos, ou componentes, de sistemas complexos é chamada de Redes Complexas (*Complex Networks*), e também, de Ciência de Redes (*Network Science*), que apresenta características como: interdisciplinaridade, ser dirigida por dados em quantidades massivas, necessidade de fundamentação matemática e uma intensa natureza computacional para tornar possível a manipulação e extração de informações e conhecimento útil.

Apesar de envolver tantos elementos e consequentemente diversas áreas do conhecimento sendo aplicada a inúmeras outras, as redes complexas ou redes de interações entre os elementos de sistemas complexos apresentam em seu estudo um conjunto de princípios e leis fundamentais que a definem como uma área de estudo própria. Os principais fundamentos teóricos são provenientes da teoria de grafos, indicando que, ao menos de certo modo, não se trata de uma disciplina com raízes recentes. Entretanto, o interesse e aplicação aos problemas modernos complexos e em escalas muito grandes são bastante recentes, do início deste século XXI. Isto pode ser claramente observado através da Fig. 4.2, onde nos últimos anos ocorreu uma explosão das citações anuais para dois dos mais importantes artigos em redes complexas [73] [74], segundo análise apresentada por [8].

4.2.1 Detecção de Comunidades

"Muitos sistemas complexos apresentam a existência de estruturas modulares que desempenham funções significantes e bem definidas no sistema. Em grafos, módulos são identificados como grupos densamente conexos com conectividade intra grupo esparsa. A

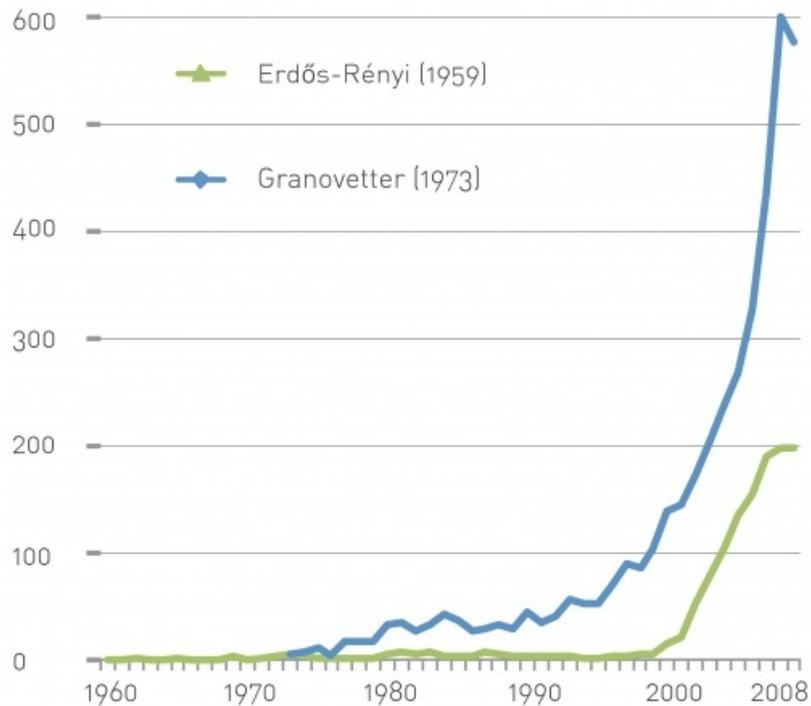


Figura 4.2: Citações anuais dos principais artigos de Redes Complexas (Retirado de [8])

identificação destes módulos é fundamental para revelar estruturas profundas de uma rede como um todo e padrões funcionais que podem ser causas da formação destas estruturas, ou resultantes delas.” [75]

Na área de Redes Complexas estes módulos são chamados de comunidades, e a tarefa de identificá-las é chamada de Detecção de Comunidades (*Community Detection*). Segundo Vieira *et al* [76] ”uma estrutura de comunidade em uma rede pode ser identificada quando existe uma divisão da rede em grupos com alta densidade das conexões internas, e, ao mesmo tempo, baixa densidade das conexões externas. O senso de comunidade se torna mais evidente com o crescimento da diferença entre o intragrupo e o intergrupo. Assim, tem-se um conceito central para quantificar a qualidade de uma divisão particular de uma rede em comunidades”.

Mais detalhadamente Barabási define comunidades em uma rede enunciando as duas hipóteses descritas abaixo:

1. Hipótese Fundamental: Uma estrutura de comunidade de uma rede é unicamente codificada no diagrama esquemático (*wiring diagram*), nas conexões entre os nós, da mesma;
2. Hipótese de Conectividade e Densidade: Uma comunidade é um sub-grafo

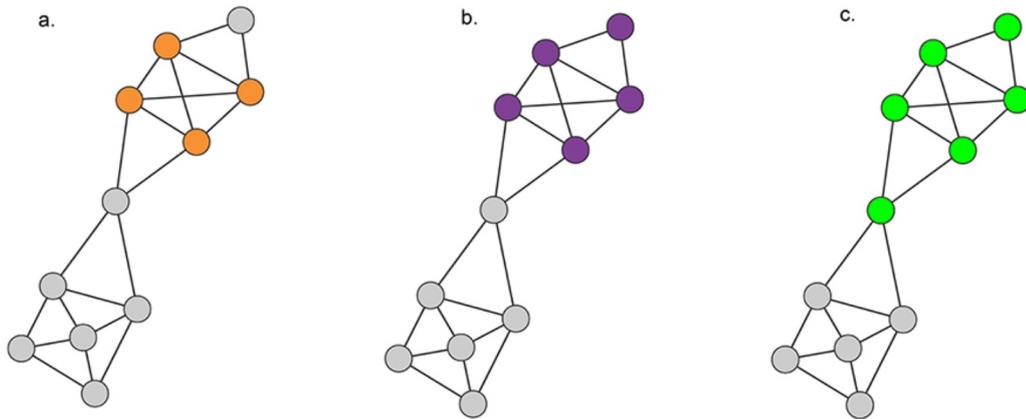


Figura 4.3: Exemplos de comunidades segundo definição em (a) clique, (b) forte, e (c) fraca (Retirado de [8]).

localmente denso em uma rede.

A hipótese fundamental assume que existe uma verdade fundamental (*ground truth*) sobre a organização das comunidades de uma rede, que pode ser descoberto por inspeção exclusiva de sua estrutura da ligação entre seus elementos, através de suas representações (por exemplo, matriz de adjacência).

Por sua vez a segunda hipótese estabelece que é esperado de cada nó pertencente a uma comunidade que tenha alta probabilidade de se conectar com outros nós da mesma comunidade ao invés de se conectar com nós externos (densidade), e que todos os membros de uma comunidade sejam acessíveis através de outros membros da mesma comunidade (conectividade). Estas hipóteses não definem por si só uma comunidade de forma rígida, entretanto, expõem o senso natural do que é uma comunidade, sendo, portanto, requisitos para identificação de uma comunidade, a qual pode ser formalmente definida de diversas maneiras, seguindo basicamente três linhas: comunidades de cliques, comunidades fortes e comunidades fracas.

Na teoria dos grafos um clique é um sub-grafo completo. Desta forma, a definição de comunidades baseada em clique é a mais restrita, exigindo que todos os componentes da comunidade comuniquem entre si (um exemplo é a comunidade laranja na Fig. 4.3(a), tendo assim a característica de produzir comunidades muito granuladas, muitas vezes apenas triângulos. Uma forma de relaxar a definição é chamada de comunidade forte (*strong community*), que consiste em sub-grafos conexos em que os nós tem mais ligações internas a comunidade do que com nós externos (a comunidade roxa na Fig. 4.3(b) é um

exemplo), formalmente esta relação é definida na Eq. 4.8.

$$d_i^{int}(C_j) > d_i^{ext}(C_j); \forall v_i \in C_j \quad (4.8)$$

Onde, C_j é a comunidade a qual o vértice (nó) v_i pertence; d_i^{int} é o grau interno de v_i , isto é, a quantidade de ligações internas a C_j d v_i , e d_i^{ext} o grau externo.

A terceira forma, é ainda mais relaxada que a segunda, aceitando comunidades com menor conectividade, e por isto uma comunidade segundo esta definição é chamada de comunidade fraca (*weak community*), e se trata de um sub-grafo no qual o grau interno total excede o grau externo total (a Fig. 4.3(c) apresenta um exemplo em verde), conforme descrito na Eq.4.9. Nota-se que a diferença de uma comunidade forte para uma fraca é que a forte se define perante aos graus (interno e externo) dos nós e a fraca perante aos graus (interno e externo) do sub-grafo, isto é, da comunidade.

$$\sum_{v_i \in C_j} d_i^{int}(C_j) > \sum_{v_i \in C_j} d_i^{ext}(C_j) \quad (4.9)$$

4.2.1.1 Particionamento de Grafos

O problema de particionamento de grafos é definido como particionar, dividir, um grafo em k sub-grafos, partições, fortemente interconectadas internamente e fracamente conectadas externamente. O uso de métodos de particionamento tem aplicação intensa em diversas áreas, destacando-se na definição do layout e encapsulamento de circuitos integrados (CIs, *integrated circuit*) e atribuição de tarefas em computação paralela [11]. Alguns dos mais difundidos são o algoritmos de Kerningham-Lin [] *apud* [11][8] e Fiduccia-Mattheyses [] *apud* [11], além de métodos iterativos baseados na confiança da rede (*network-reliability-based methods*) e métodos espectrais (*spectral methods*) [11].

Os algoritmos de particionamento de grafos são considerados os predecessores dos algoritmos de detecção de comunidades. Entretanto, segundo [8] existe uma importante diferença entre estas duas tarefas: o particionamento de grafos divide um grafo ou rede em um número k predefinido de sub-grafos, sub-redes, menores; contrastando com a detecção de comunidades que objetiva revelar as estruturas de comunidade intrínsecas na rede. Consequentemente, na maioria dos algoritmos de detecção de comunidade o número e o tamanho das comunidades não são pré-definidas, mas são objeto da busca a partir das

conexões entre os nós da rede.

Portanto, a diferença apresenta cunho tanto conceitual quanto prático. Por mais que algoritmos de particionamento de grafos, redes, foram utilizados em algumas aplicações de detecção de comunidades, principalmente no início, e continuam servindo de base para novas técnicas, os problemas tem objetivos diferentes. No caso do particionamento, também chamado de k -particionamento (*k-partitioning*) [11], dado um sistema se deseja particioná-lo em uma certa quantidade k de partições; já para a detecção de comunidades, dado um sistema se deseja revelar a estrutura de comunidades intrínseca a rede. Ou seja, ambos os problemas visam que as partições ou comunidades atendam a segunda hipótese (conectividade e densidade), entretanto apenas o problema de detecção de comunidades visa a primeira hipótese (fundamental).

Para melhor percepção da diferença, considera-se, por exemplo, uma rede aleatória (*radon network*), que por definição não apresentam estrutura de comunidades significativas. Para o problema de particionamento, não importa se existem ou não comunidades, partições, naturais na rede, apenas que se consiga subdividir da melhor forma possível a rede em k partições, o que em uma rede aleatória será aproximadamente qualquer configuração de sub-grafos conexos que produza partições de tamanhos iguais. Por sua vez, na tarefa de detectar as comunidades o ideal é que nenhuma (ou apenas um única contendo todos os nós) seja encontrada, visto a rede não apresentar nenhuma estrutura de comunidade na sua malha de interconexões, que é composta apenas de arestas aleatórias, ou seja, não há estrutura a ser revelada, nem comunidade a ser encontrada.

Em contraste com a tarefa de análise de agrupamento, nota-se que ela pode tanto ser relacionada ao particionamento, como é o mais natural para algoritmos como o K-means, onde o número de *clusters* é definido *a priori*; quanto a detecção de comunidades, quando é realizado por métodos que definem o número de grupos automaticamente de acordo com métricas de qualidade sobre a estrutura de agrupamento intrínseca aos dados, como por exemplo o X-means [65] e o X-Alter [77].

Tradicionalmente o problema de decomposição de sistemas para projeto geral ou ótimo em engenharia utilizou-se dos algoritmos de particionamento de grafo [5] [11], devido proximidade da área de decomposição de projetos com as principais aplicações de particionamento de grafos, as quais são em engenharia, mais especificamente no desenho

de circuitos integrados e computação paralela, além do evidente motivo cronológico das áreas terem ganhado foco simultaneamente durante o final do século XX e início do XXI, só que desenvolvidas primordialmente por comunidades acadêmicas distintas. A própria utilização explícita da inteligência artificial combinada a otimização remonta ao final do século XX [78]. Além disto, técnicas como representação simbólica do modelo, algoritmos de redução de dimensionalidade e agrupamento ainda não são muito difundidas, sendo [37] (2008) e [9] (2009) os primeiros artigos que intencionalmente relacionam ambas.

O presente trabalho pode ser considerado o primeiro a integrar a geração da decomposição com a detecção de comunidades em redes complexas, em conjunto com a representação simbólica e redução de dimensionalidade. Este é, devido a estas razões, mais um motivo pelo qual o presente trabalho prefere a utilização do termo decomposição em detrimento do termo particionamento para nomear a atividade de dividir um sistemas em sub-sistemas a serem projetados separadamente.

4.2.1.2 Complexidade Computacional

O problemas de detectar as comunidades de uma rede é portanto um problema maior computacionalmente do que o de particionar um grafo, visto que se para o último é necessário encontrar os melhores k grupos, para o primeiro se deseja descobrir os melhores grupos sendo a quantidade livre (definida pela estrutura e não como parâmetro). Desta forma, a complexidade da tarefa de detecção de comunidades pode ser analisada através da complexidade de um problema de particionamento caso já se saiba por outros meios qual é a quantidade de comunidades que ocorrem na rede (observe que é "saber" a quantidade de comunidades existentes e não "definir" a quantidade de partições desejadas). Sendo a complexidade total da detecção de comunidades a união da complexidade da tarefa de encontrar as k -partições e a tarefa de descobrir a quantidade de comunidades naturalmente existentes.

Barabási [8] demonstra que a quantidade de partições para uma rede com N nós, é representada pelo número de Bell (*Bell number*), B_N , apresentando um crescimento mais acentuado do que o da função exponencial (e^N), Fig. 4.4.

Ou seja, mesmo que já se soubesse a quantidade de comunidades de uma rede só o problema de particioná-las já seria de ordem exponencial. Ainda, Blondel *et al* [79] afirma que a tarefa de detectar as comunidades existentes baseado-se na métrica de

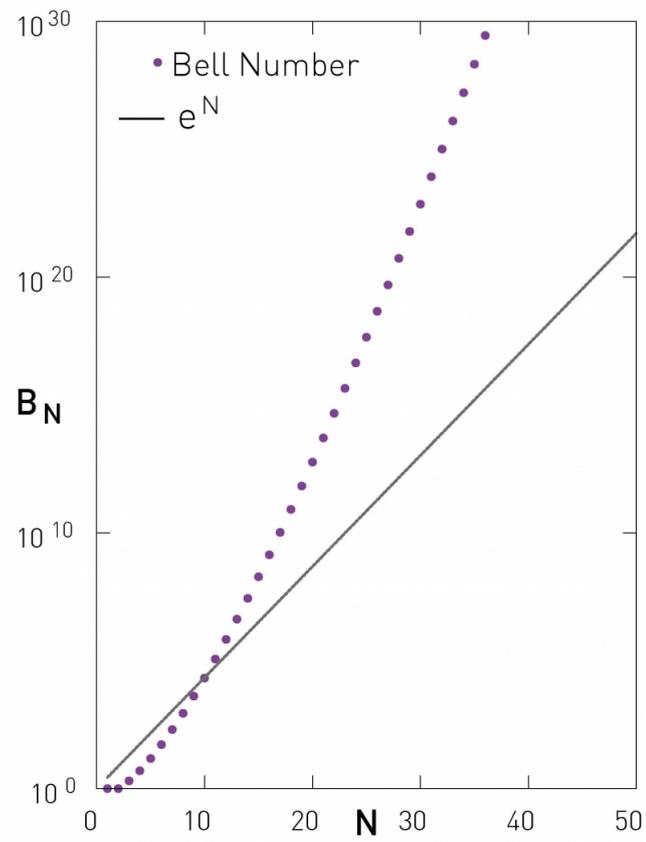


Figura 4.4: Quantidade de possíveis partições para um rede com N nós (Retirado de [8]).

qualidade é um problema NP-difícil (*NP-Hard*). Entretanto, assim como para a tarefa de agrupamento, existem diversos métodos de resolução que pretendem a identificação de uma estrutura satisfatória, que seja ao menos um mínimo local, indo de estratégias baseadas em otimização explícita de alguma métrica que indique a qualidade da estrutura de comunidade identificada, a algoritmos heurísticos que fazem isto implicitamente.

4.2.1.3 Modularidade

A modularidade é uma métrica proposta por Newman e Girvan [80] para mensurar a força da divisão de uma rede em módulos, comunidades, onde redes com altos valores de modularidade apresentam nós densamente conectados internamente aos seus módulos e esparsamente conectados externamente. Devido a esta característica é uma das principais métrica utilizada para guiar processos e algoritmos de detecção de comunidade através do objetivo de maximizar a modularidade das redes [75] [8].

Esta métrica admite que a qualidade da estrutura de comunidades pode ser mensurada considerando "a diferença entre a fração de arestas nas comunidades e a fração de arestas esperadas para uma rede com o mesmo grau de distribuição, mas aleatoriamente posicionados" [76], conforme descrito pela Eq. 4.10:

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{d_i d_j}{2m} \right) \delta(c_i, c_j) \quad (4.10)$$

Onde, Q é a modularidade, ou seja, a qualidade da estrutura de comunidades; m a quantidade de arestas na rede, A_{ij} é o elemento da i -ésima linha e j -ésima coluna da matriz de adjacência representativa da rede, d_i o grau do nó v_i ; c_i indica a comunidade a qual o vértice v_i pertence; e a função $\delta(c_i, c_j)$, chamada de delta de Kronecker, retorna 1 caso as comunidades sejam iguais e 0 se forem diferentes.

A Eq. 4.10, também pode ser escrita na forma da Eq. 4.11, onde o primeiro somatório indica o número de conexões existentes internamente a uma comunidade, enquanto o segundo indica a quantidade esperada para uma rede com mesmo grau de distribuição, mas com arestas aleatoriamente dispostas.

$$Q = \frac{1}{2m} \sum_{ij} A_{ij} \delta(c_i, c_j) - \frac{1}{2m} \sum_{ij} \frac{d_i d_j}{2m} \delta(c_i, c_j) \quad (4.11)$$

4.2.1.4 Método de Louvain

O método de Louvain, proposto por Blondel *et al* [81], é um algoritmo heurístico guloso que se baseia em otimizar a modularidade (*modularity* - Q) localmente, ou mais especificamente o ganho de modularidade (ΔQ) em cada passo, que garanta o ótimo local sendo concebido para detecção de comunidades de forma rápida em redes muito grandes (maiores que 100 milhões de nós). Apresentou um desempenho $O(n * \log(n))$ em avaliações empíricas em [79], onde solucionou o problema de detecção para uma rede com 2 milhões de nós em 2 minutos, e $O(n * \log^2(n))$ em experimentos realizados por [8].

Segundo [81] parte da eficiência do algoritmo resulta do fato de que o ganho na modularidade ΔQ é obtido movendo um nó isolado v_i para uma comunidade c_j (comunidade do vértice v_j) poder ser facilmente computado pela Eq. 4.12:

$$\Delta Q_{ij} = \left[\frac{\sum_{int} + d_{i,int}}{2m} - \left(\frac{\sum_{tot} + d_i}{2m} \right)^2 \right] - \left[\frac{\sum_{int}}{2m} - \left(\frac{\sum_{tot}}{2m} \right)^2 - \left(\frac{d_i}{2m} \right)^2 \right] \quad (4.12)$$

Onde, \sum_{int} é o somatório dos pesos das arestas internas da comunidade c_j ; \sum_{tot} é a soma dos pesos das arestas incidentes em nós pertencentes a c_j ; $d_{i,int}$ é a soma dos pesos das arestas que partem de v_i e incidem em nós de c_j ; e m é a soma dos pesos de todas as arestas da rede.

A detecção das comunidades no método de Louvain é realizada em dois passos (*steps*) repetidas iterativamente, chamados de otimização de modularidade (*modularity optimization*) e agregação das comunidades (*community aggregation*). No primeiro passo, identifica-se a qual comunidade cada nó pertence até que não haja mais melhora na modularidade; em seguida, no segundo passo, os nós de cada comunidade descoberta e suas respectivas arestas são combinados de forma a se obter uma rede de comunidades. A Fig. 4.5 apresenta o exemplo de uma rede com duas iterações (*passes*) do Louvain.

De forma mais detalhada, o passo de otimização da modularidade, o primeiro, inicia-se atribuindo a cada nó da rede uma comunidade distinta. Em seguida, para cada nó i calcula-se pela Eq. 4.12 o ganho de modularidade de movê-lo para a comunidade de cada um dos seus vizinhos (ΔQ_{ij}), havendo ganho positivo, move o nó i para a comunidade do nó j que gerou o maior ganho. Quando nenhum movimento individual puder melhorar a

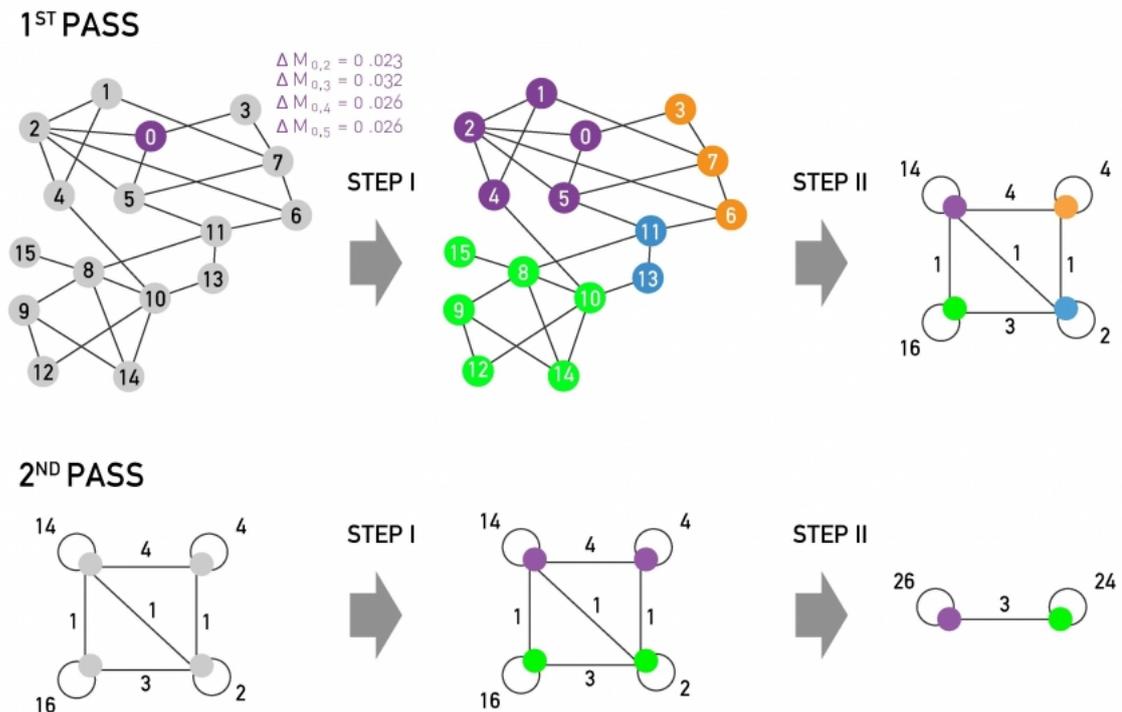


Figura 4.5: Exemplo de execução do método de Louvain (Retirado de [8]).

modularidade (mínimo local) o passo está concluído e segue-se para o passo de agregação das comunidades.

Este segundo passo, consiste em construir uma nova rede onde os nós serão as comunidades determinadas, para tanto, todos os nós de uma comunidade são agregados em um único nó que representará a comunidade como um todo, sendo que arestas entre os nós da mesma comunidade serão convertidos em laços (*self-loops*) e as arestas entre nós de uma comunidade para uma outra serão uma única aresta ligando as duas comunidades (o mesmo deve ser feito para cada comunidade que tenham pares de nós conectados). Por fim, os pesos de cada aresta (laço ou inter-comunidades) serão os somatórios de cada aresta original que foi unificada na respectiva aresta (se o grafo original for não valorado, deve-se considerar peso igual a 1 para cada aresta original).

5 Metodologia

Nos capítulos anteriores o problema de decomposição de sistemas complexos, grandes ou multidisciplinares para projetos gerais e ótimos em engenharia foi apresentado assim como as técnicas utilizadas pelos métodos de referência e propostos pelo presente trabalho para a resolução. Neste capítulo serão apresentadas as representações utilizadas, seguidas pelo método de referência para a avaliação comparativa será descrito, assim como a exposição dos métodos propostos pelo presente trabalho.

5.1 Representação

A metodologia de decomposição de sistemas baseada em modelo para projeto geral ou ótimo visa a obtenção dos melhores sub-sistemas partindo somente de informações contidas explícitas e implicitamente no modelo. Consequentemente uma representação de todas as relações de projeto é requerida, embora a forma matemática precisa não seja necessária para a decomposição [6]. A principal forma de representação do modelo utilizada para a decomposição na literatura é a tabela de dependência funcional (*functional dependence table* - FDT).

5.1.0.1 Tabela de Dependência Funcional

A representação por tabela de dependência funcional (*functional dependence table* - FDT) foi proposta por Wagner [39] *apud* [82] e consiste em uma forma de mapear as variáveis de projeto, acoplamento, ou estado, às funções objetivos e funções de restrições. As dependências que as funções tem das variáveis são representadas por uma matriz booleana chamada de FDT, tal que as linhas são rotuladas pelos nomes das funções ou outros tipos de relações (ex: simulações, superfícies de resposta) e as colunas são rotuladas pelos nomes das variáveis. A entrada da *i*-ésima linha e *j*-ésima coluna sera *Verdadeiro* se e somente se a *i*-ésima função depender da *j*-ésima variável, caso contrário a entrada será *Falsa* [11].

Tabela 5.1: FDT para o modelo da Eq.5.1

	x_1	x_2	x_3	x_4	x_5	x_6	f_1	f_2
f								
h_1								
h_2								
h_3								
h_4								
h_5								
h_6								
h_7								
h_8								

Para exemplificar, considere o ODP descrito na Eq. 5.1 [37]¹.

$$\begin{aligned}
 \min \quad & f = f_1 + f_2 \\
 \text{s.a.} \quad & h1 : f1 = x_1 + e^{x_1 x_4} \\
 & h2 : f2 = 2x_2 + 4x_5 \\
 & h3 : x_1 + 2x_2 + 5x_5 - 6 = 0 \\
 & h4 : x_1 + x_2 + x_3 - 3 = 0 \\
 & h5 : x_4 + x_6 - 2 = 0 \\
 & h6 : x_1 + x_4 - 1 = 0 \\
 & h7 : x_2 + x_5 - 2 = 0 \\
 & h8 : x_3 + x_6 - 2 = 0 \\
 & \mathbf{x} \in \mathfrak{R}^6
 \end{aligned} \tag{5.1}$$

A Tab. 5.1 apresenta a FDT derivada simbolicamente a partir da Eq. 5.1. Onde as células cinzas indicam a existência de dependência da função (equação, inequação, simulação, superfície de resposta, experimento, etc) em relação a variável (de projeto, de acoplamento - caso de $f_1 e f_2$ -, estado, etc).

A forma matricial booleana da FDT da Tab. 5.1 (relativa ao modelo na Eq. 5.1) é apresentada na Eq. 5.2. Onde, 1 (verdade, *true*) representa a existência de dependência,

¹O ODP da Eq. 5.1 retirado de [37] renomeia as restrições do apresentado em [11], o qual é uma modificação do ODP original de [83] apud [11].

Tabela 5.2: FDT conforme [11] para o modelo da Eq.5.1

	x_1	x_2	x_3	x_4	x_5	x_6
h_1						
h_2						
h_3						
h_4						
h_5						
h_6						
h_7						
h_8						

e 0 (falso, *false*) a ausência.

$$FDT(Eq.5.1) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad (5.2)$$

Formalmente, o representação matricial da FDT pode ser definida pela Eq. 5.3:

$$FDT_{ij} = \begin{cases} 1 & , \text{função}_i \text{ depende da variável}_j \\ 0 & , \text{caso contrário} \end{cases} \quad (5.3)$$

Alguns autores, como [11], não representam na FDT a função objetivo (no caso, f) - mas continuam representando as "sub-funções objetivos" (no caso, $h1 : f1 = x_1 + e^{x_1 x_4}$ e $h2 : f2 = 2x_2 + 4x_5$) -, e conseqüentemente não há mais sentido em apresentar as variáveis de acoplamento da função objetivo com suas "sub-funções" (no caso, f_1 e f_2). Desta forma, a FDT para o mesmo modelo (Eq.5.1) ficaria como apresentada na Tab. 5.2 e a forma matricial seria a da Eq. 5.4.

$$FDT_{[11]}(Eq.5.1) = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \quad (5.4)$$

Entretanto, este trabalho seguirá os trabalhos (por exemplo: [37] e [9]) que preferem a primeira forma apresentada (Tab. 5.1). A representação mais completa do modelo permitirá informar mais detalhadamente as relações ao método de decomposição, permitindo que ele identifique relações explícitas ou implícitas, inclusive entre os elementos excluídos da segunda representação. Também o método do presente trabalho se assemelha mais ao dos trabalhos que utilizam a primeira forma, ainda sendo estes os principais parâmetros de comparação das aplicações no Cap. 6.

5.1.0.2 Matriz de Ocorrência

Outra forma de representação é chamada de Matriz de Ocorrências (*occurrence matrix*), proposta por Sarkar *et al* [37]. Consiste em considerar que o modelo matemático ou não de um sistema que se deseje projetar seja um conjunto de símbolos, sendo que um grupo destes símbolos são chamados de eventos que ocorrem em episódios.

Em [22] os autores explicam que a inspiração para esta forma de representação é externa a área de projeto de sistemas em engenharia. Porém, é comuns em diversas áreas da Inteligência Artificial, como processamento estatístico de linguagem natural (*Statistical Natural Language Processing* - SNLP), abordagens de análise semântica latente (*Latent Semantic Analysis* - LSA), compressão de imagens para processamento de imagens digital (*Digital Image Processing* - DIP). Uma das aplicações mais importantes desta técnica de extração de padrão simbólico a partir de texto são as ferramentas de busca. Sendo a técnica chamada de modelo de espaço de vetores (*Vector Space Model* - VSM) um dos fundamentos da área de Recuperação de Informação (*Information Retrieval* - IR) [66].

A importância desta analogia irá além desta forma de representação, nas áreas citadas

a obtenção da matriz de ocorrência normalmente é seguida da aplicação do algoritmo SVD para redução de dimensionalidade, de forma a reduzir o volume de dados a ser manipulados minimizando a perda de informações relevantes. De forma similar, para a decomposição baseada em modelo, esta etapa também será utilizada, mas com uma motivação distinta das áreas que serviram de inspiração (Seção 5.2).

Em um mecanismo de busca (*search engine*) de IR, as palavras são consideradas eventos que ocorrem em episódios, que por sua vez são os textos. Então, o mecanismo de busca, obtém para cada episódio, texto, um vetor indicando todos os eventos que ocorrem no mesmo, isto é, todas as palavras que apareceram no texto. Isto pode ser representado por um vetor booleano que apresentará na i -ésima posição o valor 1 (verdadeiro) caso a i -ésima palavra do dicionário do mecanismo esteja presente no texto, ou 0 (falso) caso esteja ausente. Entretanto, também é possível que seja construído para cada palavra (evento) um vetor booleano que apresentará na j -ésima posição o valor 1 (verdadeiro) caso o j -ésimo texto contenha a palavra, ou 0 (falso) caso não contenha.

A partir dos vetores para diversos textos (episódios) e palavras (eventos), se obtém um espaço vetorial que representa estas relações episódios-eventos (textos-palavras). Devido este processo gerar uma massa de dados enorme é realizada uma redução de dimensionalidade, sendo a decomposição por valor singular o método mais tradicional para tanto nas áreas de RI e LSA. Vale lembrar da característica do SVD de que em uma única fatoração consiga capturar, extrair, tanto padrões das amostras (observações, linhas) quanto atributos (características, colunas), conforme apresentado no Cap. 3.

Após a redução, sempre que uma nova palavra for apresentada ao mecanismo de IR, este a transformará para o espaço vetorial reduzido e utilizará uma métrica de similaridade para identificar qual o texto mais próximo. Da mesma forma, este processo pode ser realizado para a requisição (*query*) de um texto, obtendo as palavras que o melhor representam. Ou seja, que tenham uma relação semântica (de significado) mais relevante.

Portanto, é possível apenas de informações sintáticas, a estrutura dos textos ou do modelo, extrair padrões que representem relações semânticas, ou seja, de significado, as quais poderiam estar explícitas ou implícitas na sintaxe (estrutura).

Voltando a representação para o modelo do sistema a ser decomposto, considera-se as variáveis como eventos (assim como as palavras em um mecanismo de IR) que ocorrem em episódios, no caso as funções (analogia aos textos). A matriz de ocorrência será um

Tabela 5.3: Tabla de ocorrências para o modelo da Eq.5.1

	f	h_1	h_2	h_3	h_4	h_5	h_6	h_7	h_8
x_1		■		■	■		■		
x_2			■	■	■			■	
x_3					■				■
x_4		■				■	■		
x_5			■	■				■	
x_6						■			■
f_1	■	■							
f_2	■		■						

representação vetorial da sintaxe, estrutura, do modelo, e a metodologia de decomposição extrairá os padrões semânticos, de significado, explícitos e implícitos de relacionamento entre os eventos e episódios, ou no caso, variáveis e funções.

Conseqüentemente, a confecção da matriz de ocorrências pode ser definida pela Eq. 5.5:

$$Occ_{ij} = \begin{cases} 1 & , \text{variável}_i \text{ ocorre na função}_j \\ 0 & , \text{caso contrário} \end{cases} \quad (5.5)$$

Para o modelo de exemplo descrito na Eq. 5.1, obtém-se, então, a matriz de ocorrência Occ descrita matricialmente na Eq. 5.6. A qual é apresentada também na Fig. 5.3 como uma tabela de valores booleanos (similar ao realizado para FDT na seção anterior).

$$Occ(Eq.5.1) = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (5.6)$$

Observa-se que a matriz de ocorrência (Eq. 5.6) é a transposta da representação matricial da FDT (Eq. 5.2) para o modelo da Eq. 5.1. Destaca-se que, apesar da inspiração, a matriz de ocorrência atribui 1 aos elementos que representam a ocorrência de uma variável em uma função, e 0 caso não ocorra (Eq. 5.5). Por sua vez, a FDT

atribui 1 aos elementos que representam a dependência de uma função em uma variável, e 0 caso a variável esteja ausente na função (Eq. 5.3).

Ou seja, a FDT relaciona a função com a variável, e a matriz de ocorrência relaciona a variável com a função. Reescrevendo no caso da FDT, "função na qual ocorra a variável" temos uma definição equivalente a "função que depende da variável". O contrário também acontece para a matriz de ocorrência, que poderia ser reescrita como "variável da qual a função é dependente", que seria equivalente a "variável que ocorre na função". Desta forma percebe-se que de fato toda matriz de ocorrência é a transposta da FDT, e vice-versa:

$$FDT = Occ^T \quad (5.7)$$

Desta forma, no restante deste trabalho será adotada a utilização da matriz de ocorrência por conta da sua analogia as técnicas simbólicas de Inteligência Artificial, que serão utilizadas para a decomposição baseada em modelo realizada por este trabalho. Seguindo desta forma o mesmo que foi feito em [37] e [9], visto a equivalência entre Occ e a FDT^T (Eq. 5.7).

5.1.0.3 Grafo Bipartido

Para a utilização do método de Louvain (Seção 4.2.1.4) na detecção de comunidades entre elementos (símbolos) do modelo, é necessário que este seja representado em forma de grafo. Mais exatamente como a matriz de adjacência de um grafo (devido a implementação utilizada [84]). Diversas representações são possíveis tanto em grafo quanto em hiper-grafo².

Neste trabalho, é proposto representar os modelos através de grafos bipartidos e não dirigidos (*undirected bipartite graphs*). Onde, as variáveis seriam uma partição e as restrições e funções objetivo a outra. Desta forma, não haveria arestas dentro das mesmas partições (variáveis com variáveis, nem de restrições com restrições ou funções objetivo), mas apenas arestas entre partições (variáveis com restrições ou funções objetivo). Lembrando-se que, assim como para as representações anteriores não importa se as variáveis são de projeto, de acoplamento ou de estado, nem se as restrições e

²Hiper-grafo é estrutura similar a um grafo simples, mas permite que uma mesma aresta interligue mais de um vértice ao mesmo tempo.

funções objetivo são funções, equações, inequações, simulações, superfícies de resposta, experimentos, ou outro tipo de relações.

Formalmente, pode-se definir um grafo bipartido G através da Eq. 5.8 onde tem-se:

$$\begin{aligned} G &= (V, E) ; \text{ onde} \\ V &= V_{\text{variáveis}} \cup V_{\text{funções}}; V_{\text{variáveis}} \cap V_{\text{funções}} = \emptyset ; \text{ e} \\ E &= \{e_{ij} \mid w_{ij} \neq 0; \text{ se e somente se, a variável } i \text{ ocorre na função } j\} \end{aligned} \quad (5.8)$$

Onde, V é o conjunto de vértices formados pela união dos vértices da partição das variáveis ($V_{\text{variáveis}}$) com os vértices da partição das funções ($V_{\text{funções}}$), sendo que evidentemente a interseção destes conjuntos é vazia. E o conjunto de arestas E é composto por todas arestas e_{ij} entre os vértices de variáveis e os respectivos vértices de funções que elas aparecem.

No caso, da representação para o problema de decomposição, o grafo será não valorado (*unweighted graph*), isto é o valor ou peso (*weight*) de uma aresta existente será sempre 1 ($w_{ij} = 1$), e arestas não existentes serão representadas por 0 na matriz de adjacência A . A matriz A para qualquer grafo bipartido e não dirigido é simétrica podendo ser expressa conforme a Eq. 5.9:

$$A(\text{Eq.5.1}) = \begin{bmatrix} \mathbf{O} & B \\ B^T & \mathbf{O} \end{bmatrix} \quad (5.9)$$

Sendo, B com dimensão $n \times m$ definida pela Eq.5.10, e \mathbf{O} são matrizes de zeros (no caso da superior esquerda de dimensões $n \times m$, e a inferior esquerda $m \times n$. Onde, n é a quantidade de vértices na primeira partição, e m é o número de vértices na segunda partição. Como foi definido, a ordem dos vértices sempre será primeiro às variáveis seguidas pelas funções, e, assim, conseqüentemente n será o número de variáveis presentes no modelo e m o de funções.

$$B_{ij} = \begin{cases} w_{ij} & , \exists \text{ arestas entre } i \text{ e } j \\ 0 & , \text{ caso contrário} \end{cases} \quad (5.10)$$

Onde, w_{ij} é o peso (*weight*) ou valor da aresta entre i e j , no caso, $w_{ij} = 1$ sempre que houver uma aresta ligando i e j (zero caso contrário).

Utilizando o mesmo exemplo, a Fig. 5.1 apresenta o grafo bipartido obtido para o

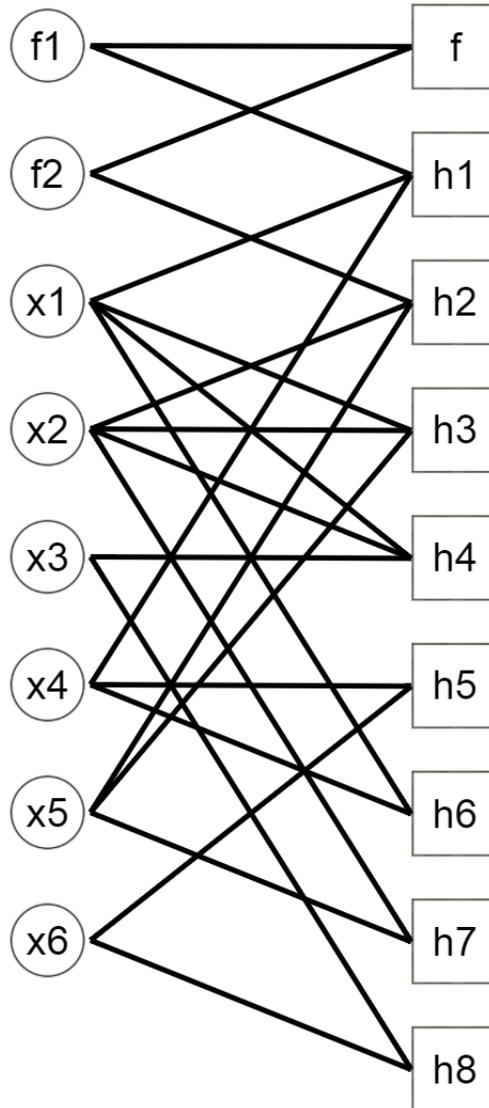


Figura 5.1: Grafo bipartido para o modelo da Eq.5.1

modelo da Eq. 5.1, onde, os vértices circulares são as variáveis e os vértices quadrados a função objetivo e restrições. A equação Eq. 5.11 mostra a matriz B que compõe a matriz de adjacência A referente ao grafo da Fig. 5.1.

$$B(\text{Eq.5.1}) = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (5.11)$$

Percebe-se que a matriz B é exatamente a matriz de ocorrências do modelo ($B = Occ$) e logicamente a matriz B^T é a representação matricial da FDT ($B^T = FDT$). Afinal na matriz de ocorrência o valor do elemento da i -ésima linha e j -ésima coluna é 1 (ou verdadeiro) se e somente se a i -ésima variável ocorrer na j -ésima função, sendo 0 caso não ocorra (como definido na Eq. 5.5). Sendo esta a mesma definição da existência ou não de aresta e_{ij} conforme a Eq. 5.8, que é refletida na Eq. 5.10 para definição dos valores dos elementos da matriz B . Portanto, a Eq. 5.9 para o problema de decomposição baseado em modelo pode ser re-escrita na Eq. 5.12.

$$A(\text{Eq.5.1}) = \begin{bmatrix} \mathbf{O} & B \\ B^T & \mathbf{O} \end{bmatrix} = \begin{bmatrix} \mathbf{O} & Occ \\ Occ^T & \mathbf{O} \end{bmatrix} = \begin{bmatrix} \mathbf{O} & Occ \\ FDT & \mathbf{O} \end{bmatrix} = \begin{bmatrix} \mathbf{O} & FDT^T \\ FDT & \mathbf{O} \end{bmatrix} \quad (5.12)$$

Onde, Occ e FDT são respectivamente a matriz de ocorrência e a tabela de dependência funcional para o modelo da Eq. 5.1.

Por fim, é importante informar que esta forma de representação foi inspirada em [75], onde não é aplicada ao problema de decomposição de sistemas. O trabalho citado utiliza o algoritmo de redução de dimensionalidade SVD para auxiliar na detecção de comunidades em grafos não direcionados. Para tanto, parte da matriz de adjacência do grafo (que é simétrica) e a transforma na matriz B , caso o grafo seja bipartido.

Ou seja, inspirado por [75] este trabalho utiliza o caminho inverso. A partir da matriz B , assumindo que seja igual a matriz de ocorrência ou ao transposto da FDT, obtem-se uma matriz de adjacência referente aos grafos bipartidos representativos dos modelos dos sistemas a serem decompostos.

5.2 Métodos de Decomposição Utilizados

5.2.1 Método de Referência

O presente trabalho tem como base os trabalhos de Sarkar *et al* [37] [9], os quais utilizaram de forma original a redução de dimensionalidade através do algoritmo SVD e a análise de agrupamento realizada pelo algoritmo K-means sobre os dados. A metodologia de decomposição do problema proposta em [37] é descrita abaixo:

1. Constrói a matriz de ocorrência a partir do modelo do problema;
2. Realiza a redução de dimensionalidade por meio do SVD da matriz de ocorrência;
3. Obtêm as representações das variáveis e das funções (simulações, etc) no espaço reduzido;
4. Executa o agrupamento sobre os dados reduzidos utilizando K-means e dissimilaridade de cosseno;
5. Reorganiza a matriz de ocorrência re-arranjando as linhas, associadas às variáveis, para cada um dos grupos obtidos, fazendo o mesmo para as funções, que são associadas às colunas;
6. Identifica as variáveis de ligação e os sub-problemas.

Para a escolha da quantidade de dimensões a serem utilizadas para definir o espaço reduzido a partir do SVD em [37] é sugerido que se inicie com duas dimensões e vá aumentando até se perceber que a matriz de ocorrência reduzida começou a retornar apenas os dados explícitos, sendo o número de dimensões máximo igual ao posto (*rank*) da matriz (devido as características da fatoração em valor singular).

Diferentemente dos métodos clássicos, em [37], considera-se o modelo do sistema como um conjunto de símbolos, a partir dos quais se obtém a matriz de ocorrência, aproveitando a interpretação do modelo como símbolos, e inspirando-se nas técnicas provindas da área de processamento de linguagem natural e de imagens digitais, aplica-se o algoritmo SVD de redução de dimensionalidade, transportando a matriz de ocorrência do espaço original para um espaço reduzido (de no máximo a dimensão do posto da matriz de ocorrência). Como o SVD é uma fatoração que captura tanto padrões nas amostras (no caso funções),

quanto nos atributos (no caso as variáveis), apenas com esta única fatoração e redução ambos os vetores reduzidos referentes as variáveis e as funções são representados no novo espaço.

Esta etapa tem por objetivo não apenas alocar as funções e variáveis no mesmo espaço, mas principalmente ressaltar as informações relevantes e remover as menos relevantes, que ao invés de contribuir para a identificação de relações significativas entre os elementos do modelo, podem "confundir" o algoritmo de aprendizado. Como descrito, esta estratégia foi originalmente proposta no trabalho citado [37] para a decomposição de sistemas. Distintamente da grande parte das aplicações de redução de dimensionalidade, onde se deseja reduzir o espaço característico perdendo o mínimo de informação relevante, a técnica apresentada visa justamente o contrário, isto é, o de eliminar as informações não relevantes de forma a evidenciar as relações semânticas mais significativas.

Após a redução, os vetores reduzidos das variáveis e funções são apresentados como entrada no formato de uma matriz de dados para o K-means, que utilizando a dissimilaridade de cosseno identifica as relações intrínsecas do modelo ressaltadas através do SVD. Por fim, com os agrupamentos de cada elemento a matriz de ocorrência é reescrita de forma a apresentar os elementos em ordem dos grupos identificados, isto é, nas primeiras linha e colunas serão dispostos as funções e variáveis, respectivamente, pertencentes ao primeiro grupo, seguidas das do segundo, e assim por diante.

5.2.2 Métodos Propostos

Seguindo os passos do método de referência, apresentado acima, o presente trabalho propõe o método a seguir com o objetivo de permitir a substituição do algoritmo K-means pelo método de Louvain, modificando, assim, a forma de tratar o problema que antes era visto como uma análise de agrupamento com k pré-definido e agora será abordado como uma detecção de comunidade. Além da mudança de técnica é importante notar que o K-means, por definir o número de sub-problemas *a priori*, particiona o grafo representativo do modelo, enquanto na detecção de comunidade pelo método de Louvain será permitido que sejam desvendados as estruturas de relações semânticas naturais, que formam as comunidades na rede de interações representativa do modelo do sistema a ser decomposto.

Para tanto, ao invés de se obter a representação vetorial dos elementos do modelo após a redução (como é feito quando o agrupamento é realizado através do K-means), a matriz

de ocorrências reduzida deve ser representada como um grafo, mais exatamente a matriz de adjacência de um grafo das similaridades de cosseno destes vetores reduzidos. Desta forma, a matriz de distância é convertida em um grafo bipartido, conforme apresentado na Seção 5.1.0.3, obtendo-se um grafo bipartido com as variáveis em uma partição e as funções na outra. Nota-se que o cálculo dos cossenos é necessário pois no K-means ele é realizado internamente a cada interação para comparar os pontos com os centroides. Entretanto, no método de Louvain é necessário que as arestas do grafo já tenham a informação da proximidade entre os elementos do modelo, este passo acaba sendo uma economia de processamento, pois só será necessário realizá-lo uma única vez, enquanto no K-means as dissimilaridades são recalculadas em todas as alterações. Outro ponto importante, é que no método proposto utiliza-se a similaridade, pois o peso das arestas em um grafo indicam o quão forte é a ligação entre dois vértices, ou seja, quanto maior o peso mais próximo são os elementos.

1. Constrói a matriz de ocorrência a partir do modelo do problemas;
2. Realiza a redução de dimensionalidade por meio do SVD da matriz de ocorrência;
3. Obtêm as representações das variáveis e das funções (simulações, etc) no espaço reduzido;
4. Calcula a similaridade de cosseno entre cada ponto expresso no espaço reduzido;
5. Monta uma representação de grafo bipartido onde os vértices serão os elementos do modelo e o peso das arestas serão as similaridade calculadas no passo anterior;
6. Executa o método de Louvain para identificar as comunidades de elementos;
7. Define cada comunidade como um sub-problema;
8. Identifica as variáveis de ligação a partir das interconexões entre as comunidades.

Em relação ao passo 8, uma forma de identificar as variáveis de ligação é através da mesma estratégia de re-ordenação da matriz de ocorrência utilizada por [37]. Outra é através da visualização da rede, onde as variáveis de ligação serão as variáveis de projeto com aresta à funções externas a sua comunidade.

Além do método proposto acima, também foi estudada a aplicação direta do método de Louvain sobre a matriz de ocorrência, realizando apenas a conversão desta para um

grafo bipartido, sem a utilização de qualquer método de redução de dimensionalidade. Neste caso, o grafo será um grafo não valorado, visto que suas arestas indicaram apenas a ocorrência ou não da variável na função. Isto não seria possível de ser realizado com o K-means, visto não haver como comparar variáveis e funções antes da redução de dimensionalidade via SVD.

1. Constrói a matriz de ocorrência a partir do modelo do problemas;
2. Monta uma representação de grafo bipartido não valorado onde os vértices serão os elementos do modelo;
3. Executa o método de Louvain para identificar as comunidades de elementos;
4. Define cada comunidade como um sub-problema;
5. Identifica as variáveis de ligação a partir das interconexões entre as comunidades.

No próximo capítulo, as estratégias propostas para a decomposição de problemas serão avaliadas, visando verificar seu potencial principalmente em relação ao método de referencia.

6 Aplicação e Experimentos

Para a validação dos métodos propostos eles foram aplicados a problemas de otimização de projeto (ODPs) em problemas de engenharia disponíveis na literatura, tanto na de decomposição de projeto, quanto diretamente na de otimização de projetos (*design optimization*). Além da descrição dos problemas, esta seção, também, relata os resultados obtidos pelas decomposições realizadas através do método de referência [37] e dos métodos propostos neste trabalho para cada um dos problemas de projeto em engenharia descritos, e ainda, são apresentados os valores obtidos para as medidas de qualidade da decomposição. Sendo que, para métodos parametrizados, SVD e o K-means, serão apresentados somente os resultados para os parâmetros com melhores resultados e os resultados relativos a parâmetros de interesse comparativos, mesmo que não sejam os melhores.

Por fim, os resultados das aplicações e experimentos deste capítulo serão discutidos perante as medidas de qualidade e comparativamente entre si na Seção 6.7.

6.1 Projeto 1 - Modelo de Hock-Schittkowski Modificado

O primeiro projeto a ser decomposto será o proposto por Hock e Schittkowski [83] *apud* [11] modificado por Michelena e Papalambros em [11], e novamente por Sarkar *et al* em [37], o mesmo utilizado como exemplo das representações na Seção 5.1 no Cap. 5. A

Tabela 6.1: Projeto 1 - Tabela de ocorrências

	f	h_1	h_2	h_3	h_4	h_5	h_6	h_7	h_8
x_1		■		■	■		■		
x_2			■	■	■			■	
x_3				■					■
x_4		■				■	■		
x_5			■	■				■	
x_6						■			■
f_1	■	■							
f_2	■		■						

Eq. 6.1 descreve o modelo do ODP conforme exposto por [37]:

$$\begin{aligned}
 \min \quad & f = f_1 + f_2 \\
 \text{s.a.} \quad & h1 : f1 = x_1 + e^{x_1 x_4} \\
 & h2 : f2 = 2x_2 + 4x_5 \\
 & h3 : x_1 + 2x_2 + 5x_5 - 6 = 0 \\
 & h4 : x_1 + x_2 + x_3 - 3 = 0 \\
 & h5 : x_4 + x_6 - 2 = 0 \\
 & h6 : x_1 + x_4 - 1 = 0 \\
 & h7 : x_2 + x_5 - 2 = 0 \\
 & h8 : x_3 + x_6 - 2 = 0 \\
 & \mathbf{x} \in \mathbb{R}^6
 \end{aligned} \tag{6.1}$$

Extraindo as ocorrências através de processamento simbólico dos elementos do modelo obtém-se a Tab. 6.1, a matriz de ocorrência na Eq. 6.2 e o gráfico bipartido na Fig. 6.1.

$$\text{Occ}(Eq.6.1) = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{6.2}$$

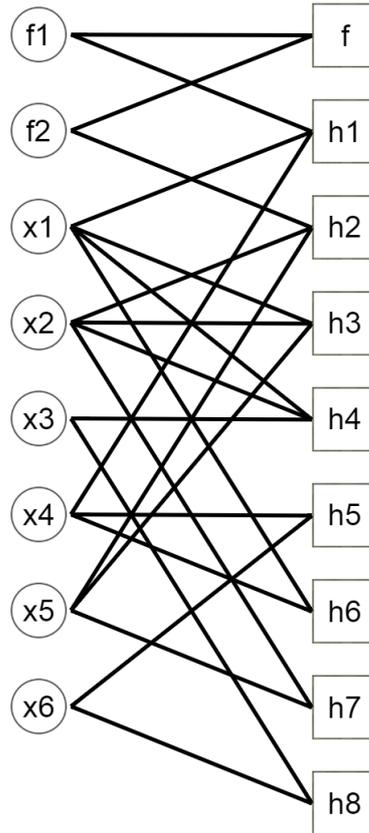


Figura 6.1: Grafo bipartido para o Projeto 1

6.1.1 Escolha das Dimensões do Espaço Reduzido

Segundo [37], a escolha da quantidade de dimensões a serem retidas, p , para o problema de decomposição de projetos, diferentemente do problema de redução de dimensionalidade de base de dados tradicional ao KDD, visa a eliminação de relações não relevantes. Propõe, então, uma heurística de seleção que consiste em escolher um valor p entre dois e a dimensão onde se percebe que as estruturas das matrizes de ocorrência começam a se aproximar significativamente, de forma a evitar uma aproximação exacerbada que não remova as informações indesejadas.

Ao invés do processo iterativo do artigo de referência [37], neste trabalho, a identificação da aproximação é realizada aplicando a mesma heurística sobre a tabela das proporções acumuladas de aproximação da representação reduzida perante a original (conforme descrito no Cap. 3), a qual é apresentada pela Tab. 6.2 para o Projeto 1. Isto foi validado sobre os ODPs (Projetos 1, 2 e 3) retirados de [37], onde tanto a aplicação direta deste trabalho quanto o processo iterativo, ambos baseados na mesma heurística, obtiveram os mesmos valores de p .

Tabela 6.2: Projeto 1 - Proporção acumulada de aproximação das p reduções de dimensionalidade através do SVD

p	Aproximação
1	0.3907
2	0.6246
3	0.7635
4	0.8790
5	0.9648
6	0.9869
7	0.9966
8	1.0000

Tabela 6.3: Projeto 1 - Sub-problemas decompostos pelo método de referência com SVD(2) e K-means(2)

Sub-problema	Elementos do sub-problema									
1	' x1'	' x3'	' x4'	' x6'	' f1'	' f'	' h1'	' h5'	' h6'	' h8'
2	' x2'	' x5'	' f2'	' h2'	' h3'	' h4'	' h7'			

A Fig. 6.2 apresenta os elementos do modelo (Eq. 6.1) no espaço bidimensional obtido através dos SVD sobre a matriz de ocorrência (Eq. 6.2).

6.1.2 Resultados

6.1.2.1 Método Referência com Dois Agrupamentos

Realizando a decomposição através do método de referência com uma redução para um espaço bidimensional e agrupamento em dois *clusters* ($k = 2$), obteve-se os dois sub-problemas da Tab. 6.3, que podem ser representados pela matriz de adjacência na Eq. 6.3, e ao se reordenar a matriz de ocorrência na Tab. 6.4, identifica-se as ligações realizadas pelas variáveis de projeto x_1 e x_3 e pela variável de acoplamento f_2 . Portanto, esta decomposição apresenta duas variáveis de ligação, x_1 e x_3 . Pois, não se considera variáveis de acoplamento como sendo de ligação, visto que não adicionam complexidade a tarefa de integração, já que a combinação delas ocorre de forma automática, sendo utilizadas como constantes no cálculo da função objetivo f , e obtidas pelo cálculo da respectiva função expressa, no caso de f_2 pela restrição h_2 .

$$A_{SP}(\text{SVD}_2, 2\text{-means}) = \begin{bmatrix} 10 & 4 \\ 4 & 8 \end{bmatrix} \quad (6.3)$$

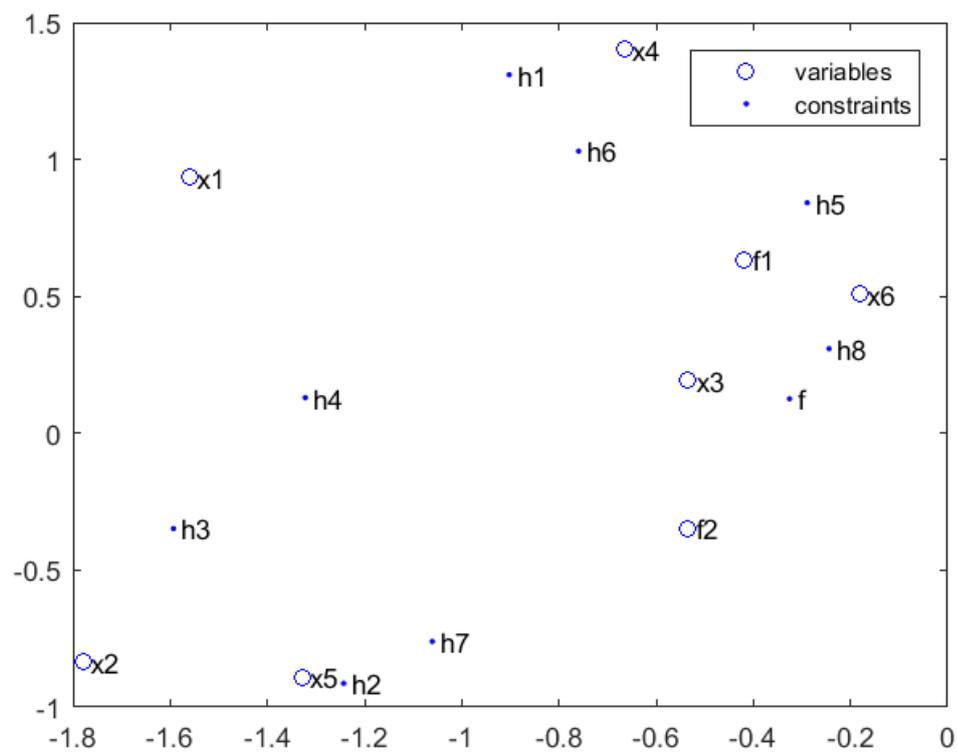


Figura 6.2: Projeto 1 - Projeção dos elementos do modelo no espaço reduzido bidimensional obtido pelo SVD sobre a matriz de ocorrência

Tabela 6.4: Projeto 1 - Tabela de ocorrências reordenada ressaltando os agrupamentos (colorido) e as ligações (cinza), para método SVD(2) + 2-means

	f	h_1	h_5	h_6	h_8	h_2	h_3	h_4	h_7
x_1	0	1	0	1	0	0	1	1	0
x_3	0	0	0	0	1	0	0	1	0
x_4	0	1	1	1	0	0	0	0	0
x_6	0	0	1	0	1	0	0	0	0
f_1	1	1	0	0	0	0	0	0	0
x_2	0	0	0	0	0	1	1	1	1
x_5	0	0	0	0	0	1	1	0	1
f_2	1	0	0	0	0	1	0	0	0

O resultado do agrupamento em dois *clusters* também pode ser observado sobre a projeção dos elementos no espaço reduzido pelo SVD, conforme apresentado na Fig. 6.3, onde a propriedade da dissimilaridade de cosseno não ser influenciada pela magnitude dos vetores relativos a cada elemento pode ser observada.

6.1.2.2 Método Referência com Três Agrupamentos

Por sua vez, a decomposição através do método de referência com uma redução para um espaço bidimensional e agrupamento em três *clusters* ($k = 3$), obteve-se os dois sub-problemas da Tab. 6.5, que podem ser representados pela matriz de adjacência na Eq. 6.4, e ao se reordenar a matriz de ocorrência na Tab. 6.4, identifica-se as ligações realizadas pelas variáveis de projeto x_1 e x_3 e pelas variáveis de acoplamento f_1 e f_2 . Portanto, esta decomposição apresenta duas variáveis de ligação, x_1 e x_3 . Ainda, na reordenação nota-se que mesmo f , x_1 e x_2 não compartilhando qualquer aresta estes elementos foram posicionados no mesmo agrupamento. Isto se deve a relações não explícitas na forma de matriz de ocorrência, mas extraída pela redução de dimensionalidade e revelada pelo agrupamento. Nota-se, ainda, que estas variáveis que se relacionam com f são as de ligação, o que é esperado, visto que a função objetivo é naturalmente uma função de ligação entre os dois agrupamentos (como é perceptível pelas ligações com as variáveis de acoplamento f_1 e f_2).

$$A_{SP}(\text{SVD}_2, 3\text{-means}) = \begin{bmatrix} 2 & 4 & 3 \\ 4 & 6 & 0 \\ 3 & 0 & 7 \end{bmatrix} \quad (6.4)$$

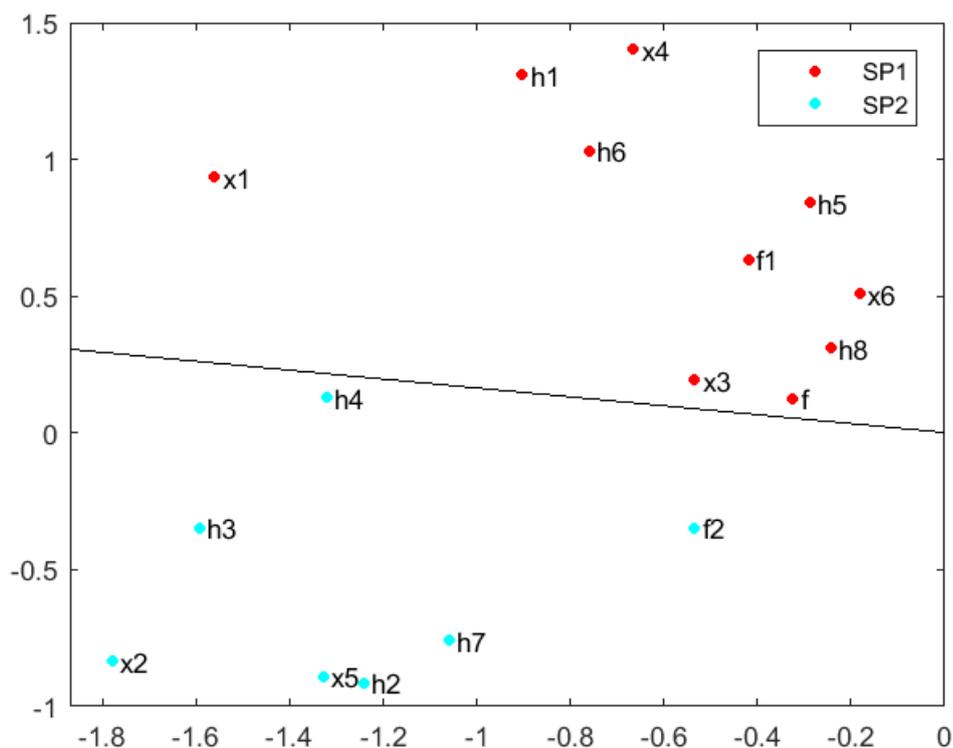


Figura 6.3: Projeto 1 - Projeção do resultado do agrupamento com $k = 2$ no espaço reduzido bidimensional obtido pelo SVD

Tabela 6.5: Projeto 1 - Sub-problemas decompostos pelo método de referência com SVD(3) e K-means(3)

Sub-problema	Elementos do sub-problema						
1	'x1'	'x3'	'f'	'h4'			
2	'x4'	'x6'	'f1'	'h1'	'h5'	'h6'	'h8'
3	'x2'	'x5'	'f2'	'h2'	'h3'	'h7'	

Tabela 6.6: Projeto 1 - Tabela de ocorrências reordenada ressaltando os agrupamentos (coloridas) e as ligações (cinza), para método SVD(2) + 3-means

	f	h_4	h_1	h_5	h_6	h_8	h_2	h_3	h_7
x_1	0	1	1	0	1	0	0	1	0
x_3	0	1	0	0	0	1	0	0	0
x_4	0	0	1	1	1	0	0	0	0
x_6	0	0	0	1	0	1	0	0	0
f_1	1	0	1	0	0	0	0	0	0
x_2	0	1	0	0	0	0	1	1	1
x_5	0	0	0	0	0	0	1	1	1
f_2	1	0	0	0	0	0	1	0	0

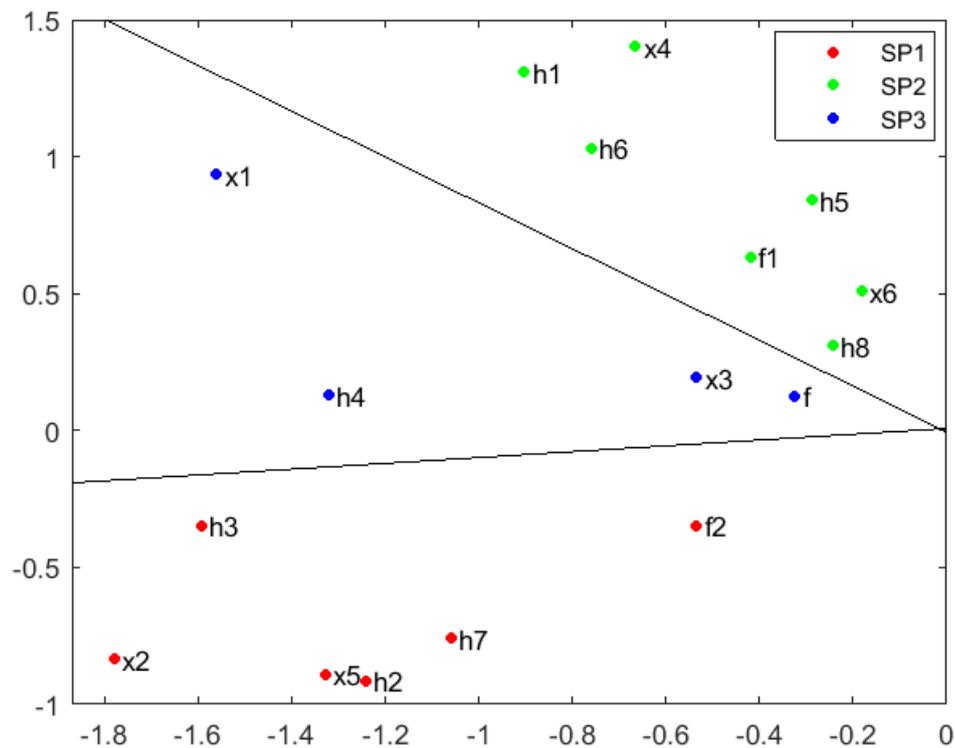


Figura 6.4: Projeto 1 - Projeção do resultado do agrupamento com $k = 3$ no espaço reduzido bidimensional obtido pelo SVD

Da mesma forma que o resultado do agrupamento em três *clusters* também pode ser observado sobre a projeção dos elementos no espaço reduzido pelo SVD, conforme apresentado na Fig. 6.4, onde, novamente a propriedade da dissimilaridade de cosseno não ser influenciada pela magnitude dos vetores relativos a cada elemento pode ser observada.

6.1.2.3 Método Proposto com SVD e Louvain

A decomposição obtida do método proposto por este trabalho com espaço reduzido bidimensional e o método de Louvain para detectar os sub-problemas é exibido na Tab. 6.7, a matriz de adjacência da rede das comunidades resultante é apresentada na Eq. 6.5, e as comunidades detectadas são destacadas no grafo em 6.5. Nota-se que a comunidade verde e azul são conectadas apenas através das variáveis de projeto, x_1 e x_3 , e da variável de acoplamento f_2 . Portanto, a decomposição apresenta duas variáveis de ligação. Pois, variáveis de acoplamento não são contabilizadas, visto que sua integração é automática como componente explícita da função objetivo, obtida no sub-problema através do cálculo de uma restrição (no caso h_2), desta forma, não implica em qualquer complexidade para

Tabela 6.7: Projeto 1 - Sub-problemas decompostos pelo método proposto com SVD(2) e Louvain

Sub-problema	Elementos do sub-problema									
1	' x1'	' x3'	' x4'	' x6'	' f1'	' f'	' h1'	' h5'	' h6'	' h8'
2	' x2'	' x5'	' f2'	' h2'	' h3'	' h4'	' h7'			

Tabela 6.8: Projeto 1 - Tabela de ocorrências reordenada ressaltando as comunidades (coloridas) e as ligações (cinza), para método SVD(2) + Louvain

	f	h_1	h_5	h_6	h_8	h_2	h_3	h_4	h_7
x_1	0	1	0	1	0	0	1	1	0
x_3	0	0	0	0	1	0	0	1	0
x_4	0	1	1	1	0	0	0	0	0
x_6	0	0	1	0	1	0	0	0	0
f_1	1	1	0	0	0	0	0	0	0
x_2	0	0	0	0	0	1	1	1	1
x_5	0	0	0	0	0	1	1	0	1
f_2	1	0	0	0	0	1	0	0	0

a tarefa de integração.

$$A_{SP}(\text{SVD}_2, \text{Louvain}) = \begin{bmatrix} 10 & 4 \\ 4 & 8 \end{bmatrix} \quad (6.5)$$

Além da representação do grafo colorido com as comunidades detectadas, também, pode-se visualizar os sub-problemas através da mesma técnica de reordenação da matriz de ocorrência que foi utilizada para os métodos com o K-means. Isto é apresentado para este método na Tab. 6.8. Observa-se que os resultados para este método e o de referência com $k = 2$ obtiveram exatamente os mesmos resultados.

Por fim o resultado do sub-problemas detectados pelo método proposto também pode ser observado sobre a projeção dos elementos no espaço reduzido pelo SVD, conforme apresentado na Fig. 6.6, igualmente realizado para o método de referência.

6.1.2.4 Método Proposto com Louvain e sem RD

No caso do método proposto que não realiza a etapa de redução de dimensionalidade, partindo diretamente para o método de Louvain, os sub-problemas da decomposição são exibidos na Tab. 6.9, e a matriz de adjacência da rede das comunidades resultante é apresentada na Eq. 6.6, as comunidades detectadas são destacadas no grafo em 6.7.

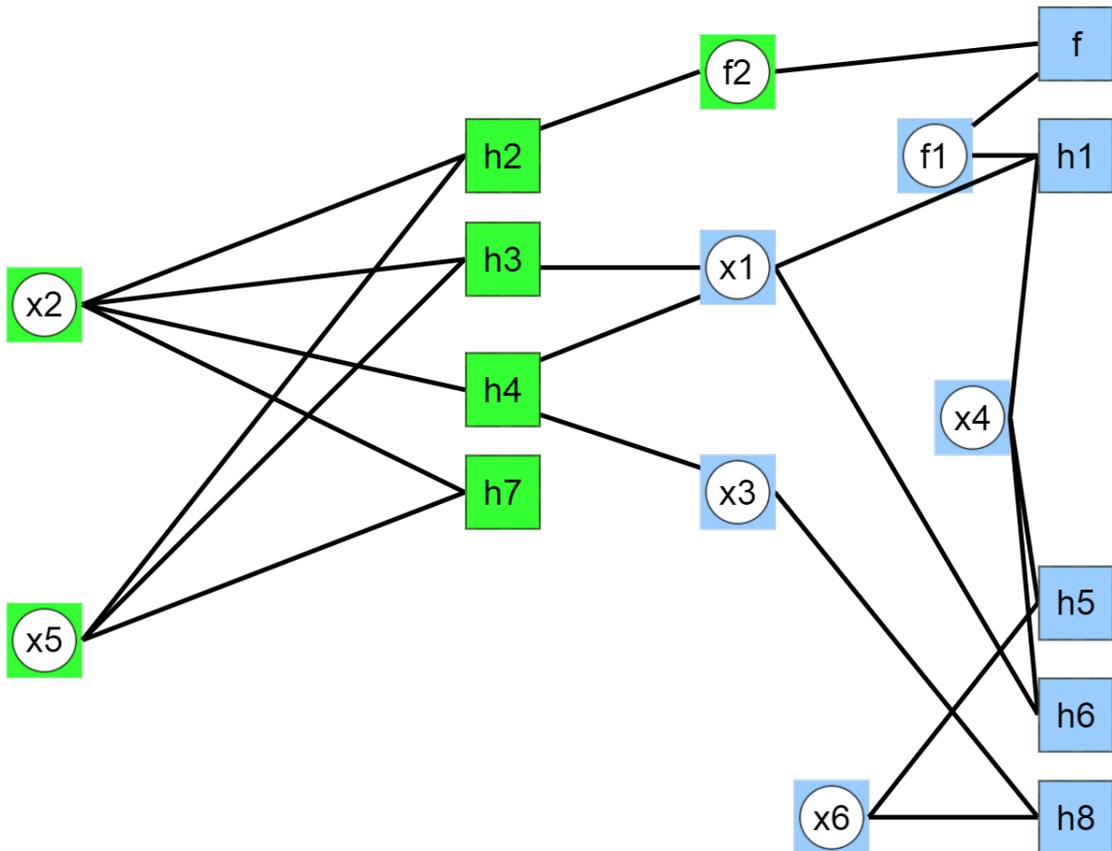


Figura 6.5: Projeto 1 - Comunidades detectadas pelo método proposto com SVD(2) + Louvain

Tabela 6.9: Projeto 1 - Sub-problemas decompostos pelo método proposto sem RD e com Louvain

Sub-problema	Elementos do sub-problema								
1	' x2'	' x5'	' f1'	' f2'	' f'	' h2'	' h3'	' h7'	
2	' x3'	' x6'	' h4'	' h5'	' h8'				
3	' x1'	' x4'	' h1'	' h6'					

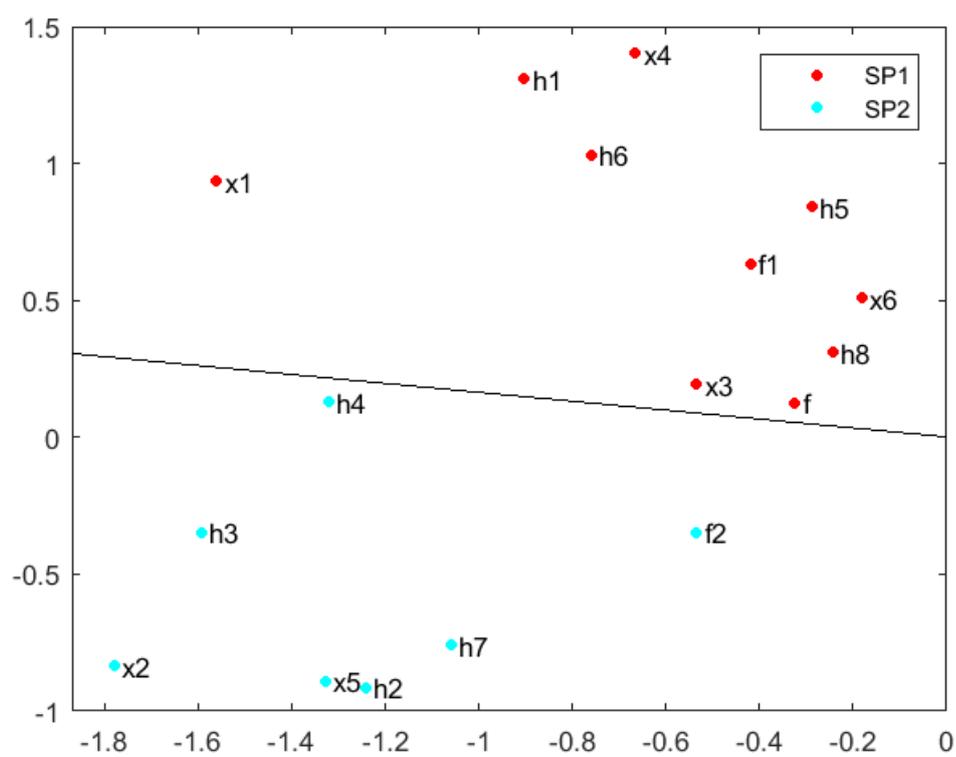


Figura 6.6: Projeto 1 - Projeção do resultado da detecção de comunidades no espaço reduzido bidimensional obtido pelo SVD

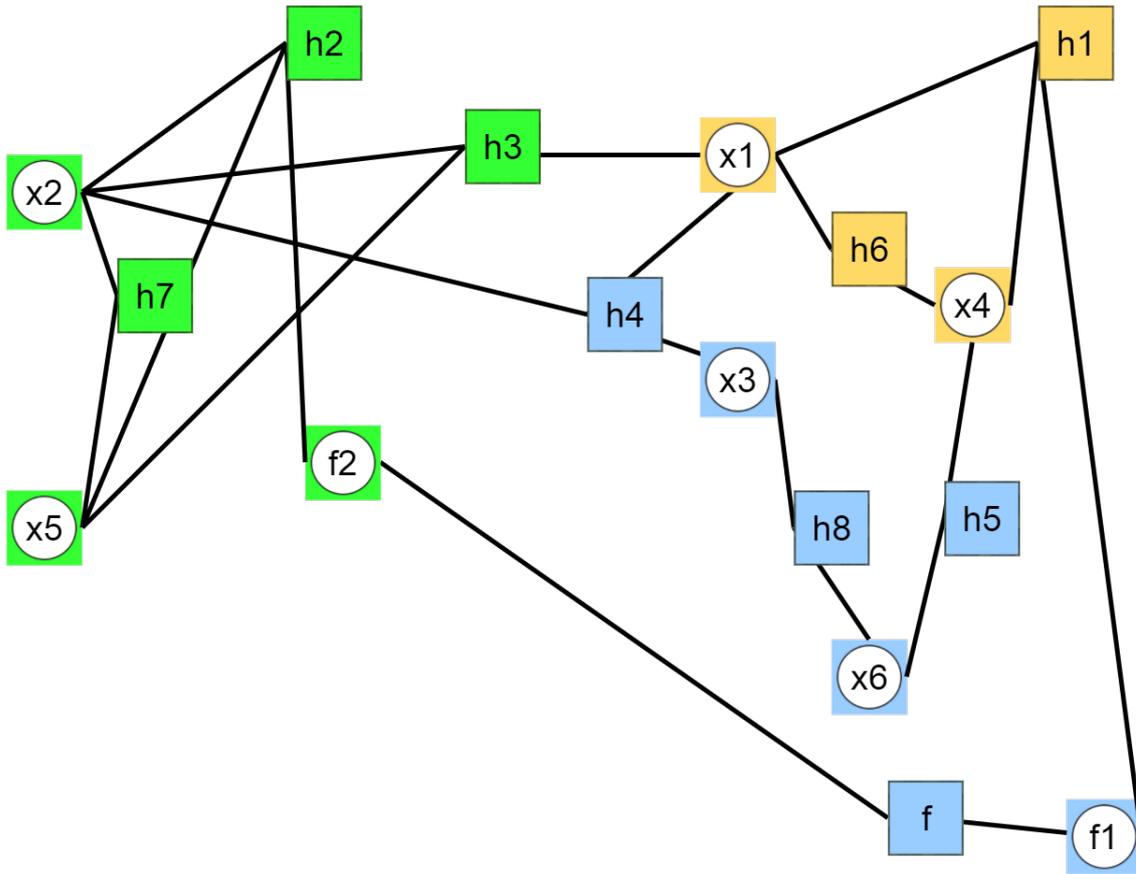


Figura 6.7: Projeto 1 - Comunidades detectadas pelo método proposto sem RD + Louvain

$$A_{SP}(\text{Sem RD, Louvain}) = \begin{bmatrix} 9 & 1 & 2 \\ 1 & 4 & 2 \\ 2 & 2 & 4 \end{bmatrix} \quad (6.6)$$

6.1.2.5 Medidas de Qualidade

As medidas de qualidade para as decomposições obtidas pelos diferentes métodos aplicados ao Projeto 1 são apresentadas comparativamente nas Tabelas: 6.10, 6.11 e 6.12. Vale ressaltar que, exclusivamente para o cálculo das medidas baseada em grafos, foi-se considerado que eram dígrafos simétricos, ou seja, cada aresta do grafo original não direcionado é contada duas vezes, de forma bidirecional, nunca havendo a ocorrência de uma aresta sem a existência de outra ligando os mesmo nós com o mesmo peso no sentido contrário. Desta forma a representação continua sendo pela mesma matriz de adjacência

Tabela 6.10: Projeto 1 - Quantidade de variáveis de ligação e sub-problemas obtidos

Método	Variáveis de Ligação	Sub-problemas
SVD + 2-means	2	2
SVD + 3-means	2	3
SVD + Louvain	2	2
Sem RD + Louvain	3	3

Tabela 6.11: Projeto 1 - Medidas de qualidade das comunidades/agrupamentos (a)

Método	Modularidade	<i>Cut Size</i>	BIC	AIC
SVD + 2-means	0.3140	8	100.5604	72.2312
SVD + 3-means	0.3378	14	148.5288	106.0349
SVD + Louvain	0.3140	8	100.5604	72.2312
Sem RD + Louvain	0.4081	10	148.5040	106.0101

simétrica de um grafo não direcionado, havendo apenas alteração na implementação das medidas, e os resultados destas se mantêm equivalentes mas com valores dobrados.

6.2 Projeto 2 - Cilindro Hidráulico Mono-objetivo

Tanto o Projeto 2 quanto o Projeto 3 são relativos a um cilindro hidráulico (*hydraulic cylinder design* - HCD) o qual pode ser formulado e solucionado de diversas formas. Neste trabalho, considera-se duas formulações do problema: uma mono-objetiva, Projeto 2 [85] *apud* [9], e outra multi-objetiva, Projeto 3 (Seção 6.3) [86] *apud* [9]. A Fig. 6.8 ilustra o problema de obtenção do projeto ótimo para o cilindro hidráulico modelado pela Eq. 6.7 com um único objetivo.

Tabela 6.12: Projeto 1 - Medidas de qualidade das comunidades/agrupamentos (b)

Método	Silhueta Empírica	Silhueta Igualitária
SVD + 2-means	0.1220	0.1349
SVD + 3-means	0.0512	0.0387
SVD + Louvain	0.1220	0.1348
Sem RD + Louvain	0.0655	0.0666

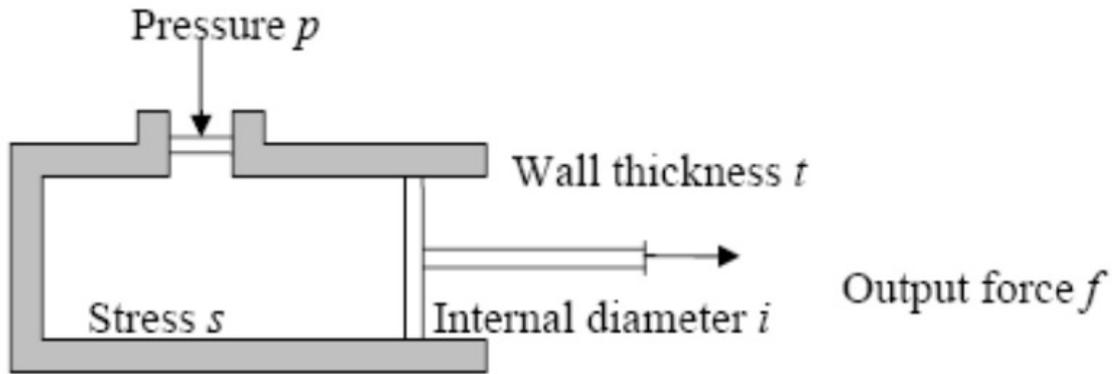


Figura 6.8: Projeto do cilindro hidráulico mono-objetivo (Retirado de [9]).

$$\begin{aligned}
 \min \quad & obj = i + 2t \\
 \text{s.a.} \quad & g_1 : t - T \geq 0 \\
 & g_2 : f - F \geq 0 \\
 & g_3 : p - P \leq 0 \\
 & g_4 : s - S \leq 0 \\
 & h_1 : f = \frac{\pi}{4} i^2 p \\
 & h_2 : s = \frac{ip}{2t} \\
 & \mathbf{x} \in \mathfrak{R}^5
 \end{aligned} \tag{6.7}$$

O modelo de ODP definido na Eq. 6.7 contém as seguintes variáveis de projeto:

- i - Diâmetro interno (*Internal diameter*);
- t - Espessura da parede (*Wall thickness*);
- f - Força de saída (*Output force*);
- s - Tensão (*Stress*);
- p - Pressão (*Pressure*).

E os seguintes parâmetros de projeto:

- T - Espessura mínima da parede;
- F - Força mínima de saída;
- S - Tensão máxima;
- P - Pressão máxima.

Tabela 6.13: Projeto 2 - Proporção acumulada de aproximação das p reduções de dimensionalidade através do SVD

p	Aproximação
1	0.4306
2	0.6145
3	0.7511
4	0.8687
5	0.9489
6	0.9801
7	1.0000

Tabela 6.14: Projeto 2 - Sub-problemas decompostos pelo método de referência com SVD(2) e K-means(2)

Sub-problema	Elementos do sub-problema								
1	' i'	' t'	' s'	' T'	' S'	' obj'	' g1'	' g4'	' h2'
2	' f'	' p'	' F'	' P'	' g2'	' g3'	' h1'		

6.2.1 Escolha das Dimensões do Espaço Reduzido

Para a escolha da quantidade de dimensões retidas pelo SVD, também observou-se a possibilidade de utilização direta da proporção de aproximação, ao invés da proporção acumulada de aproximação, sendo ambas equivalentes. A fim de ilustrar este fato, para o Projeto 2, a Fig. 6.9 apresenta o *scree plot* das proporções de aproximação, enquanto a Tab. 6.13 exhibe as proporções acumuladas de aproximação (mais facilmente interpretadas quando apresentadas no formato numérico do que a proporção não acumulada).

Analisando o gráfico e a tabela com as considerações sobre a escolha do número de dimensões realizadas para o Projeto 2, definiu-se utilizar $p = 2$, devido ser um valor que ainda que não se aproxime tanto da matriz original, também não se distancia tanto (quanto $p = 1$) de forma a perder informações relevantes; e claramente está posicionado no "cotovelo" da curva de decrescimento dos percentuais de aproximação.

6.2.2 Resultados

6.2.2.1 Método de Referência

Os resultados obtidos para decomposição do modelo para o Projeto 2 através do método de referência com $k = 2$ são apresentados na Tab. 6.14. Enquanto a Eq. 6.8 a matriz de adjacência entre os dois sub-problemas obtidos.

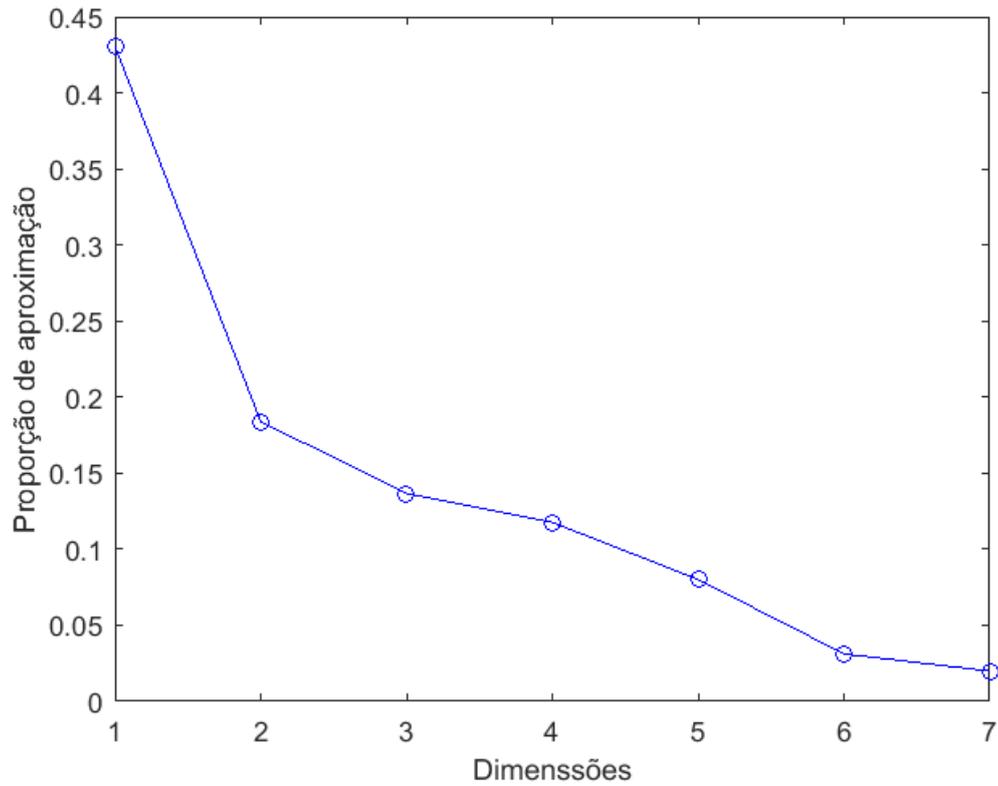


Figura 6.9: Projeto 2 - *Scree plot* das proporções de aproximação

Tabela 6.15: Projeto 2 - Sub-problemas decompostos pelo método proposto com SVD(2) e Louvain

Sub-problema	Elementos do sub-problema
1	'i' 't' 's' 'T' 'S' 'obj' 'g1' 'g4' 'h2'
2	'f' 'p' 'F' 'P' 'g2' 'g3' 'h1'

$$A_{SP}(SVD_2, 2\text{-means}) = \begin{bmatrix} 9 & 2 \\ 2 & 6 \end{bmatrix} \quad (6.8)$$

6.2.2.2 Método Proposto com SVD e Louvain

Os dois sub-problemas obtidos decompondo o modelo para o Projeto 2 pelo método proposto 1 são descritos na Tab. 6.15. A matriz de adjacência entre as comunidades encontradas, isto é entre os sub-problemas obtidos, é exposta pela Eq. 6.9.

$$A_{SP}(SVD_2, \text{Louvain}) = \begin{bmatrix} 9 & 2 \\ 2 & 6 \end{bmatrix} \quad (6.9)$$

Tabela 6.16: Projeto 2 - Sub-problemas decompostos pelo método proposto sem RD e com Louvain

Sub-problema	Elementos do sub-problema			
1	' s'	' S'	' g4'	' h2'
2	' i'	' obj'	' h1'	
3	' t'	' T'	' g1'	
4	' f'	' F'	' g2'	
5	' p'	' P'	' g3'	

Tabela 6.17: Projeto 2 - Quantidade de sub-problemas obtidos

Método	Sub-problemas
SVD + 2-means	2
SVD + Louvain	2
Sem RD + Louvain	5

6.2.2.3 Método Proposto com Louvain e sem RD

O método proposto 2 encontrou as cinco comunidades expostos na Tab. 6.16, sendo a matriz de adjacência entre elas apresentada pela Eq. 6.10.

$$A_{SP}(\text{Sem RD, Louvain}) = \begin{bmatrix} 3 & 1 & 1 & 0 & 1 \\ 1 & 2 & 1 & 1 & 1 \\ 1 & 1 & 2 & 0 & 0 \\ 0 & 1 & 0 & 2 & 0 \\ 1 & 1 & 0 & 0 & 2 \end{bmatrix} \quad (6.10)$$

6.2.2.4 Medidas de Qualidade

As medidas de qualidade para as decomposições obtidas pelos diferentes métodos aplicados ao Projeto 2 são apresentadas comparativamente nas Tabelas: 6.17, 6.18 e 6.19.

Tabela 6.18: Projeto 2 - Medidas de qualidade das comunidades/agrupamentos (a)

Método	Modularidade	<i>Cut Size</i>	BIC	AIC
SVD + 2-means	0.3668	4	92.9588	68.2359
SVD + Louvain	0.3668	4	92.9588	68.2359
Sem RD + Louvain	0.4377	12	225.2118	163.4047

Tabela 6.19: Projeto 2 - Medidas de qualidade das comunidades/agrupamentos (b)

Método	Silhueta Empírica	Silhueta Iguatária
SVD + 2-means	0.0777	0.0770
SVD + Louvain	0.0777	0.0770
Sem RD + Louvain	0.0570	0.0589

6.3 Projeto 3 - Cilindro Hidráulico Multi-objetivo

Por sua vez, o problema de projeto do cilindro hidráulico visto como um problema mono-objetivo no Projeto 2 (Seção 6.2), aqui será abordado através de uma formulação multi-objetiva modelada pela Eq. 6.11 [86] *apud* [9].

$$\begin{aligned}
 \min \quad & \text{obj} = Z = z_1 w_1 + z_2 w_2 + z_3 w_3 \\
 \text{s.a.} \quad & z_1 = \pi(it + t^2) \\
 & z_2 = \frac{s}{S} \\
 & z_3 = \frac{p}{P} \\
 & g_1 : t - T \geq 0 \\
 & g_2 : f - F \geq 0 \\
 & g_3 : p - P \leq 0 \\
 & g_4 : s - S \leq 0 \\
 & h_1 : f = \frac{\pi}{4} i^2 p \\
 & h_2 : s = \frac{p}{E} \left(\frac{i}{2t} + 0,6 \right) \\
 & \mathbf{x} \in \mathfrak{R}^5
 \end{aligned} \tag{6.11}$$

O modelo de ODP definido na Eq. 6.11 é composto por três objetivos, onde Z é a combinação aditiva dos três objetivos (ou sub-objetivos) z_1, z_2, z_3 , e pelas variáveis de projeto abaixo:

- i - Diâmetro interno (*Internal diameter*);
- t - Espessura da parede (*Wall thickness*);
- f - Força de saída (*Output force*);
- s - Tensão *Stress*;
- p - Pressão (*Pressure*);

Tabela 6.20: Projeto 3 - Sub-problemas decompostos pelo método de referência com SVD(3) e K-means(3)

Sub-problema	Elementos do sub-problema										
1	' f'	' p'	' F'	' P'	' h5'	' g2'	' g3'	' h1'			
2	' i'	' t'	' s'	' T'	' S'	' E'	' h3'	' h4'	' g1'	' g4'	' h2'
3	' Z'	' z1'	' z2'	' z3'	' w1'	' w2'	' w3'	' obj'			

- w_1 - Peso da contribuição do objetivo z_1 na composição de objetivos Z ;
- w_2 - Peso da contribuição do objetivo z_2 na composição de objetivos Z ;
- w_3 - Peso da contribuição do objetivo z_3 na composição de objetivos Z .

E os seguintes parâmetros de projeto:

- T - Espessura mínima da parede;
- F - Força mínima de saída;
- S - Tensão máxima;
- P - Pressão máxima;
- E - Eficiência de ligação (*Join efficiency*).

6.3.1 Escolha das Dimensões do Espaço Reduzido

Através da análise do *scree plot* (Fig.6.10) para a definição do número de dimensões, observa-se que o valor $p = 3$, com proporção de aproximação acumulada igual a 0,6533, se apresenta como uma escolha adequada conforme a mesma técnica de escolha utilizada no Projeto 2.

6.3.2 Resultados

6.3.2.1 Método de Referência

Os agrupamentos, sub-problemas, encontrados pelo método de referência para o Projeto 3 são apresentados na Tab. 6.20, e matriz de adjacência entre os eles pela Eq. 6.12.

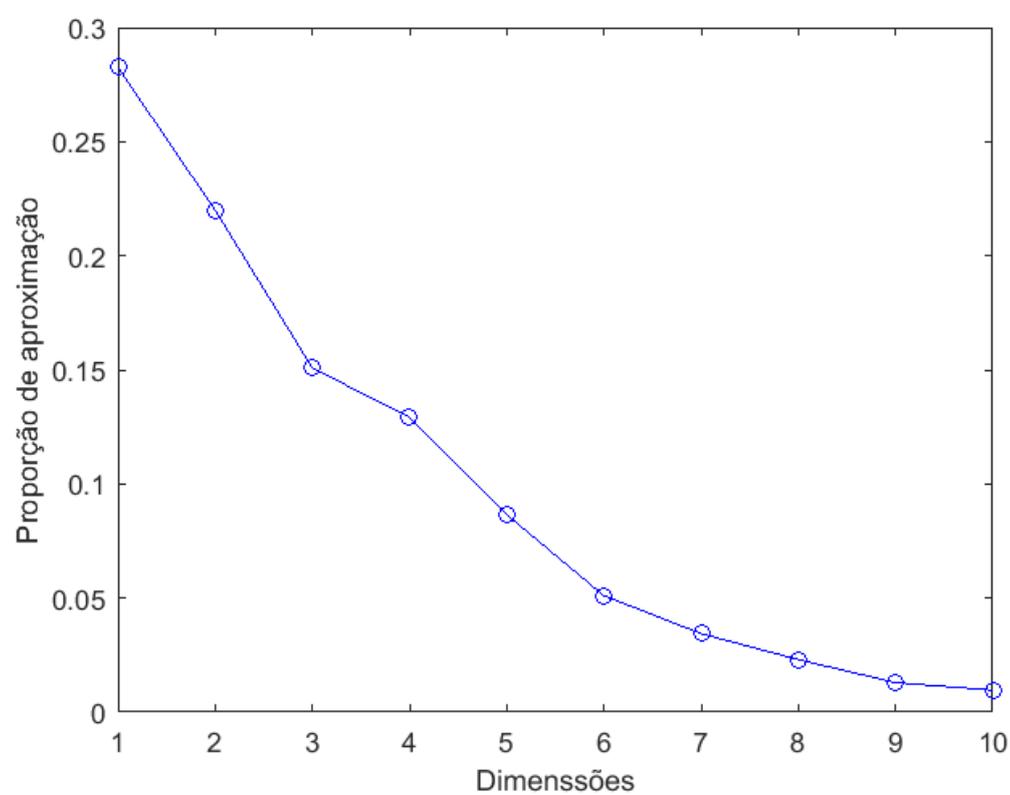


Figura 6.10: Projeto 3 - Proporção de aproximação das p reduções de dimensionalidade através do SVD

Tabela 6.21: Projeto 3 - Sub-problemas decompostos pelo método proposto com SVD(3) e Louvain

Sub-problema	Elementos do sub-problema										
1	' i'	' t'	' s'	' T'	' S'	' E'	' h3'	' h4'	' g1'	' g4'	' h2'
2	' Z'	' z1'	' z2'	' z3'	' w1'	' w2'	' w3'	' obj'			
3	' f'	' p'	' F'	' P'	' h5'	' g2'	' g3'	' h1'			

Tabela 6.22: Projeto 3 - Sub-problemas decompostos pelo método proposto sem RD e com Louvain

Sub-problema	Elementos do sub-problema									
1	' i'	' f'	' Z'	' z1'	' w1'	' w2'	' w3'	' F'	' obj'	' h3'
	' g2'	' h1'								
2	' p'	' z3'	' P'	' h5'	' g3'					
3	' s'	' z2'	' S'	' h4'	' g4'					
4	' t'	' T'	' E'	' g1'	' h2'					

$$A_{SP}(\text{SVD}_3, \text{3-means}) = \begin{bmatrix} 8 & 2 & 1 \\ 2 & 12 & 2 \\ 1 & 2 & 7 \end{bmatrix} \quad (6.12)$$

6.3.2.2 Método Proposto com SVD e Louvain

As três comunidades, sub-problemas, encontradas pelo método proposto para o Projeto 3 são apresentados na Tab. 6.21, e matriz de adjacência entre os elas pela Eq. 6.13.

$$A_{SP}(\text{SVD}_3, \text{Louvain}) = \begin{bmatrix} 12 & 2 & 2 \\ 2 & 7 & 1 \\ 2 & 1 & 8 \end{bmatrix} \quad (6.13)$$

6.3.2.3 Método Proposto com Louvain e sem RD

As quatro comunidades, sub-problemas, encontradas pelo método proposto para o Projeto 3 são apresentados na Tab. 6.22, e matriz de adjacência entre os elas pela Eq. 6.14.

Tabela 6.23: Projeto 3 - Quantidade de sub-problemas obtidos

Método	Sub-problemas
SVD(3) + 2-means	2
SVD(3) + 3-means	3
SVD(3) + 4-means	4
SVD(4) + 2-means	2
SVD(4) + 3-means	3
SVD(4) + 4-means	4
SVD(3) + Louvain	3
SVD(4) + Louvain	4
Sem RD + Louvain	4

Tabela 6.24: Projeto 3 - Medidas de qualidade das comunidades/agrupamentos (a)

Método	Modularidade	<i>Cut Size</i>	BIC	AIC
SVD(3) + 2-means	0.3940	6	183.2751	113.2999
SVD(3) + 3-means	0.4937	10	271.9182	166.9554
SVD(3) + 4-means	0.5503	12	360.6482	220.6978
SVD(4) + 2-means	0.3940	6	183.2751	113.2999
SVD(4) + 3-means	0.4458	8	271.9562	166.9934
SVD(4) + 4-means	0.5503	12	360.6482	220.6978
SVD(3) + Louvain	0.4937	10	271.9182	166.9554
SVD(4) + Louvain	0.5503	12	360.6482	220.6978
Sem RD + Louvain	0.4917	14	360.9666	221.0162

$$A_{SP}(\text{Sem RD, Louvain}) = \begin{bmatrix} 11 & 2 & 1 & 2 \\ 2 & 5 & 0 & 1 \\ 1 & 0 & 5 & 1 \\ 2 & 1 & 1 & 4 \end{bmatrix} \quad (6.14)$$

6.3.2.4 Medidas de Qualidade

As medidas são apresentadas nas Tabelas: 6.23, 6.24 e 6.25 tanto para os sub-problemas obtidos para o método de referência (com o melhor k identificado) e para o método proposto 1, ambos combinados com a redução via SVD para um espaço tridimensional, quanto para os sub-problemas gerados pelo método proposto 2.

Tabela 6.25: Projeto 3 - Medidas de qualidade das comunidades/agrupamentos (b)

Método	Silhueta Empírica	Silhueta Iguatária
SVD(3) + 2-means	0.1613	0.1715
SVD(3) + 3-means	0.2223	0.2398
SVD(3) + 4-means	0.2360	0.2280
SVD(4) + 2-means	0.1613	0.1715
SVD(4) + 3-means	0.2272	0.2767
SVD(4) + 4-means	0.2360	0.2280
SVD(3) + Louvain	0.2223	0.2398
SVD(4) + Louvain	0.2360	0.2280
Sem RD + Louvain	0.0966	0.1066

6.4 Projeto 4 - Mola sob Tração e Compressão

O objetivo do projeto da mola sob tração e compressão proposto por [87] *apud* [10], e representado na Fig. 6.11, é minimizar o volume V . As variáveis de projeto são o número de espirais ativas da mola ($N = x_1 \in [2, 15]$), o diâmetro de cada volta ($D = x_2 \in [0.25, 1.3]$) e o diâmetro do arame ($d = x_3 \in [0.05, 2]$). A Eq. 6.15 apresenta a formulação matemática deste ODP:

$$\begin{aligned}
 \min \quad & V = (x_1 + 2)x_2x_3^2 \\
 \text{s.a.} \quad & g_1(x) : 1 - \frac{x_3^2x_1}{71785x_3^4} \leq 0 \\
 & g_2(x) : \frac{4x_2^2 - x_3x_2}{12566(x_2x_3^3 - x_3^4)} + \frac{1}{5108x_3^2} - 1 \leq 0 \\
 & g_3(x) : 1 - \frac{140.45x_3}{x_2^2x_1} \leq 0 \\
 & g_4(x) : \frac{x_2 + x_3}{1.5} \leq 0
 \end{aligned} \tag{6.15}$$

6.4.1 Escolha das Dimensões do Espaço Reduzido

Avaliando o *scree plot* (Fig.6.12) obtido para o SVD aplicação a matriz de ocorrência derivada do modelo do Projeto 4, foi decidido trabalhar com $p = 2$, pois apesar de apresentar uma aproximação acumulada de 87,39%, segundo a heurística proposta por Sarkar *et al.* [37] deve-se iniciar o incremento das dimensões em $p = 2$. Mas, pensar do auto valor de aproximação, o espaço reduzido bidimensional continua apresentando uma das característica desejáveis, segundo a aplicação da heurística diretamente sobre o *scree plot* proposta neste trabalho, que é estar localizado no "cotovelo" da curva.

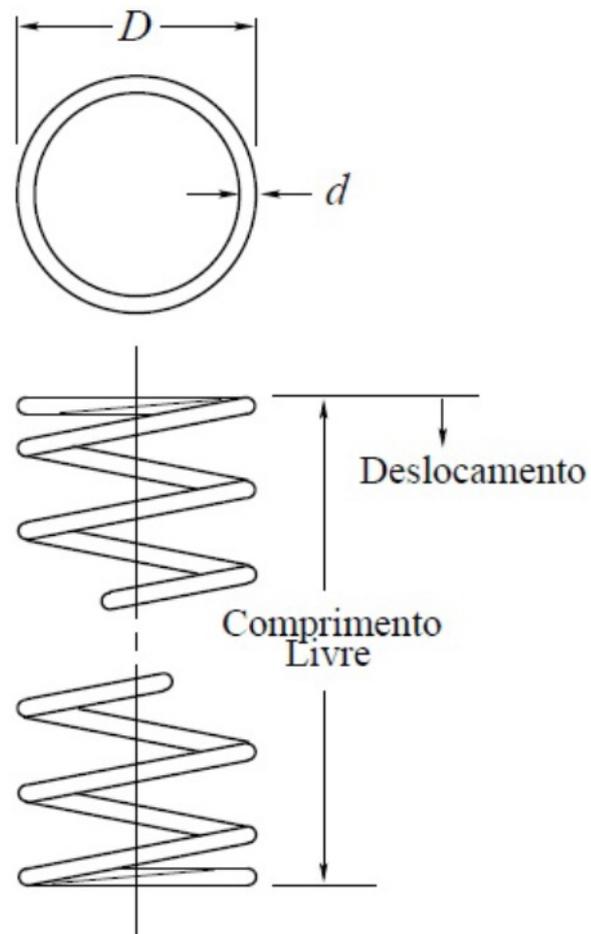


Figura 6.11: Mola sob tração e compressão (Retirado de [10])

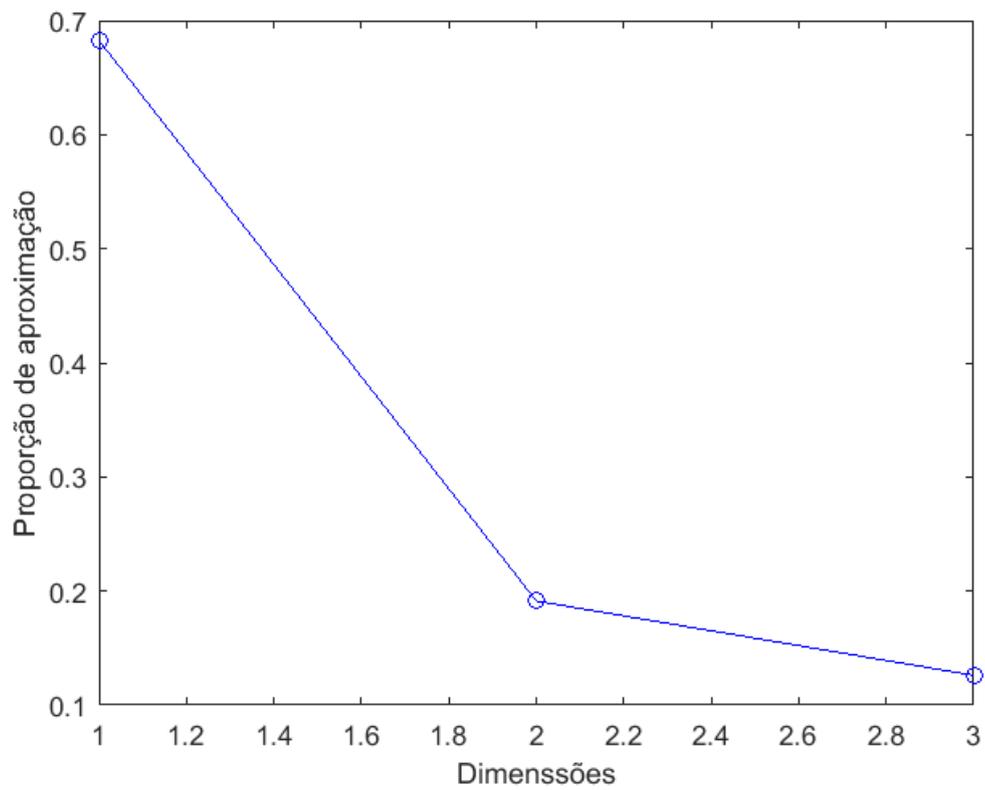


Figura 6.12: Projeto 4 - Proporção de aproximação das p reduções de dimensionalidade através do SVD

Tabela 6.26: Projeto 4 - Sub-problemas decompostos pelo método de referência com SVD(2) e K-means(2)

Sub-problema	Elementos do sub-problema									
1	' x2'	' x3'	' V'	' g2'	' g3'	' g4'	' g7'	' g8'	' g9'	' g10'
2	' x1'	' g1'	' g5'	' g6'						

Tabela 6.27: Projeto 4 - Sub-problemas decompostos pelo método proposto com SVD(2) e Louvain

Sub-problema	Elementos do sub-problema								
1	' x2'	' x3'	' g2'	' g4'	' g7'	' g8'	' g9'	' g10'	
2	' x1'	' V'	' g1'	' g3'	' g5'	' g6'			

6.4.2 Resultados

6.4.2.1 Método de Referência

O método de referência para a quantidade de dimensões escolhida ($p = 2$) e com o agrupamento através do algoritmo 2 – *means* obteve os sub-problemas definidos na Tab. 6.26, que interagem entre si conforme a matriz na Eq. 6.16.

$$A_{SP}(\text{SVD}_2, 2\text{-means}) = \begin{bmatrix} 12 & 3 \\ 3 & 3 \end{bmatrix} \quad (6.16)$$

6.4.2.2 Método Proposto com SVD e Louvain

O método proposto 1 para a quantidade de dimensões escolhida ($p = 2$) e com a detecção de comunidades realizada através do método de Louvain encontrou os dois sub-problemas descritos na Tab. 6.27, sendo as interações entre as comunidades, sub-problemas, apresentadas pela matriz de adjacência da Eq. 6.17.

$$A_{SP}(\text{SVD}_2, \text{Louvain}) = \begin{bmatrix} 8 & 5 \\ 5 & 5 \end{bmatrix} \quad (6.17)$$

6.4.2.3 Método Proposto com Louvain e sem RD

O método proposto 2, sem redução de dimensionalidade e com a detecção de comunidades realizada através do método de Louvain encontrou os três sub-problemas descritos na Tab. 6.28, sendo as interações entre as comunidades, sub-problemas,

Tabela 6.28: Projeto 4 - Sub-problemas decompostos pelo método proposto sem RD e com Louvain

Sub-problema	Elementos do sub-problema				
1	' x1'	' V'	' g3'	' g5'	' g6'
2	' x2'	' g2'	' g4'	' g7'	' g8'
3	' x3'	' g1'	' g9'	' g10'	

Tabela 6.29: Projeto 4 - Quantidade de sub-problemas obtidos

Método	Sub-problemas
SVD + 2-means	2
SVD + Louvain	2
Sem RD + Louvain	3

apresentadas pela matriz de adjacência da Eq. 6.18.

$$A_{SP}(\text{Sem RD, Louvain}) = \begin{bmatrix} 4 & 2 & 3 \\ 2 & 4 & 2 \\ 3 & 2 & 3 \end{bmatrix} \quad (6.18)$$

6.4.2.4 Medidas de Qualidade

As medidas de qualidade para as decomposições obtidas pelos diferentes métodos aplicados ao Projeto 4 são apresentadas comparativamente nas Tabelas: 6.29, 6.30 e 6.31.

6.5 Projeto 5 - Redutor de Velocidade

O problema de projetar um redutor de velocidade como o da Fig. 6.13 definido em [87] *apud* [10], tem por objetivo minimizar o peso W do redutor de velocidade. As variáveis de projeto são a largura da face ($b = x_1 \in [2.6, 3.6]$), o módulo dos dentes ($m = x_2 \in [0.7, 0.8]$), o número de dentes do pinhão ($n = x_3 \in [17, 28]$), o comprimento

Tabela 6.30: Projeto 4 - Medidas de qualidade das comunidades/agrupamentos (a)

Método	Modularidade	Cut Size	BIC	AIC
SVD + 2-means	0.2083	6	76.9573	59.0637
SVD + Louvain	0.2083	10	76.9573	59.0637
Sem RD + Louvain	0.2762	14	113.5560	86.7155

Tabela 6.31: Projeto 4 - Medidas de qualidade das comunidades/agrupamentos (b)

Método	Silhueta Empírica	Silhueta Iguatária
SVD + 2-means	0.2581	0.2595
SVD + Louvain	0.2129	0.2254
Sem RD + Louvain	0.1699	0.1654

do eixo 1 entre os rolamentos ($l_1 = x_4 \in [7.3, 8.3]$), o comprimento do eixo entre os dois rolamentos ($l_2 = x_5 \in [7.8, 8.3]$), o diâmetro do eixo 1 ($d_1 = x_6 \in [2.9, 3.9]$), e, finalmente, o diâmetro do eixo 2 ($d_2 = x_7$). A variável x_3 é inteira e todas as outras são contínuas.

O peso e as restrições mecânicas são dadas por:

$$\begin{aligned}
\min \quad & W = 0.7854x_1x_2^2(3.3333x_3^2 + 14.9334x_3 - 43.0934) \\
\text{s.a.} \quad & -1.508x_1(x_6^2 + x_7^2) + 7.4777(x_6^3 + x_7^3) \\
& + 0.7854(x_4x_6^2 + x_5x_7^2) \\
& g_1(x) = 27x_1^{-1}x_2^{-2}x_3^{-1} \leq 1 \\
& g_2(x) = 397.5x_1^{-1}x_2^{-2}x_3^{-2} \leq 1 \\
& g_3(x) = 1.93x_2^{-1}x_3^{-1}x_4^3x_6^{-4} \leq 1 \\
& g_4(x) = 1.93x_2^{-1}x_3^{-1}x_5^3x_7^{-4} \leq 1 \\
& g_5(x) = \frac{1}{0.1x_6^3} \left[\left(\frac{745x_4}{x_2x_3} \right)^2 + \{16.9\}10^6 \right]^{0.5} \leq 1100 \\
& g_6(x) = \frac{1}{0.1x_7^3} \left[\left(\frac{745x_5}{x_2x_3} \right)^2 + (157.5)10^6 \right]^{0.5} \leq 850 \\
& g_7(x) = x_2x_3 \leq 40 \quad g_8(x) = x_1/x_2 \geq 5 \\
& g_9(x) = x_1/x_2 \leq 12 \quad g_{10}(x) = (1.5x_6 + 1.9)x_4^{-1} \leq 1 \\
& g_{11}(x) = (1.1x_7 + 1.9)x_5^{-1} \leq 1
\end{aligned}$$

6.5.1 Escolha das Dimensões do Espaço Reduzido

Estudando o *scree plot* (Fig.6.14) gerado a partir das proporções de aproximações de cada valor p para o SVD aplicado a matriz de ocorrência derivada do modelo do Projeto 5, foi decidido trabalhar com $p = 2$, o qual apresenta proporção acumulada de aproximação igual a 65,58% e está localizado no "cotovelo" da curva, conforme a técnica

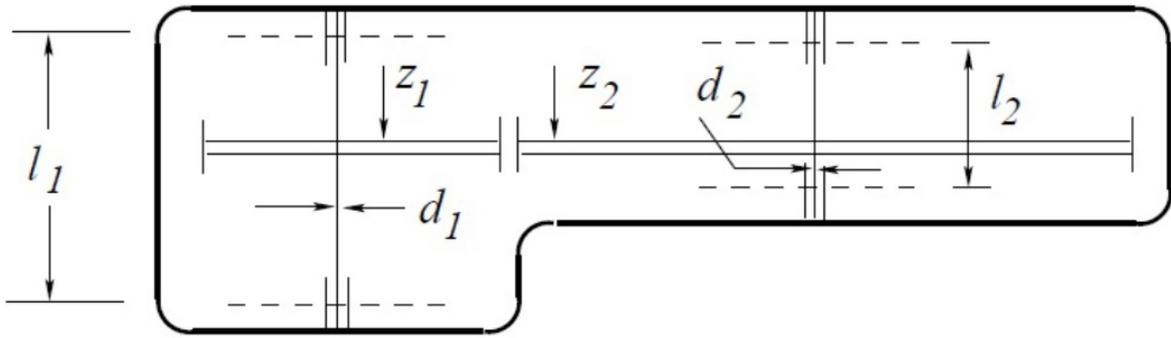


Figura 6.13: Redutor de velocidade (Retirado de [10])

Tabela 6.32: Projeto 5 - Sub-problemas decompostos pelo método de referência com SVD(2) e K-means(2)

Sub-problema	Elementos do sub-problema									
1	' x3'	' x4'	' x5'	' x6'	' x7'	' g3'	' g4'	' g5'	' g6'	' g10'
	' g11'	' g16'	' g17'	' g18'	' g19'	' g20'	' g21'	' g22'		
2	' x1'	' x2'	' f'	' g1'	' g2'	' g7'	' g8'	' g9'	' g12'	' g13'
	' g14'	' g15'								

também utilizada para os demais projetos.

6.5.2 Resultados

Pelo estudo das proporções de aproximação e do *scree plot*, percebe-se claramente que a quantidade mais adequada de dimensões são duas ($p = 2$).

6.5.2.1 Método de Referência

O método de referência encontrou os sub-problemas descritos na Tab. 6.32, sendo as interações entre os sub-problemas apresentadas pela matriz de adjacência da Eq. 6.19.

$$A_{SP}(\text{SVD}_2, 2\text{-means}) = \begin{bmatrix} 24 & 10 \\ 10 & 15 \end{bmatrix} \quad (6.19)$$

6.5.2.2 Método Proposto com SVD e Louvain

O método proposto 1 encontrou os sub-problemas descritos na Tab. 6.33, sendo as interações entre os sub-problemas apresentadas pela matriz de adjacência da Eq. 6.20.

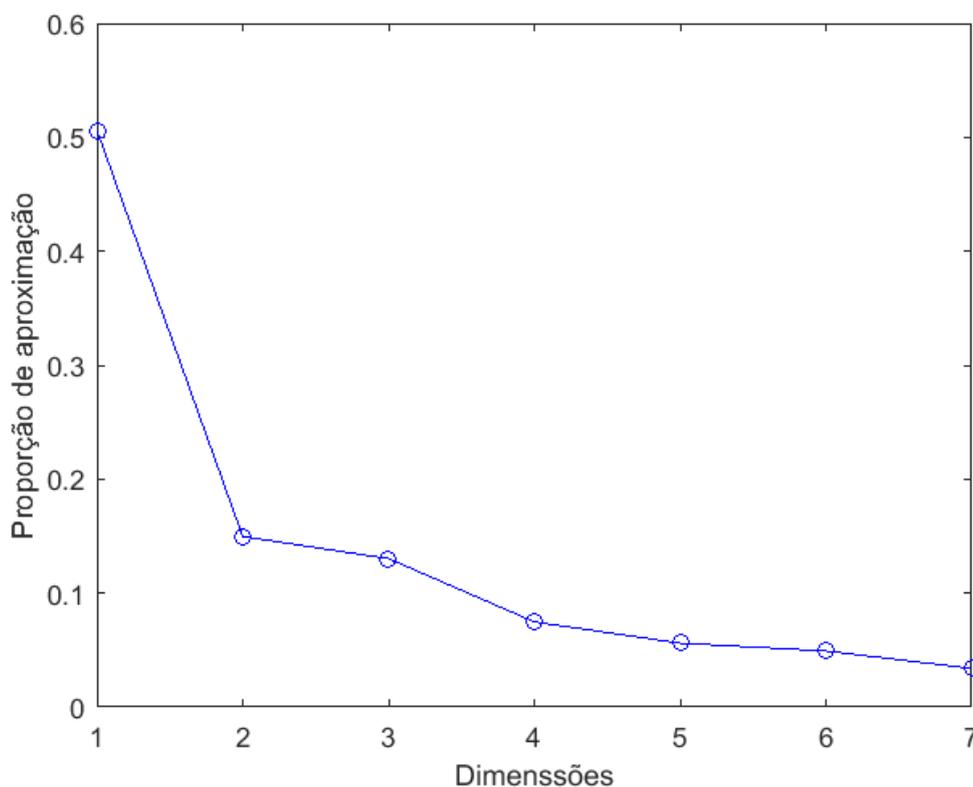


Figura 6.14: Projeto 5 - Proporção de aproximação das p reduções de dimensionalidade através do SVD

Tabela 6.33: Projeto 5 - Sub-problemas decompostos pelo método proposto com SVD(2) e Louvain

Sub-problema	Elementos do sub-problema									
1	' x3'	' x4'	' x5'	' x6'	' x7'	' g4'	' g5'	' g6'	' g10'	' g11'
	' g16'	' g17'	' g18'	' g19'	' g20'	' g21'	' g22'	' g23'		
2	' x1'	' x2'	' W'	' g1'	' g2'	' g3'	' g7'	' g8'	' g9'	' g12'
	' g13'	' g14'	' g15'							

Tabela 6.34: Projeto 5 - Sub-problemas decompostos pelo método proposto sem RD e com Louvain

Sub-problema	Elementos do sub-problema									
1	' x3'	' x5'	' x7'	' W'	' g4'	' g6'	' g7'	' g11'	' g16'	' g17'
	' g20'	' g21'								
2	' x1'	' x2'	' g1'	' g2'	' g8'	' g9'	' g12'	' g13'	' g14'	' g15'
3	' x4'	' x6'	' g3'	' g5'	' g10'	' g18'	' g19'	' g22'	' g23'	

Tabela 6.35: Projeto 5 - Quantidade de sub-problemas obtidos

Método	Sub-problemas
SVD + 2-means	2
SVD + Louvain	2
Sem RD + Louvain	3

$$A_{SP}(\text{SVD}_2, \text{Louvain}) = \begin{bmatrix} 21 & 12 \\ 12 & 16 \end{bmatrix} \quad (6.20)$$

6.5.2.3 Método Proposto com Louvain e sem RD

O método proposto 2 encontrou os sub-problemas descritos na Tab. 6.34, sendo as interações entre os sub-problemas apresentadas pela matriz de adjacência da Eq. 6.21.

$$A_{SP}(\text{Sem RD}, \text{Louvain}) = \begin{bmatrix} 15 & 7 & 4 \\ 7 & 12 & 2 \\ 4 & 2 & 9 \end{bmatrix} \quad (6.21)$$

6.5.2.4 Medidas de Qualidade

As medidas de qualidade para as decomposições obtidas pelos diferentes métodos aplicados ao Projeto 5 são apresentadas comparativamente nas Tabelas: 6.35, 6.36 e 6.37.

Tabela 6.36: Projeto 5 - Medidas de qualidade das comunidades/agrupamentos (a)

Método	Modularidade	<i>Cut Size</i>	BIC	AIC
SVD + 2-means	0.2791	20	218.2743	129.3671
SVD + Louvain	0.2499	24	218.2994	129.3922
Sem RD + Louvain	0.3863	26	324.4280	191.0672

Tabela 6.37: Projeto 5 - Medidas de qualidade das comunidades/agrupamentos (b)

Método	Silhueta Empírica	Silhueta Iguatária
SVD + 2-means	0.1608	0.1940
SVD + Louvain	0.1519	0.1759
Sem RD + Louvain	0.1881	0.1937

6.6 Projeto 6 - Viga Soldada

No problema de projetar a viga soldada (*welded beam*) da Fig. 6.15 descrito por [88] *apud* [10], o objetivo é minimizar o custo $C(h, l, t, b)$ da viga sujeita a um carregamento F onde $h \in [0.125, 10]$, e $0.1 \leq l, t, b \leq 10$. O objetivo e restrições são:

$$\begin{aligned}
 \min \quad & C(h, l, t, b) = 1.10471h^2l + 0.04811tb(14.0 + l) \\
 \text{s.a.} \quad & g_1(\tau) = 13,600 - \tau \geq 0 \\
 & g_2(\sigma) = 30,000 - \sigma \geq 0 \\
 & g_3(b, h) = b - h \geq 0 \\
 & g_4(P_c) = P_c - 6,000 \geq 0 \\
 & g_5(\delta) = 0.25 - \delta \geq 0
 \end{aligned}$$

Onde as expressões para τ , σ , P_c , e δ são dadas por:

$$\begin{aligned}
 \tau &= \sqrt{(\tau')^2 + (\tau'')^2 + l\tau'\tau''/\alpha} \\
 \tau' &= \frac{6000}{\sqrt{2hl}} \\
 \alpha &= \sqrt{0.25(l^2 + (h + t)^2)} \\
 \sigma &= \frac{504000}{t^2b} \\
 P_c &= 64746.022(1 - 0.0282346t)tb^3 \\
 \delta &= \frac{2.1952}{t^3b} \\
 \tau'' &= \frac{6000(14 + 0.5l)\alpha}{2(0.707hl(l^2/12 + 0.25(h + t)^2))}
 \end{aligned}$$

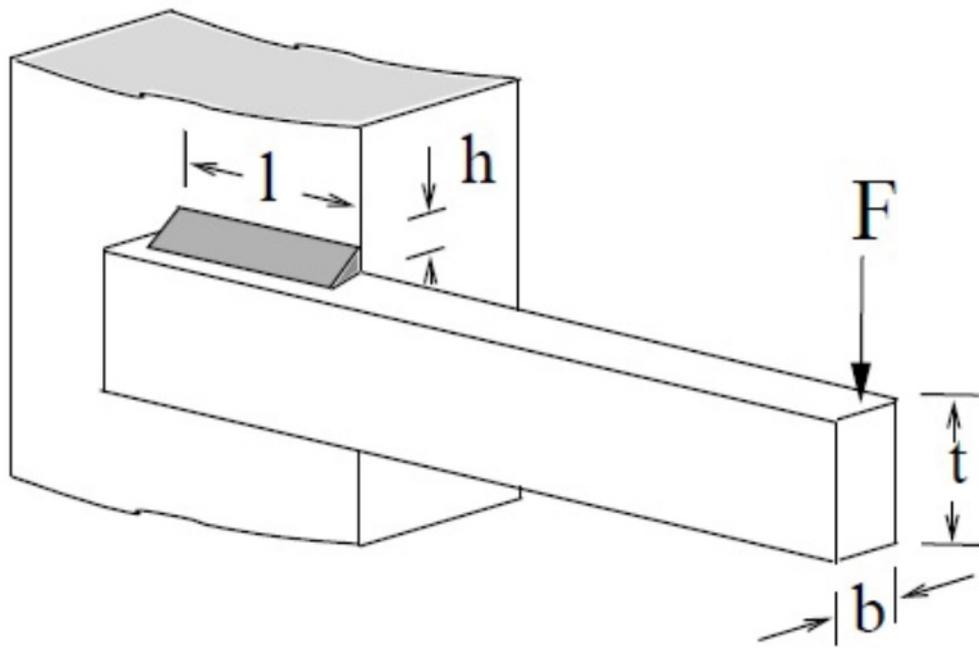


Figura 6.15: Viga soldada (Retirado de [10])

Tabela 6.38: Projeto 6 - Sub-problemas decompostos pelo método de referência com SVD(3) e K-means(3)

Sub-problema	Elementos do sub-problema																	
1	' b'	' σ '	' P_c '	' δ '	' g2'	' g4'	' g5'	' h4'	' h5'	' h6'	' g10'	' g11'	' g12'	' g13'				
2	' l'	' τ '	' τ'' '	' α '	' τ''' '	' g1'	' h1'	' g8'	' g9'	' h'	' t'	' C'	' g3'	' h2'	' g6'	' g7'	' h3'	' h7'

6.6.1 Escolha das Dimensões do Espaço Reduzido

Através do *scree plot* (Fig.6.16) referente as as proporções de aproximações das dimensões p para o SVD sobre a matriz de ocorrência do modelo do Projeto 6, foi decidido trabalhar com $p = 3$, com proporção acumulada de aproximação igual a 69,61% e está localizado no "cotovelo" da curva do declive, de acordo com a heurística utilizada.

6.6.2 Resultados

6.6.2.1 Método de Referência

Executando o método de referência através do SVD com $p = 3$ e do K-means com $k = 3$ foram encontrados os sub-problemas da Tab. 6.38, os quais podem ter sua interação expressa conforme a matriz da Eq. 6.22.

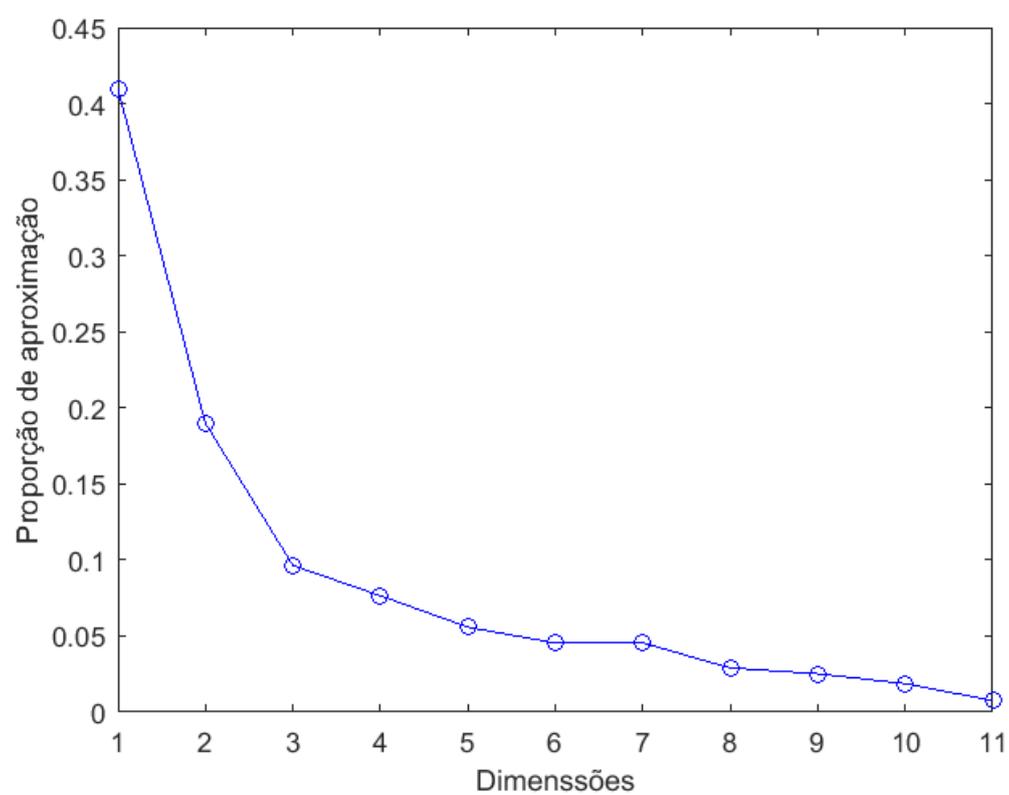


Figura 6.16: Projeto 6 - Proporção de aproximação das p reduções de dimensionalidade através do SVD

Tabela 6.39: Projeto 6 - Sub-problemas decompostos pelo método proposto com SVD(3) e Louvain

Sub-problema	Elementos do sub-problema								
1	' b'	' σ '	' P_c '	' δ '	' g2'	' g4'	' g5'	' h4'	' h5'
	' h6'	' g10'	' g11'	' g12'	' g13'				
2	' l'	' τ '	' τ' '	' α '	' τ'' '	' g1'	' h1'	' h3'	' h7'
	' g8'	' g9'							
3	' h'	' t'	' C'	' g3'	' h2'	' g6'	' g7'		

$$A_{SP}(\text{SVD}_3, \text{3-means}) = \begin{bmatrix} 11 & 0 & 7 \\ 0 & 8 & 7 \\ 7 & 7 & 11 \end{bmatrix} \quad (6.22)$$

6.6.2.2 Método Proposto com SVD e Louvain

Executando o método proposto 1 através do SVD com $p = 3$ e do método de Louvain foram encontrados os três sub-problemas da Tab. 6.39, os quais podem ter sua interação expressa conforme a matriz de adjacência entre comunidades conforme a Eq. 6.23.

$$A_{SP}(\text{SVD}_3, \text{Louvain}) = \begin{bmatrix} 11 & 0 & 7 \\ 0 & 13 & 6 \\ 7 & 6 & 7 \end{bmatrix} \quad (6.23)$$

6.6.2.3 Método Proposto com Louvain e sem RD

Executando o método proposto 2 através do método de Louvain sem a etapa de redução de dimensionalidade, foram encontrados os onze sub-problemas da Tab. 6.39, os quais podem ter sua interação expressa conforme a matriz de adjacência entre comunidades conforme a Eq. 6.23.

Tabela 6.40: Projeto 6 - Sub-problemas decompostos pelo método proposto sem RD e com Louvain

Sub-problema	Elementos do sub-problema			
1	' h '	' g3 '	' g6 '	' g7 '
2	' l '	' C '	' g8 '	' g9 '
3	' σ '	' g2 '	' h4 '	
4	' P_c '	' g4 '	' h5 '	
5	' δ '	' g5 '	' h6 '	
6	' α '	' h3 '	' h7 '	
7	' t '	' g10 '	' g11 '	
8	' b '	' g12 '	' g13 '	
9	' τ '	' g1 '		
10	' τ'' '	' h1 '		
11	' τ' '	' h2 '		

Tabela 6.41: Projeto 6 - Quantidade de sub-problemas obtidos

Método	Sub-problemas
SVD + 3-means	3
SVD + Louvain	3
Sem RD + Louvain	11

$$A_{SP}(\text{Sem RD, Louvain}) = \begin{bmatrix} 3 & 1 & 0 & 0 & 0 & 2 & 0 & 1 & 0 & 0 & 1 \\ 1 & 3 & 0 & 0 & 0 & 2 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 1 & 1 & 0 & 0 & 0 \\ 2 & 2 & 0 & 0 & 0 & 2 & 2 & 0 & 0 & 2 & 0 \\ 0 & 1 & 1 & 1 & 1 & 2 & 2 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 2 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{bmatrix} \quad (6.24)$$

6.6.2.4 Medidas de Qualidade

As medidas de qualidade para as decomposições obtidas para os métodos com redução de dimensionalidade aplicados ao Projeto 6 são apresentadas comparativamente nas Tabelas: 6.41, 6.42 e 6.43.

Tabela 6.42: Projeto 6 - Medidas de qualidade das comunidades/agrupamentos (a)

Método	Modularidade	<i>Cut Size</i>	BIC	AIC
SVD + 3-means	0.3376	28	338.1492	197.4385
SVD + Louvain	0.3696	26	338.2138	197.5031
Sem RD + Louvain	0.3737	46	1.2246e+03	708.6330

Tabela 6.43: Projeto 6 - Medidas de qualidade das comunidades/agrupamentos (b)

Método	Silhueta Empírica	Silhueta Iguatária
SVD + 3-means	0.1246	0.1388
SVD + Louvain	0.0971	0.1139
Sem RD + Louvain	-0.0083	-0.0390

6.7 Discussão

Neste capítulo, os três métodos foram aplicados na decomposição de seis problemas de otimização de projeto em engenharia, provindos tanto da literatura de decomposição de sistemas, quanto de trabalhos gerais de otimização de projeto.

O objetivo destes experimentos é a compreensão do comportamento dos métodos propostos perante as aplicações comparativamente ao método de referência [37], e a validação da possibilidade de utilização destas técnicas oriundas de outras áreas do conhecimento como forma de resolução adequada e eficaz no contexto de decomposição.

A fim de permitir o estudo e avaliação dos métodos propostos perante os resultados, os problemas de otimização de projeto escolhidos para aplicação são sistemas de tamanho, complexidade, e multidisciplinaridades menores que sistemas complexos utilizados na prática. Entretanto, assume-se que os comportamentos observados nos experimentos anteriormente apresentados serão refletidos para qualquer cenário, visto que as técnicas fundamentais são reconhecidamente escaláveis em questão de custo e qualidade quanto sua utilização nas áreas de origem.

No Projeto 1, os processos e resultados foram descritos com maior riqueza de detalhes, basicamente com a intenção de esclarecer o funcionamento dos fluxos de resolução e da interpretação. Utilizando, assim, este projeto como exemplo, e gradativamente nos demais projetos, reduziu-se o detalhamento, conforme diferentes cenários necessários de ilustração eram apresentados. Contudo, se manteve constante o relato das medidas avaliativas. As quais, são listadas a seguir em conjunto com o respectivo objetivo desejado:

- Maximizar a quantidade de sub-problemas;
- Maximizar a modularidade;
- Minimizar a quantidade de arestas cortadas pelas fronteiras dos sub-problemas (*cut size*);
- Minimizar o critério de informação Bayesiano (*Bayesian information criterion - BIC*);
- Minimizar o critério de informação de Akaike (*Akaike information criterion - AIC*);
- Maximizar a silhueta dos sub-problemas, nas duas formulações:
 - empírica: cada agrupamento contribui proporcionalmente ao seu tamanho;
 - igualitária: cada agrupamento contribui igualmente.

A primeira é diretamente obtida para o problema de decomposição no caso dos métodos propostos e definida com parâmetro para o K-means; a modularidade e o *cut size* são medidas oriundas das áreas de rede complexas [81][8][75] e particionamento de grafos [11][75]; por sua vez, os critérios de informação Bayesiano e de Akaike [65][77][89], assim como as duas formulações da silhueta (*clusters silhouette*) [90][91][59] são amplamente utilizadas na definição do valor k dentro da análise de agrupamento.

Em todos os três métodos - K-means com SVD (método de referência [37], "SVD+K-means"), Louvain com SVD (método proposto 1, principal, "SVD+Louvain"), e Louvain sem redução de dimensionalidade (método proposto 2, estudo de comportamento, "SemRD+Louvain") -, a primeira tarefa realizada nos experimentos foi a interpretação dos modelos como um conjunto de símbolos e a obtenção da matriz de ocorrências.

Em seguida, para os métodos de referência e o proposto 1, é realizada a redução de dimensionalidade através da decomposição em valor singular, havendo a necessidade de interação do projetista, ou engenheiro responsável pela decomposição, na definição do valor p de dimensões a serem retidas.

Sarkar *et al* [9] propõem uma heurística iterativa para a realização desta escolha. Já neste trabalho, definiu-se diretamente pela análise da proporção de aproximação, da proporção acumulada de aproximação e seus respectivos *scree plots*, guiando a decisão sempre pelos princípios assumidos na heurística referida. Observando que no caso dos

projetos 1, 2 e 3, a escolha realizada para p foi a mesma da definida na decomposição realizada para o método de referência em [37]. Portanto, a mesma estratégia, também foi utilizada para os problemas 4, 5 e 6 (retirados da literatura de otimização de projeto, sem decomposição).

Tendo sido definido o espaço dimensional reduzido e os elementos do modelo representados neste novo espaço, a matriz de dados considerando estes elementos como vetores é enviada como entrada para diversos K-means, variando o parâmetro k e sempre utilizando a dissimilaridade de cosseno. Por sua vez, para o método SVD+Louvain, calcula-se a similaridade de cosseno entre os elementos do modelo no espaço reduzido e transforma a matriz de distância na matriz de adjacência do grafo bipartido representante, o qual é enviado como entrada para o método de Louvain detectar as comunidades sem a necessidade de qualquer parâmetro. Comparativamente, nota-se duas vantagens do método proposto: o cálculo das dissimilaridades/similaridades é realizado uma única vez e não a cada iteração (que é o caso do K-means tradicional); e a ausência de definição de um parâmetro.

Por fim, os algoritmos informam os agrupamentos ou comunidades de cada elemento, que são interpretados como os respectivos sub-problemas. Então as medidas avaliativas são obtidas sobre a matriz de ocorrência antes do SVD, mas agora categorizada com os sub-projetos de cada elemento. Para a identificação das variáveis de ligação pode-se utilizar tanto a estratégia de [9] através da matriz de ocorrência reorganizada, onde as variáveis que ocorrerem em funções externas ao seu grupo serão as variáveis de ligação¹. Outra maneira, mais natural para o método proposto, é identificar no grafo representativo da matriz de ocorrência original quais são os nós de variáveis que possuem arestas para funções² externas ao seu sub-problema.

Ao analisar os resultados para os dois métodos com redução de dimensionalidade o desempenho do método proposto utilizando o Louvain é comparativamente muito bom em relação ao método com o K-means. Observa-se que para os projetos 1, 2, 3 e 6 em todas as medidas o método proposto obteve exatamente o mesmo resultado que o K-means com k igual a quantidade de comunidades detectadas pelo algoritmo de Louvain.

Desta forma, o método proposto se mostra como mais vantajoso, visto não exigir

¹Lembrando, que se as variáveis que apresentam este relacionamento com funções externas ao seu sub-problema forem variáveis de acoplamento elas não serão consideradas como variáveis de ligação devido a não adicionarem dificuldades ao problema de integração

²Por se tratar de um grafo bipartido, não há arestas entre as variáveis

a definição da quantidade de sub-problemas *a priori* e por mais que tanto a tarefa de agrupamento quanto a de detecção de comunidades sejam problemas NP-difíceis, a complexidade computacional do Louvain para o caso médio é melhor que o K-means, como era de se esperar, pois o Louvain foi criado exatamente para trabalhar em redes complexas em alta escala. Outra vantagem do método proposto é que o K-means perde a propriedade de garantir a convergência para um ótimo local ao ser utilizado em conjunto com a dissimilaridade de cosseno, enquanto para o algoritmo de Louvain esta propriedade se mantém imutável, a final, entende a similaridade de cosseno como apenas o peso das arestas, sendo indiferente a como este peso foi definido.

Seguindo a avaliação para os projetos 4 e 5, a detecção de comunidades obteve resultados ligeiramente piores que o agrupamento para as medidas de *cut size* e as silhuetas em ambos os projetos, e também para a modularidade, BIC e AIC no Projeto 5, mas, os valores obtidos foram muito próximos. Portanto, devido aos resultados nos projetos 1, 2 e 3 serem os mesmo do método de referência e bastante aproximados nos projetos 4, 5 e 6, considera-se que a hipótese de utilização do SVD em conjunto com o Louvain conforme o fluxo do método proposto 1 para a decomposição de sistemas para a resolução de ODPs através de seus sub-projetos é válida.

O outro foco de interesse nos experimentos era o comportamento do método proposto 2, com a detecção de comunidades sem a realização prévia da redução de dimensionalidade. Neste caso, observa-se que houve grande variação qualitativa do método, apresentando boas medidas avaliativas no Projeto 6, mas um resultado ruim, por exemplo no Projeto 1, onde a própria Fig. 6.7, que representa a decomposição em sub-problemas, define um sub-sistema com diversos nós que para irem de um a outro do mesmo sub-projeto tem que passar por nós externos. Conseqüentemente, não há como considerar o método proposto 2 como robusto, por mais que para algumas medidas obteve-se resultados satisfatórios, o que pode indicar o valor de investigações mais aprofundadas, principalmente com outras representações de entrada, que podem produzir um método adequado. Ainda, como o resultado do método sem a redução de dimensionalidade não pode ser considerado confiável, percebe-se a contribuição da etapa de redução de dimensionalidade na extração de padrões semânticos entre os elementos do modelo.

7 Considerações Finais

A modelagem de problemas em diversas áreas das ciências como engenharias (em suas mais variadas disciplinas), ciências econômicas, biológicas entre diversas outras tem gerado cada vez modelos maiores, com viés multidisciplinar e, principalmente, mais complexos. Estes sistemas complexos, apesar de representar com maior completude o fenômeno em estudo, necessitam de técnicas específicas para que se consiga obter soluções em tempo hábil ou, até mesmo, para que se consiga chegar a alguma solução.

Uma das vertentes que tem se apresentado como estratégia relevante para a manipulação de sistemas complexos, principalmente no que tange aos problemas de otimização de projeto são os métodos de decomposição e integração de sistemas. Usualmente tratados separadamente na literatura, seus avanços e suas combinações claramente carecem de desenvolvimentos para que se tornem ferramentas usuais e práticas, visando viabilizar a utilização de sistemas altamente complexos compostos por diversos níveis de componentes que apresentam vários níveis hierárquicos de interligação, conexão.

Neste contexto, o presente trabalho visa apresentar uma contribuição, principalmente, no que tange a questão da decomposição de um problema complexo em sub-problemas que apresentem conexão e aderência ao modelo original. Para isto, desenvolveu-se a construção de um procedimento de decomposição baseado em modelo usando técnicas pouco usuais na área mas que, pelas características que lhes são inerentes, podem indicar uma nova vertente para a geração de decomposições efetivas.

Nas diversas etapas que constituem um processo de decomposição baseada em modelo foram assumidas as seguintes diretrizes para o método:

- a possibilidade do uso de representação simbólica no modelo;
- o uso de redução de dimensionalidade visando ressaltar e desacoplar os padrões menos relevantes do modelo;
- a modelagem do problema como redes complexas.

Aliando, assim, a área de decomposição baseada em modelo para projetos de engenharia, com as de inteligência artificial, processamento de linguagem natural,

recuperação da informação, processamento de imagens digitais, visão computacional, aprendizado de máquina, descoberta de conhecimento em bases de dados, e com os estudos crescentes em ciência de redes, que por sua vez, são amplamente aplicados a sistemas complexos, mas oriundos de outras áreas (principalmente em ciências biológicas e sociais).

De maneira geral, estas combinações originaram um método de decomposição baseado em modelo que apresenta características únicas e interessantes que por si só merecem ser investigadas. Além de introduzir a integração de técnicas provindas de áreas tradicionalmente disjuntas para a resolução do problema de decomposição de projetos, abrindo, então, um novo ramo para estudos e avanços como: a obtenção de novos métodos e algoritmos, ou novas formas de utilização de técnicas oriundas desta áreas anteriormente não relacionadas.

Tal investigação se deu através de aplicações em problemas de otimização conhecidos e testados na literatura específica de decomposição de sistemas, bem como em alguns problemas tradicionais de otimização em engenharia. Medidas de avaliação qualitativas e quantitativas serviram de referência para a avaliação dos modelos.

Os resultados mostraram que o método proposto que combina a redução de dimensionalidade, utilizando o algoritmo SVD, e a detecção de comunidades, através do algoritmo de Louvain, consegue identificar os relacionamentos semânticos entre os elementos do modelo e produzir uma decomposição igual ou similar a obtida pelo método de referência que combina o SVD com o K-means. Sendo que, o método proposto, não necessita de definir previamente a quantidade de sub-problemas, e apresenta uma complexidade computacional melhor que o K-means e diferentemente deste, garante um ótimo local mesmo utilizando similaridade/dissimilaridade de cosseno.

Os experimentos sobre o segundo método proposto, que realiza a detecção de comunidades pelo Louvain sem a redução de dimensionalidade, revelaram que realmente é importante a extração de padrão desempenhada pelo SVD. Entretanto, o método obteve em alguns casos medidas boas, apesar de um desempenho ruim em outros, isto incentiva futuros estudos para compreensão dos cenários em que pode ser utilizado, ou a construção de outras representações para que consiga decomposições satisfatórias.

Vale ressaltar que o uso inicial em problemas "não tão complexos, grandes ou multidisciplinares", assim como de modelos originais com definição matemática formal,

é necessário para um melhor entendimento e validação das propriedades e características do método de decomposição apresentado, visto que estes problemas servem até mesmo de *benchmark* para comparação com outros métodos. Tem-se a expectativa de que as propriedades do modelo sejam conservadas, mesmo em se tratando de sistemas realmente complexos, grandes, multidisciplinares e com definições não matemáticas.

7.1 Trabalhos Futuros

Procedimentos de decomposição visando dividir um problema complexo no que tange a dimensão, multidisciplinaridade, interdependência das variáveis, entre outros aspectos, em sub-problemas relevantes e com alto nível de aderência ao problema original, continua a ser um desafio. O mesmo pode-se afirmar em relação ao processo subsequente de integração, o qual visa resolver eficientemente e com precisão o problema decomposto. Neste sentido, trabalhos futuros se alinham indicando direções de desenvolvimentos e avaliações adicionais.

De maneira geral, um maior aprofundamento em cada uma das etapas construtivas do método é de grande interesse, principalmente um melhor entendimento do processo de redução de dimensionalidade, como a utilização de outros algoritmos; e a determinação dos sub-problemas associados por meio de ferramentas oriundas de Redes Complexas, vão permitir novos caminhos, possibilidades e avanços na eficiência do processo de decomposição baseado em modelos. Entre as avaliações de maior interesse pode-se ressaltar:

- Aplicar o método a modelos de sistemas enunciados de formas não matemáticas, principalmente: DSM (*Design Structure Matrix*), DMM (*Domain Mapping Matrix*) e MDM (*Multiple-Domain Matrix*);
- Estudar a relevâncias de outras formas de interpretação simbólica do modelo, sendo uma sugestão a utilização da técnica Tf/Idf, comumente aplicada dentro de Recuperação de Informação em conjunto com as técnicas que inspiraram a utilização do SVD e da dissimilaridade de cosseno à decomposição de sistemas;
- Verificar a possibilidade e qualidade do método em conjunto com outras formas de representação do modelo, sendo algumas sugestões, o Laplaciano, Laplaciano

absoluto, ou a forma normalizada da matriz de adjacência, bem como, outros arranjos sobre as estruturas de grafos e hiper-grafos;

Uma tarefa importante a ser realizada são os testes da decomposição baseada em modelo em problemas com maior nível de complexidade. Assim, como, solucionar a tarefa de integração sobre problemas de projeto de sistemas decompostos pelo método proposto e aplicação em cenários reais em engenharia, incluindo a obtenção do projeto final.

REFERÊNCIAS

- [1] KHAN ACADEMY, “Algoritmos de divisão e conquista”, Disponível em: <https://pt.khanacademy.org/computing/computer-science/algorithms/merge-sort/a/divide-and-conquer-algorithms>, Acessado em: 30/Agosto/2016.
- [2] CARQUEIJO, L. F., “Sistemas de tração 4x4”, Disponível em: <http://www.autoentusiastas.com.br/2015/11/sistemas-tracao-4x4/>, Novembro 2015, Acessado em: 30/Agosto/2016.
- [3] WAGNER, T. C., PAPALAMBROS, P. Y., “Decomposition Analysis and Optimization of an Automotive Powertrain”, *Engineering Optimization*, v. 31, n. 3, pp. 273–299, 1999.
- [4] RESENDE, N., “BIM, o futuro dos projetos de engenharia”, Disponível em: <https://blogdopetcivil.com/2013/02/22/bim-o-futuro-dos-projetos-de-engenharia/>, Fevereiro 2013, Acessado em: 30/Agosto/2016.
- [5] MICHELENA, N. F., PAPALAMBROS, P. Y., “A Network Reliability Approach To Optimal Decomposition of Design Problems”. In: *Journal of Mechanical Design, ASME*, pp. 433–440, 1994.
- [6] MICHELENA, N. F., PAPALAMBROS, P. Y., “Optimal Model-Based Decomposition of Powertrain System Design”. In: *Proceedings of the ASME Design Engineering Technical Conferences*, pp. 165–172, Boston, MA, EUA, Setembro 1995.
- [7] “Inner-product-angle.svg”, 2016.
- [8] BARABÁSI, A.-L., “Network Science”, Disponível em: <https://pt.khanacademy.org/computing/computer-science/algorithms/merge-sort/a/divide-and-conquer-algorithms>, Acessado em: 30/Agosto/2016.
- [9] SARKAR, S., DONG, A., GERO, J. S., “Design Optimization Problem Reformulation Using Singular Value Decomposition”, *Journal of Mechanical Design*, v. 131, n. 8, 2009.

- [10] CARVALHO, É. C. R., LEMONGE, A. C. C., BERNARDINO, H. S., HALLAK, P. H., “Um Método De Penalização Adaptativa (APM) Para Problemas de Otimização Com Restrições Aplicado a Problemas de Engenharia”. In: *SIMMEC / EMMCOMP 2014*, pp. 28–30, 2014.
- [11] MICHELENA, N. F., PAPALAMBROS, P. Y., “A hypergraph framework for optimal model-based decomposition of design problems”, *Computational optimization and applications*, v. 8, n. 2, pp. 173–196, 1997.
- [12] PARKINSON, A., BALLING, R., HEDENGREN, J., *Optimization Methods for Engineering Design*. Brigham Young University, 2013.
- [13] ARORA, J. S., *Introduction to optimum design - 2nd ed.*. Elsevier Inc, 2010.
- [14] PAPALAMBROS, P. Y., “Design Science in Design Education”. In: *XI Mudd Design Workshop on Design Thinking in Design Education*, pp. 28–30, Clairmont, CA, EUA, 2015.
- [15] ARORA, J. S., *Introduction to optimum design - 4th ed.*. Academic Press, 2016.
- [16] RAO, S. S., *Engineering Optimization*. John Wiley & Sons, Inc.: Hoboken, NJ, EUA, Julho 2009.
- [17] PAPALAMBROS, P., MICHELENA, N., “Trends and Challenges in System Design Optimization”. In: *Proceedings of the International Workshop on Multidisciplinary Optimization*, pp. 1–15, Pretoria, Africa do Sul, Agosto 2000.
- [18] BALDWIN, C. Y., CLARK, K. B., “Modularity in the Design of Complex Engineering Systems”, *Complex Engineered Systems Understanding Complex Systems*, , n. January, pp. 175–205, 2004.
- [19] MARTINS, J. R. R. A., LAMBE, A. B., “Multidisciplinary Design Optimization: A Survey of Architectures”, *AIAA Journal*, v. 51, n. 9, pp. 2049–2075, 2013.
- [20] KARIMIAN, P., HERRMANN, J. W., “Separating Engineering Design Optimization Problems Application : Motor Design”, .
- [21] PIMMLER, T. U., EPPINGER, S. D., “Integration analysis of product decompositions”. pp. 343–351, 1994.

- [22] SARKAR, S., DONG, A., GERO, J. S., “Learning symbolic formulations in design: Syntax, semantics, and knowledge reification”, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, v. 24, n. 01, pp. 63, 2010.
- [23] BRADLEY, N., *THE RESPONSE SURFACE METHODOLOGY*, Ph.D. Thesis, Department of Mathematical Sciences, Indiana University of South Bend, 2007.
- [24] MURALI, S., “REVERSE ENGINEERING APPROACH FOR EFFECTIVE INDUSTRY INSTITUTE INTERACTION”. In: *International Conference on e-resources in higher education: Developments, Opportunities and Challenges*, pp. 346–350, Tiruchirappalli, Índia, 2011.
- [25] RUSU, R. B., “Clustering and Segmentation”, In: *Semantic 3D Object Maps for Everyday Robot Manipulation, STAR 85*, chap. 6, pp. 75–85, Springer Berlin Heidelberg, 2013.
- [26] NAKAI, H., “Linear-scaling divide-and-conquer correlation theory for treating large systems with high accuracy”, .
- [27] CAESAR, G. I., GITLBAUER, M., *Commentarii de bello Gallico*. Herder, 1884.
- [28] MACHIAVELLI, N. D. B. D., MORAES, E. V. D., *A arte da guerra (Dell’arte della guerra)*. L&PM, 2008.
- [29] SCHNEIDER, B., “Divide and Conquer”, Setembro 2015.
- [30] SHRIVASTAVA, S., “Marketing in Politics”, Junho 2014.
- [31] POSNER, E. A., SPIER, K., VERMEULE, A., *Divide and Conquer*, Tech. rep., Harvard Law School, Cambridge, MA, EUA, 2009.
- [32] FEOFILOFF, P., *Minicurso de Análise de Algoritmos*. 2nd ed. Dep. de Ciência da Computação, Instituto de Matemática e Estatística, Universidade de São Paulo: São Paulo, SP, Brasil, 2011.
- [33] WEBBER, H., *Divide and Conquer: Target Your Customers Through Market Segmentation*. John Wiley & Sons, Inc, 1998.
- [34] JONES, O., “Even in housing, the Tories stoop to divide and rule”, Setembro 2016.

- [35] RAJ, T. S., LAKSHMINARAYANAN, S., FORBES, J. F., “Divide and Conquer Optimization for Closed Loop Supply Chains”, *Industrial & Engineering Chemistry Research*, v. 52, n. 46, pp. 16267–16283, Novembro 2013.
- [36] WAGNER, T. C., PAPALAMBROS, P. Y., “A General Framework for Decomposition Analysis in Optimal Design”, 1993.
- [37] SARKAR, S. A., DONG, A. A., GERO, J. S. B., “A learning and inference mechanism for design optimization problem (re)-Formulation using singular value decomposition”. In: *ASME 2008 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, v. 4, pp. 97–106, American Society of Mechanical Engineers, 2008.
- [38] MICHELENA, N. F., PAPALAMBROS, P. Y., “Optimal Model-Based Decomposition of Powertrain System Design”, *Journal of Mechanical Design*, v. 117, n. 4, pp. 499, 1995.
- [39] WAGNER, T. C., *A General Decomposition Methodology for Optimal System Design*, Ph.D. Thesis, University of Michigan, 1993.
- [40] WAGNER, T. C., PAPALAMBROS, P. Y., *Decomposition Analysis of an Automotive Powertrain Design Problem: Model Development, Partitioning, and Optimization*, Tech. rep., 1996.
- [41] KRISHNAMACHARI, R. S., PAPALAMBROS, P. Y., “Optimal Hierarchical Decomposition Synthesis Using Integer Programming”, *ASME Journal of Mechanical Design*, v. 119, n. 4, pp. 440–447, 1997.
- [42] KRISHNAMACHARI, R. S., PAPALAMBROS, P. Y., “Hierarchical Decomposition Synthesis in Optimal Systems Design”, *ASME Journal of Mechanical Design*, v. 119, n. 4, pp. 448–457, 1997.
- [43] KRISHNAMACHARI, R. S., PAPALAMBROS, P. Y., “Optimal Design of A Hybrid Electric Powertrain System”, *Mechanics of Structures and Machines*, v. 25, n. 3, pp. 267–285, 1997.
- [44] MARIANO, A., JUNIOR, D. S., CORINA, C., LOBO, L., ROBERTA, P., AZEVEDO, P. D., “Compatibilização de projeto arquitetônico , estrutural e

sanitário : Uma abordagem teórica e estudo de caso .” *Revista Monografias Ambientais - REMOA - Centro do Ciências Naturais e Exatas*, v. 2, pp. 3236–3244, Março 2014.

- [45] TANENBAUM, A. S., *Sistemas Operacionais Modernos*. 3rd ed. Pearson, 2010.
- [46] WAGNER, T. C., PAPALAMBROS, P. Y., *Defining properties for decomposition in nonlinear programming*, Tech. rep., 1997.
- [47] ALLISON, J. T., KOKKOLARAS, M., PAPALAMBROS, P. Y., “Optimal Partitioning and Coordination Decisions in Decomposition-Based Design Optimization”. v. 131, p. 81008, 2007.
- [48] LIU, H. L., GU, F., ZHANG, Q., “Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems”, *IEEE Transactions on Evolutionary Computation*, v. 18, n. 3, pp. 450–455, 2014.
- [49] LI, K., KWONG, S., ZHANG, Q., DEB, K., “Inter-Relationship Based Selection for Decomposition Multiobjective Optimization”, , n. 5, pp. 1–11.
- [50] GERING, D., *Linear and Nonlinear Data Dimensionality Reduction*, Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, EUA, 2002.
- [51] ROWEIS, S. T., SAUL, L. K., “Nonlinear dimensionality reduction by locally linear embedding”, *SCIENCE*, v. 290, pp. 2323–2326, 2000.
- [52] JOHNSON, R. A., WICHERN, D. W., *Applied Multivariate Statistical Analysis*. 6th ed. Prentice-Hall, Inc.: Upper Saddle River, NJ, EUA, 2007.
- [53] CUNNINGHAM, *Dimension Reduction*, Tech. rep., School of Computer Science and Informatics, University College Dublin, Dublin, Rep. Irlanda, 2007.
- [54] MAATEN, L. J. P. V. D., *An Introduction to Dimensionality Reduction Using MatLab*, Tech. rep., Faculty of Humanities & Sciences, MICCIKAT, Maastricht University, Maastricht, Países Baixos, 2009.
- [55] FODOR, I. K., *A survey of dimension reduction techniques*, Tech. rep., Lawrence Livermore National Lab., California, EUA, 2002.

- [56] SHLENS, J., *A tutorial on principal component analysis*, Tech. rep., 2003.
- [57] HAN, J., KAMBER, M., PEI, J., *Data mining : concepts and techniques*. 3rd ed., v. 1. 2012.
- [58] TAN, P.-N., STEINBACH, M., KUMAR, V., *Introduction to Data Mining, (First Edition)*. Addison-Wesley Longman Publishing Co., Inc.: Boston, MA, EUA, 2005.
- [59] MATLAB, *R2015a*. The MathWorks Inc.: Natick, Massachusetts, EUA, 2015.
- [60] NG, A., “Dimensionality Reduction. Lecture Notes on Machine Learning”, 2013.
- [61] LÄUCHLI, P., “Jordan-Elimination Und Ausgleichung Nach Kleinsten Quadraten”, *Numer. Math.*, v. 3, n. 1, pp. 226–240, Dec. 1961.
- [62] NEWMAN, C. L., BLAKE, D. J., MERZ, C. J., *UCI Repository of machine learning databases*, Tech. rep., Dept. of Information and Computer Sciences, University of California, Irvine, California, EUA, 2002.
- [63] MOTTA, C. G., *Metodologia para Mineração de Regras de Associação Multiníveis Incluindo Pré e Pós-Processamento*, Ph.D. Thesis, COPPE, Universidade Federal do Rio de Janeiro, Rio de Janeiro, RJ, Brasil, 2010.
- [64] MACQUEEN, J. B., “Kmeans Some Methods for classification and Analysis of Multivariate Observations”, *5th Berkeley Symposium on Mathematical Statistics and Probability 1967*, v. 1, n. 233, pp. 281–297, 1967.
- [65] PELLEGG, D., MOORE, A., “X-means: Extending K-means with Efficient Estimation of the Number of Clusters”. In: *Proceedings of the Seventeenth International Conference on Machine Learning*, pp. 727–734, Morgan Kaufmann: San Francisco, CA, EUA, 2000.
- [66] MANNING, C. D., RAGHAVAN, P., SCHÜTZE, H., *Introduction to Information Retrieval*. Cambridge University Press, 2008.
- [67] MAHAJAN, M., NIMBHORKAR, P., VARADARAJAN, K., “The Planar k-Means Problem is NP-Hard”, In: *WALCOM: Algorithms and Computation*, pp. 274–285, Springer Berlin Heidelberg: Berlin-Heidelberg, Alemanha, 2009.

- [68] DASGUPTA, S., FREUND, Y., “Random projection trees for vector quantization”, Maio 2008.
- [69] ALOISE, D., DESHPANDE, A., HANSEN, P., POPAT, P., “NP-hardness of Euclidean sum-of-squares clustering”, *Machine Learning*, v. 75, n. 2, pp. 245–248, Maio 2009.
- [70] INABA, M., KATOH, N., IMAI, H., “Applications of weighted Voronoi diagrams and randomization to variance-based k-clustering: (extended abstract)”. In: *Proceedings of the tenth annual symposium on Computational geometry - SCG '94*, pp. 332–339, ACM Press: New York, New York, EUA, 1994.
- [71] LLOYD, S., “Least squares quantization in PCM”, *IEEE Transactions on Information Theory*, v. 28, n. 2, pp. 129–137, Março 1982.
- [72] ARTHUR, D., VASSILVITSKII, S., “K-means++: The Advantages of Careful Seeding”, *Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '07*, 2007.
- [73] ERDŐS, P., RÉNYI, A., “On random graphs. Publicationes Mathematicae, 6: 290, 1959.” *Publicationes Mathematicae*, 1959.
- [74] GRANOVETTER, M. S., “The strength of weak ties”, *American Journal of Sociology*, 1973.
- [75] SARKAR, S., DONG, A., “Community detection in graphs using singular value decomposition”, *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, v. 83, n. 4, 2011.
- [76] VIEIRA, V. D. F., XAVIER, C. R., EBECKEN, N. F. F., EVSUKOFF, A. G., “Performance evaluation of modularity based community detection algorithms in large scale networks”, *Mathematical Problems in Engineering*, v. 2014, pp. 1–15, 2014.
- [77] LALOË, T., SERVIEN, R., “The X-Alter algorithm: a parameter-free method to perform unsupervised clustering”, *Journal of modern applied statistical methods: JMASM*, v. 12, n. 1, pp. 90–102, 2013.

- [78] CAGAN, J., GROSSMANN, I. E., HOOKER, J., “A conceptual framework for combining artificial intelligence and optimization in engineering design”, 1997.
- [79] BLONDEL, V., GUILLAUME, J.-L., LAMBIOTTE, R., “Louvain method: Finding communities in large networks”, 2011.
- [80] NEWMAN, M. E. J., GIRVAN, M., “Finding and evaluating community structure in networks”, *Physical Review E: Statistica, Nonlinear, and Soft Matter Physics*, v. 69, n. 2, Agosto 2004.
- [81] BLONDEL, V. D., GUILLAUME, J.-L., LAMBIOTTE, R., LEFEBVRE, E., “Fast unfolding of communities in large networks”, *Journal of Statistical Mechanics: Theory and Experiment*, v. 10008, n. 10, pp. 6, 2008.
- [82] ALLISON, J. T., KOKKOLARAS, M., PAPALAMBROS, P. Y., “Optimal Partitioning and Coordination Decisions in Decomposition-Based Design Optimization”, *Journal of Mechanical Design*, v. 131, n. 8, pp. 81008, 2009.
- [83] HOCK, W., SCHITTKOWSKI, K., “Test Examples for Nonlinear Programming Codes”, In: BECKMAN, M., KÜNZI, H. (eds), *Lecture Notes in Economics and Mathematical Systems*, v. 187, Springer-Verlag: Berlin, Alemanha, 1981.
- [84] SCHERRER, A., “Louvain method MatLab implementation”, 2011.
- [85] PAPALAMBROS, P. Y., WILDE, D. J., *Principles of Optimal Design*. Cambridge University Press, 2000.
- [86] MICHELENA, N. F., AGOGINO, A. M., “Multi-Objective Hydraulic Cylinder Design”, *Journal of Mechanisms, Transmissions and Automation in Design*, v. 110, pp. 81–87, 1988.
- [87] MEZURA-MONTES, E., COELLO, C. A. C., “Constraint-handling in nature-inspired numerical optimization: Past, present and future”, *Swarm and Evolutionary Computation*, v. 1, n. 4, pp. 173–194, 2011.
- [88] DEB, K., “An efficient constraint handling method for genetic algorithms”, *Computer Methods in Applied Mechanics and Engineering*, v. 186, n. 2-4, pp. 311–338, 2000.

- [89] PELLEGG, T., “Notes on Bayesian Information Criterion Calculation for X-Means Clustering”, v. 2, n. 2000, pp. 1–5, 2012.
- [90] KAUFMAN, L., ROUSSEEUW, P. J., *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley & Sons, Inc.: Hoboken, NJ, EUA, 1990.
- [91] ROUSSEEUW, P. J., “Silhouettes: a graphical aid to the interpretation and validation of cluster analysis”, *Journal of Computational and Applied Mathematics*, v. 20, pp. 53–65, 1987.

UFJF	ALEXANDRE CANÇADO CARDOSO	DECOMPOSIÇÃO BASEADA EM MODELO DE PROBLEMAS DE OTIMIZAÇÃO DE PROJETO UTILIZANDO REDUÇÃO DE DIMENSIONALIDADE E REDES COMPLEXAS	2016
-------------	----------------------------------	--	-------------