

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA QUÍMICA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA QUÍMICA

**SPLINE DYNAMIC MATRIX CONTROLLER: UMA NOVA
FORMULAÇÃO DE CONTROLADORES PREDITIVOS LINEARES**

DISSERTAÇÃO DE MESTRADO

Matheus de Souza Escobar

Porto Alegre

2012

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA QUÍMICA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA QUÍMICA

**SPLINE DYNAMIC MATRIX CONTROLLER: UMA NOVA
FORMULAÇÃO DE CONTROLADORES PREDITIVOS LINEARES**

Matheus de Souza Escobar

Dissertação de Mestrado apresentada
como requisito parcial para obtenção do
título de Mestre em Engenharia

Área de concentração: Controle de
Processos

Linha de Pesquisa: Engenharia de
Sistemas – Projeto, Modelagem, Controle e
Otimização de Processos

Orientador:
Prof. Dr. Jorge Otávio Trierweiler

Porto Alegre

2012

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA QUÍMICA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA QUÍMICA

A Comissão Examinadora, abaixo assinada, aprova a Dissertação *Spline Dynamic Matrix Controller: Uma Nova*, elaborada por Matheus de Souza Escobar, como requisito parcial para obtenção do Grau de Mestre em Engenharia.

Comissão Examinadora:

Prof. Dr. Darci Odloak

Prof. Dr. Pedro Rafael Bolognese Fernandes

Dr. Luís Gustavo Soares Longhi

Resumo

No campo de simulação dinâmica de processos, o entendimento da natureza dinâmica de processos químicos industriais é fundamental quando questões de projeto e de sistemas de processos surgem. As abordagens atualmente aplicadas, como Resposta Degrau (RD), Função de Transferência (FT) e representação Discreta em Espaços de Estado (DEE) apresentam algumas desvantagens, principalmente relacionadas a questões de amostragem fixa e à carga computacional envolvida na sua simulação. Tendo isso em mente, uma nova abordagem é proposta, valendo-se da parametrização das entradas do sistema por *splines piecewise* cúbicas e da representação de sistemas em espaços de estado no domínio contínuo, onde se torna possível, a partir de poucos dados que revelem a informação dinâmica do processo, a simulação do mesmo ao longo de um horizonte. A essa abordagem foi dada o nome de *Spline Dynamic Matrix* (SDM), tratando-se de uma construção análoga à Matriz Dinâmica (Cutler and Ramaker 1980) proposta no início da década de 80, onde a multiplicação da variação das entradas no sistema por uma matriz que compreenda toda a informação dinâmica do sistema permite o cálculo rápido das saídas do mesmo. Suportado por essa metodologia, propõe-se juntamente a formulação de um novo controlador preditivo baseado em modelo (MPC) linear, *Spline Dynamic Matrix Controller* (SDMC), onde a redução do número de variáveis do processo pode levar a uma resolução mais rápida e mais robusta. Para corroborar as proposições apresentadas, são apresentados resultados referentes à simulação de processos dinâmicos e à comparação do desempenho do novo MPC proposto com um MPC baseado em resposta degrau (versão clássica), para um estudo de caso SISO (*Single Input Single Output*), e com um MPC baseado em espaços de estado, para o caso MIMO (*Multiple Input Multiple Output*), validando o potencial e versatilidade da técnica.

Abstract

In the field of dynamic simulation, an understanding of the dynamic behavior of chemical processes is important from both process design and process system perspectives. Several methodologies have been developed so far, such as Step Response (SR), Transfer Function (TF) and Discrete State Space representation (DSS). Such approaches, however, have some disadvantages, regarding simulation speed, computational load and fixed sampling. From this premise, a new approach is proposed, where the input parametrization by piecewise cubic splines is combined with state space representation in the continuous domain, guaranteeing, from few data that gathers all the process dynamic information, simulation along a certain horizon. Such representation is called Spline Dynamic Matrix (SDM), inspired in the likewise representation of the Dynamic Matrix (DM) (Cutler and Ramaker 1980) proposed in the early 80's. The main idea resides in a multiplication of the input variation by a matrix that can embrace all the dynamic data, allowing a fast output calculation. By using this methodology, a new linear model predictive controller (MPC) is also proposed, Spline Dynamic Matrix Controller (SDMC), where the variable diminishment could lead to a faster and more robust resolution. So that such propositions can be validated, results are presented regarding dynamic process simulation and comparison of the performance achieved by the MPC proposed in this dissertation. In order to accomplish that, one MPC based on SR representation, for a SISO (Single Input Single Output) case study, and another one based on state space representation, for a MIMO (Multiple Input Multiple Output) system, were used, confirming that the potential and versatility of SDM technique can be truly perceived.

*“Give a man a truth and he will think for a day.
Teach a man to reason and he will think for a lifetime.”*

Agradecimentos

Aos meus pais pelo apoio dado durante todos esses anos para que eu perseguisse os meus sonhos, pela crença inabalável no meu potencial e pelo amor e educação dado durante a minha vida sem os quais eu não seria metade do homem que hoje sou. Em especial ao meu pai pela constante lembrança do quão importante é a cultura e o conhecimento. Em especial a minha mãe pelos valiosos conselhos dados durante momentos de dúvida e incerteza.

Aos meus irmãos, pela relação de amizade e amor mantida durante todos esses anos. Em especial ao meu irmão, cuja conduta e personalidade servem de inspiração para que eu me torne uma pessoa melhor.

Ao restante dos meus familiares, pelo apoio incondicional durante essa jornada e reconhecimento constante dos meus potenciais.

Ao meu orientador, Prof. Dr. Jorge Otávio Trierweiler, pela sabedoria, conhecimento e exigência, cujo suporte foi fundamental para a realização desta dissertação.

Aos meus colegas e amigos da UFRGS, Renan, Alice, Cauã, Lara, Viviane e Rodrigo, cujas vitórias e apreensões foram sempre divididas, sem que o bom humor percesse. Em especial ao Rodrigo, pelos inúmeros dias de acaloradas discussões que contribuíram para o crescimento deste trabalho.

A todos os meus amigos que, de um jeito ou de outro, contribuíram para a realização deste trabalho durante este ano, seja através de uma partida de RPG, das aulas de japonês, dos treinos de beisebol ou de uma simples ida ao cinema no fim de semana.

Finalmente, à PETROBRAS, pelo fomento durante toda a pesquisa.

SUMÁRIO

Capítulo 1 – Introdução.....	1
1.1 Motivação.....	1
1.2 Estrutura da Dissertação.....	3
Capítulo 2 – Modelos Lineares para Controle Preditivo	5
2.1 Resposta Degrau (RD) / Matriz Dinâmica (DM)	5
2.2 Função de Transferência (FT)	7
2.3 Representação em Espaços de Estado.....	8
2.4 Simulações Através de Parametrização	10
Capítulo 3 – Splines.....	13
3.1 Curvas de Bézier	14
3.2 B-Splines	15
3.3 Hermite Splines	16
3.3.1 Piecewise Cubic Hermite Interpolator (PCHIP).....	19
3.3.2 Kochanek-Bartels Hermite Spline (TCB)	20
Capítulo 4 – Spline Dynamic Matrix - SDM	23
4.1 Representação Matricial das Entradas.....	24
4.2 Simulação das Saídas do Sistema / Estrutura SDM.....	26
4.3 Sistemas Variantes no Tempo	30
Capítulo 5 – Controlador Preditivo Baseado em Modelo (MPC)	31
5.1 Conceitos Gerais.....	31
5.2 Tipos de MPCs	33
5.3 DMC.....	35
5.4 DEE-MPC.....	35
5.5 Spline Dynamic Matrix Controller (SDMC).....	37
5.5.1 Ajuste do SDMC	39
Capítulo 6 – Resultados.....	41
6.1 Ferramenta Computacional	42
6.2 Estudos de Caso.....	42
6.2.1 Planta Laboratorial de Três Tanques Esféricos.....	42
6.2.2 Planta Laboratorial de Seis Tanques Esféricos	43
6.3 Resultado do Caso SISO - Planta de Três Tanques Esféricos.....	46
6.3.1 Número de Nós.....	57
6.3.2 Move Supression	58
6.3.3 Tempo Computacional.....	59
6.4 Resultado do Caso MIMO - Planta de Seis Tanques Esféricos	60
Capítulo 7 – Conclusão e Trabalhos Futuros	65
Referências	68

LISTA DE FIGURAS

Figura 2.1: Função Resposta Degrau Unitário.	5
Figura 3.1: Representação esquemática da obtenção de uma função quadrática pelo algoritmo de Casteljau.....	14
Figura 3.2: Representação esquemática da obtenção de uma função cúbica pelo algoritmo de Casteljau, partindo de funções quadráticas parametrizadas.....	15
Figura 3.3: Representação esquemática de B-splines.	16
Figura 3.4: Representação esquemática das condições de contorno utilizadas	17
Figura 3.5: Representação esquemática do cálculo dos deltas em cada intervalo.	19
Figura 3.6: Efeito dos parâmetros da TCB na geração das <i>splines</i> para o nó vermelho em destaque, para tensão igual a: a) 1 e b) -1; continuidade igual a: c) 1 e d) -1; e bias igual a: e) 1 e f) -1.	20
Figura 3.7: Curva correspondente a uma resposta degrau de um sistema de primeira ordem comparando o desempenho da PCHIP com a Hermite TCB, padrão e otimizada, conforme os dados apresentados na tabela 3.1.	21
Figura 3.8: Curva correspondente a uma resposta degrau de um sistema de segunda ordem comparando o desempenho da PCHIP com a Hermite TCB, padrão e otimizada, conforme os dados apresentados na tabela 3.1.	22
Figura 4.1: Representação esquemática da distinção entre nós e pontos discretizados ao longo do horizonte de simulação para um conjunto de entradas sigmoidal.....	24
Figura 4.2: Composição das parcelas X_0 , X_U e X_{U0} na simulação de um degrau unitário na vazão de entrada de uma planta laboratorial de três tanques e seu efeito no nível do terceiro tanque.	29
Figura 5.1: Representação Esquemática de um MPC.	32
Figura 5.2: Ilustração do horizonte de predição e horizonte de controle. (Figura adaptada de Matlab® Model Predictive Control Toolbox)	33
Figura 5.3: Histórico aproximado dos Algoritmos dos MPCs.....	34
Figura 6.1: Representação esquemática de uma planta laboratorial de três tanques esféricos.....	43
Figura 6.2: Planta laboratorial de seis tanques esféricos.	44
Figura 6.3: Resposta a um degrau unitário na entrada F_0 para três curvas distintas, avaliando o impacto do número e posição dos nós na qualidade da resposta.....	46
Figura 6.4: Resposta a um perfil de entrada F_0 sigmoide para três curvas distintas, avaliando o impacto do número e posição dos nós na qualidade da resposta.....	47
Figura 6.5: Resposta a um perfil de entrada F_0 oscilatório para três curvas distintas, avaliando o impacto do número e posição dos nós na qualidade da resposta.....	49
Figura 6.6: Resposta a um degrau unitário na entrada F_0 para três diferentes pontos de operação em uma planta laboratorial, mostrando a diferença entre o modelo linear (linhas tracejadas) e não linear (linhas sólidas).	51
Figura 6.7: Simulação da planta de três tanques expressando o comportamento de dois controladores projetados para o PO 3 aplicados no PO 3, quando submetidos à mudança de <i>set point</i> e distúrbio na carga unitários.....	53
Figura 6.8: Predição do MPC amostragem a amostragem, indicando, para a entrada, a posição otimizada dos nós em cada iteração.	54

Figura 6.9: Simulação da planta de três tanques expressando o comportamento de dois controladores projetados para o PO 3 aplicados no PO 1, quando submetidos à mudança de <i>set point</i> e distúrbio na carga unitários.	55
Figura 6.10: Simulação da planta de três tanques expressando o comportamento de dois controladores projetados para o PO 3 aplicados no PO 2, quando submetidos à mudança de <i>set point</i> e distúrbio na carga unitários.	55
Figura 6.11: Simulação da planta de três tanques expressando o comportamento de dois controladores projetados para o PO 3 aplicados no PO 3, quando submetidos à mudança de <i>set point</i> e distúrbio na carga unitários com restrição superior na entrada de 9,3 L/min.	56
Figura 6.12: Simulação da planta de três tanques expressando o comportamento de dois controladores projetados para o PO 3 aplicados no PO 3, quando submetidos à mudança de <i>set point</i> e distúrbio na carga unitários com restrição inferior na entrada de 8 L/min.	56
Figura 6.13: Simulação da planta de três tanques expressando o comportamento de dois controladores projetados para o PO 3 aplicados no PO 1, onde dois SDMCs com número de pontos distintos são empregados.	57
Figura 6.14: Simulação da planta de três tanques expressando o comportamento de controladores projetados para o PO 3 aplicados no PO 3, variando o <i>Move Supression</i>	58
Figura 6.15: Resposta degrau na planta de seis tanques, comparando o modelo linear com o não linear.	61
Figura 6.16: Parâmetros das SDM gerada para a reconstrução de perturbações do tipo sigmoide, com dinâmicas diferentes.	62
Figura 6.16: Simulação da planta de seis tanques expressando o comportamento de dois controladores projetados para o PO em questão: o DEE-MPC (linha tracejada) e o SDMC (linha sólida).	64

LISTA DE TABELAS

Tabela 3.1: Parâmetros da TCB otimizada para as funções, uma análoga a um sistema de primeira ordem e outra, a um de segunda ordem.	22
Tabela 6.1: Parâmetros da planta de três tanques esféricos.	43
Tabela 6.2: Parâmetros da planta de seis tanques esféricos.....	45
Tabela 6.3: Caracterização de fases do sistema.	45
Tabela 6.4: Parâmetros das SDMs geradas para uma perturbação degrau.	46
Tabela 6.5: Parâmetros das SDMs geradas para a reconstrução de uma perturbação do tipo sigmoide.	47
Tabela 6.6: Somatório do erro quadrático para as perturbações do tipo sigmoide reconstruídas a partir da SDM.....	48
Tabela 6.7: Parâmetros das SDMs geradas para a reconstrução de uma perturbação oscilatória.	49
Tabela 6.8: Somatório do erro quadrático para as perturbações oscilatórias reconstruídas a partir da SDM.....	50
Tabela 6.9: Valores estacionários das alturas em cada tanque e da vazão de entrada para os três pontos de operação.	52
Tabela 6.10: Parâmetros base empregados para o SDMC e o MPC-STEP.....	53
Tabela 6.11: Parâmetros empregados para o SDMC.	58
Tabela 6.12: Parâmetros base empregados para o SDMC e o MPC-STEP.....	59
Tabela 6.13: Tempo de simulação total e da parcela referente à otimização para dois controladores em função do tempo de amostragem.	60
Tabela 6.14: Tempo de simulação total e da parcela referente à otimização para dois controladores em função do tempo de amostragem.	60
Tabela 6.15: Valores estacionários das alturas em cada tanque e da vazão de entrada.	61
Tabela 6.5: Parâmetros das SDMs geradas para a reconstrução de uma perturbação do tipo sigmoide.	62
Tabela 6.16: Parâmetros base para o DEE-MPC.	63
Tabela 6.10: Parâmetros base empregados para o SDMC.	63

ABREVIações

<i>DEE</i>	Representação Discreta em Espaço de Estado (<i>Discrete State Space Representation</i>)
<i>DEE-MPC</i>	Controlador Preditivo baseado na representação discreta em espaços de estado (<i>Discrete State Space Model Predictive</i>
<i>DM</i>	Matriz Dinâmica (<i>Dynamic Matrix</i>)
<i>DMC</i>	Controlador Baseado em Matriz Dinâmica (<i>Dynamic Matrix Controller</i>)
<i>FT</i>	Função de Transferência (<i>Transfer Function</i>)
<i>GPC</i>	Controlador Preditivo Generalizado (<i>Generalized Predictive Controller</i>)
<i>IDCOM</i>	Identificação e Comando (<i>Identification and Command</i>)
<i>LQG</i>	Controlador Quadrático Gaussiano Linear (<i>Linear Quadratic Gaussian Controller</i>)
<i>MPC</i>	Controlador Preditivo Baseado em Modelo (<i>Model Predictive Controller</i>)
<i>MPC-STEP</i>	Controlador Preditivo Baseado em Resposta Degrau implementado em MatLab® (<i>MatLab® implemented Controller based on Step Response</i>)
<i>MPHC</i>	Controlador Preditivo Baseado em Modelo Heurístico (<i>Model Predictive Heuristic Controller</i>)
<i>PCHIP</i>	Interpolador de Hermite Cúbico <i>Piecewise</i> (<i>Piecewise Cubic Hermite Interpolator</i>)
<i>PO</i>	Ponto de Operação (<i>Operating Point</i>)
<i>QP</i>	Programação Quadrática (<i>Quadratic Programming</i>)
<i>RD</i>	Resposta Degrau (<i>Step Response</i>)
<i>RPN</i>	Número de Desempenho Robusto (<i>Robust Performance Number</i>)
<i>SDM</i>	Matriz Dinâmica Baseada em <i>Splines</i> (<i>Spline Dynamic Matrix</i>)
<i>SDMC</i>	Controlador Preditivo Baseado em SDM (<i>SDM based Controller</i>)
<i>SMOC</i>	Controlador e Otimizador Multivariável Shell (<i>Shell Multivariable Optimizer and Controller</i>)
<i>TCB</i>	Hermite <i>Spline</i> de Kochanek-Bartels (<i>Kochanek-Bartels Hermite Spline</i>)

NOTAÇÃO E SIMBOLOGIA

A	Matriz de estados – Espaços de Estado (<i>State Matrix</i>)
$a - h$	Coefficientes das funções cúbicas paramétricas de uma <i>spline</i> genérica
B	Matriz de entradas – Espaços de Estado (<i>Input Matrix</i>)
C	Matriz de saídas – Espaços de Estado (<i>Output Matrix</i>)
c	Indicação de domínio contínuo
CD	Coefficiente de descarga
<i>Coef</i>	Matriz de Coeficientes das <i>splines</i>
D	Matriz de Transmissão Direta – Espaços de Estado (<i>Feedthrough Matrix</i>) / Diâmetro do Tanque
d	Indicação de domínio discreto / Distúrbio medido do sistema
dU	Matriz de derivadas analíticas das equações parametrizadas
e	Matriz exponencial
E_u	Termo de variação das entradas
E_y	Erro entre o valor desejado e o valor predito pela planta
F	Vazão de entrada no tanque
F_{obj}	Função objetivo do problema de otimização
H	Vetor de alturas dos nós
<i>Hess</i>	Hessiana da função objetivo
I	Sequência de respostas do tipo impulso
j	Variável paramétrica auxiliar
L	Nível do tanque / Matriz de Escalonamento - RPN
M	Horizonte de Controle
m^{in}	Tangente de entrada em um nó
m^{out}	Tangente de saída em um nó
P	Horizonte de Predição
p	Ponto cartesiano no espaço
Q	Parcela linear do problema de otimização QP
R	Resposta livre do sistema / Matriz de Escalonamento - RPN
r	Número de pontos em cada intervalo
<i>Ref</i>	Vetor de valores ou trajetória de referência para as variáveis de saída
s	Variável do domínio de Laplace
Sd	Matriz do efeito da variação dos distúrbios medidos na saída do sistema

S_u	Matriz dinâmica da representação em espaços de estado
S_x	Matriz do efeito da variação dos estados na saída do sistema
T	Vetor que confere característica cúbica à <i>spline</i>
t	Tempo de simulação
$TCBMat$	Matriz auxiliar para incorporação da tangente de entrada e saída na metodologia SDM
T_s	Tempo de amostragem
U	Variável de entrada do sistema
u	Expressão analítica para as entradas em função do tempo
U_m	Parâmetro de sintonia do DEE-MPC através da metodologia RPN
w	Discrepância entre o modelo linear e o não linear
W	Pontos de operação do sistema.
x	Variável de estado do sistema
X	Fração da vazão de entrada F desviada para alimentação cruzada
X_0	Matriz de resposta inercial
X_U	Matriz do efeito das entradas parametrizadas na saída
X_{U0}	Matriz de resposta inercial acumulada
Y	Variável de saída do sistema
Y_{BIAS}	Valor base das variáveis de saída
Y_d	Vetor de distúrbios medidos constante
Y_{INER}	Respostas inerciais acumuladas das variáveis de saída
Y_m	Parâmetro de sintonia do DEE-MPC através da metodologia RPN
Y_{PLAN}	Valor medido na planta a cada amostragem
Y_{PRED}	Valor predito pela simulação linear da planta a cada amostragem
Z	Matriz de entradas iniciais a cada entrenó
z	Número de nós
Δ	Delta das secantes dos nós nos intervalos
γ	Continuidade
Γ	Peso do termo de erro das variáveis de saída
δ	Delta entre coordenadas temporais de dois nós subsequentes
Λ	Peso do termo de variação das entradas
τ	Tensão
ω	Bias

Capítulo 1 – Introdução

1.1 Motivação

Um maior entendimento da natureza dinâmica de processos químicos industriais é de grande importância quando questões relacionadas a projeto, assim como sistemas, de processos surgem. Tal entendimento permite que o engenheiro de processo preveja comportamentos, analise tendências e atue preventivamente, evitando assim instabilidades na planta, o que garante, do ponto de vista de operação de planta, eficiência e segurança, contemplando também qualidade e confiabilidade de produção.

Partindo dessa premissa, diversas metodologias vêm sendo desenvolvidas, promovendo, através de modelagem matemática, a simulação dinâmica de processos ao longo do tempo. Dentre uma variada gama de técnicas desenvolvidas, Resposta Degrau (RD) (Liu e Gao 2011) Função de Transferência (FT) (Luyben 1990) e representação Discreta em Espaços de Estado (DEE) (Maciejowski 2002) são as metodologias mais recorrentes.

Em se tratando de modelagem de processos, entretanto, determinados pontos relacionados à carga computacional, velocidade de simulação e aderência do modelo devem ser levados em consideração. Uma vez que tais questões são analisadas a partir das abordagens citadas anteriormente, percebe-se que são simulações dispendiosas, em termos de tempo utilizado, apesar de simularem com razoável acurácia sistemas distintos. Isso se deve em grande parte à discretização do horizonte de simulação, associado juntamente com o fato de que as entradas do sistema em cada instante de tempo são as variáveis do problema, acarretando em simulações tão mais lentas, quanto maior for a complexidade do modelo.

Associadas diretamente a essa questão, problemas de otimização dinâmica também são afetados pelas variáveis de entrada disponíveis ao longo do tempo para a resolução, como é o caso de controladores preditivos baseados em modelos (MPC), por exemplo. Deseja-se determinar nesse tipo de problema quais são as

entradas que levam as saídas a valores desejados. A quantidade dessas variáveis afeta o tamanho das matrizes inerentes à solução do algoritmo de otimização, aumentando conseqüentemente a carga computacional envolvida, analogamente à simulação de processos.

O que aconteceria, entretanto, se apenas alguns meros pontos no espaço pudessem reunir toda a informação dinâmica do processo, utilizando-a para uma representação eficiente das entradas do sistema ao longo do tempo, sem perder a confiabilidade quanto à qualidade da simulação? Tal proposição é apresentada nesta dissertação ao introduzir o conceito da *Spline Dynamic Matrix* (SDM). Ao representar a entrada do sistema ao longo de um horizonte definido por alguns poucos pontos no espaço, *splines*, *i.e.*, funções polinomiais cúbicas, podem interpolar esses dados, a fim de criar expressões analíticas para essas entradas. As variáveis do problema, logo, tornam-se simplesmente a posição desses pontos no espaço. Ao combinar *splines* (Boor 2001) com a representação em espaços de estado no domínio contínuo (Ogata 2003), uma matriz dinâmica que é dependente de um número significativamente menor de variáveis apresenta-se como uma nova possibilidade. A representação baseada nessas ideias é chamada de SDM, sendo o tópico principal nesta dissertação.

O paralelo entre a SDM e o conceito de matriz dinâmica derivado da RD (Cutler e Ramaker 1980) mostram-se evidentes, visto que o objetivo das duas formas de modelo é equivalente. Deseja-se obter a simulação do problema apenas como uma multiplicação entre duas matrizes, uma delas representando o comportamento dinâmico do sistema e a outra, as entradas ao longo do tempo. No caso da SDM, contudo, propõe-se determinar o valor da saída do sistema em um horizonte discretizado, valendo-se para tanto de apenas alguns pontos esparsos nesse horizonte.

Ao se estabelecer a metodologia, aplicações diversas mostram-se também possíveis. Uma extensão natural da SDM seria a sua aplicação em controladores preditivos baseados em modelos (MPC) lineares. O problema de otimização inerente à concepção de um MPC depende enormemente do número de variáveis presentes no sistema, afetando diretamente o tamanho da hessiana empregada para a resolução desse problema, o que afeta, portanto, a carga computacional envolvida. A redução de variáveis, logo, auxiliaria consideravelmente no aumento da velocidade de simulação do problema em um primeiro momento, o que permitiria lidar com sistemas mais complexos, como, por exemplo, sistemas rígidos, sistemas de elevada ordem, tempo morto, etc.

Em se tratando, ainda, de características associadas ao MPC, é de conhecimento notório que a robustez desses controladores é considerada baixa (Camacho e Bordons 2004), onde tal característica está intimamente ligada com a estabilidade e performance dos mesmos. Tomando como base a SDM, o reduzido número de pontos empregados para a formulação de uma nova abordagem de um MPC pode conferir ao sistema tal robustez, atacando um problema que, ao longo do desenvolvimento de MPCs, mostra-se recorrente.

Ao longo deste trabalho, a metodologia SDM será apresentada, investigada e testada, a fim de estabelecer claramente as suas vantagens e o seu potencial, tanto como ferramenta de simulação de processos dinâmicos, como base de simulação linear de processos para o desenvolvimento de controladores preditivos lineares.

1.2 Estrutura da Dissertação.

Nesta dissertação, será apresentada inicialmente no capítulo 2 uma revisão bibliográfica referente às metodologias empregadas para simulação dos modelos dinâmicos, de modo a apresentar as vantagens e desvantagens de cada abordagem, apresentando como fechamento do capítulo quais as características extraídas das diversas metodologias existentes que motivariam a proposição da SDM.

O capítulo 3 apresenta o conceito de *Splines*, mostrando os diferentes tipos existentes, indicando quais os parâmetros empregados em cada abordagem, assim como suas aplicações, vantagens e limitações. Focando-se na problemática de modelos dinâmicos, procede-se, no mesmo capítulo, à escolha de qual *spline* seria a mais adequada para a aplicação em questão, sendo esta a *spline* base empregada para o desenvolvimento da metodologia da SDM propriamente dita.

O capítulo 4 trata da metodologia que origina a SDM. A partir da combinação dos conhecimentos apresentados nos capítulos 2 e 3, introduz-se em detalhes como a SDM é concebida, apresentando quais são os parâmetros de ajuste existentes.

No capítulo 5, a aplicação da SDM para o desenvolvimento de uma nova formulação de MPC é apresentada, tratando inicialmente de uma breve revisão referente ao assunto, que culmina na apresentação breve do *Dynamic Matrix Control* (DMC) (Cutler e Ramaker 1980), que serve de inspiração para o MPC proposto nesta dissertação. Seguindo tal apresentação, procede-se à formulação do MPC, salientando suas peculiaridades e ajustes.

O capítulo 6 introduz a metodologia empregada para a realização dos testes, a ferramenta computacional empregada, e descreve em detalhes os dois estudos de caso empregados nesse capítulo, que são baseados em plantas laboratoriais de tanques esféricos. Ainda nesse capítulo, os resultados referentes à simulação dinâmica são apresentados para os dois sistemas supracitados, seguido por resultados validando a teoria do MPC empregada. Os resultados apresentados são comparados com um MPC que se vale da resposta degrau (RD) implementado em MatLab® e outro, baseado na teoria de espaços de estados, implementado em Python.

O capítulo 7 apresenta as conclusões finais e sugestões de trabalhos futuros. Finalmente, dois apêndices complementam esta dissertação.

Capítulo 2 – Modelos Lineares para Controle Preditivo

Antes de apresentar o conceito da SDM propriamente dita, é importante rever as técnicas de simulação de processos dinâmicos existentes, apresentando as suas vantagens e desvantagens, a fim de entender mais sobre o real potencial existente na SDM. As seguintes subseções apresentam tais modelos.

2.1 Resposta Degrau (RD) / Matriz Dinâmica (DM)

A Resposta Degrau (*Step Response*) é gerada a partir de um conceito intuitivo. Ao aplicar um degrau para cada entrada na planta, grava-se a resposta em malha aberta de cada uma das variáveis de saída. O efeito de cada entrada é capturado, sendo, logo, possível inferir a resposta do sistema para qualquer padrão de entradas que possam ser representados como uma sequência de degraus ao longo do tempo. Importante salientar que esse método é tão mais eficiente, quanto mais linear e sem ruído for o efeito. Uma representação esquemática de um degrau unitário aplicado a essa metodologia pode ser visto na figura 2.1.

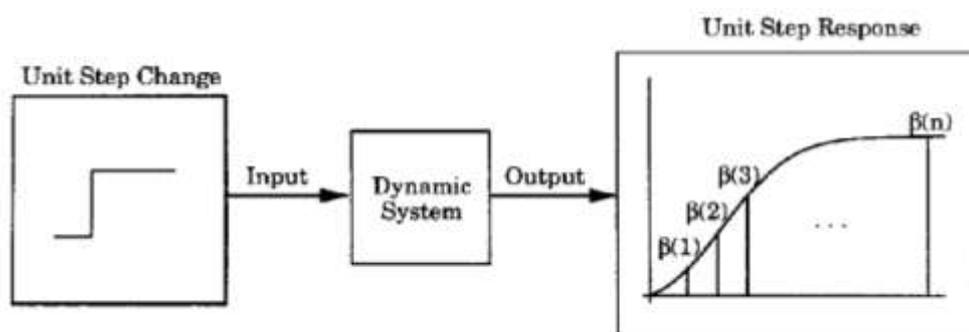


Figura 2.1: Função Resposta Degrau Unitário.

FONTE: (Ogunnaike e Ray 1994)

Onde $\beta(i)$ representa o efeito que o degrau na entrada tem na saída ao longo de um horizonte discretizado. Partindo do pressuposto que tais respostas são proporcionais a um degrau unitário, é possível simular a saída de um determinado sistema dinâmico ao longo desse horizonte (Ogunnaike e Ray 1994). Cada variação na entrada atua, logo, como um degrau, tendo um efeito nos valores futuros da saída. Isso implica que, em um determinado instante, uma saída é dependente de diversos degraus aplicados em instantes anteriores. A expressão para o cálculo das saídas, portanto, é tal como se segue em (2.1) (Maciejowski 2002), sendo nada mais que a versão integrada da resposta impulso do sistema:

$$Y(t) = \sum_{i=0}^t \beta(i) \Delta U(t - i) \quad (2.1)$$

Onde ΔU representa um degrau aplicado no sistema a cada instante i . A superposição desses degraus, aliada com a representação mostrada na figura 2.1, culmina na representação de uma matriz dinâmica do sistema, chamada de *Dynamic Matrix* (DM), que levaria ao desenvolvimento do *Dynamic Matrix Controller* (DMC) no início da década de 80 (Cutler e Ramaker 1980). A equação (2.2) mostra essa matriz para M entradas aplicadas ao longo de um horizonte P de simulação:

$$DM = \begin{bmatrix} \beta(1) & 0 & 0 & \dots & 0 \\ \beta(2) & \beta(1) & 0 & \dots & 0 \\ \beta(3) & \beta(2) & \beta(1) & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \beta(M) & \beta(M-1) & \beta(M-2) & \dots & \beta(1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \beta(P) & \beta(P-1) & \beta(P-2) & \dots & \beta(P-M+1) \end{bmatrix} \quad (2.2)$$

A expressão final, logo, toma a seguinte forma, conforme (2.3):

$$\Delta Y_{[k+P]}^{[k+1]} = DM \Delta U_{[k+M]}^{[k+1]} \quad (2.3)$$

A ideia de uma matriz fundamental capturando toda a informação dinâmica foi a motivação para o desenvolvimento da SDM. Apesar de essa dissertação ser focada em uma representação mais compacta e concisa de modelos dinâmicos, o princípio geral mantém-se o mesmo, mudando apenas a quantidade de informação armazenada para a simulação. Somente as variáveis significantes, *i.e.*, que guardam a informação dinâmica do processo são utilizadas, valendo-se para tanto das *splines*. Uma vez que tal conhecimento possa ser representado por uma matriz base, e as entradas sejam dadas ao longo do tempo, a simulação das saídas do processo pode ser obtida apenas por uma mera multiplicação matricial.

A representação da SDM, entretanto, não depende de degraus a serem dados no sistema. Em vez disso, a SDM é dependente de funções polinomiais baseadas em *splines* ao longo do tempo, combinadas com a teoria de espaços de estado. A

seção a seguir trata de uma representação alternativa de sistemas, Funções de Transferência, seguida pela representação em Espaços de Estado.

2.2 Função de Transferência (FT)

A utilização de FT, valendo-se para tanto do conceito de Transformada de Laplace, possibilita simplificações quanto à notação e computação do modelo de um sistema (Luyben 1990). A partir de um sistema de equações diferenciais ordinárias no domínio do tempo, $f(t)$, são obtidas expressões algébricas em função de uma variável característica do domínio de Laplace, também chamado de domínio da frequência, s . Da definição tem-se que, de acordo com (2.4):

$$\mathcal{L}\{f(t)\} \equiv \int_0^{\infty} f(t)e^{-st} dt \equiv F(s) \quad (2.4)$$

Esse tipo de representação traz diversas vantagens quanto à representação de diferenciação, integração, tempo morto, etc. As equações diferenciais no domínio do tempo, logo, podem ser substituídas pela manipulação de equações algébricas, o que possibilita, em se tratando de controle de processos e simulação dinâmica de processos, estabelecer facilmente relações entre as variáveis de entrada e saída do mesmo.

É, dentre as metodologias existentes, aquela que apresenta a representação mais elegante. A representação em FT de um processo é realizada de maneira compacta, onde poucos parâmetros são necessários para a construção da função, quando comparado com a DM. Além disso, a estrutura da função gerada permite que diversas características do sistema sejam analisadas, como estabilidade, característica da resposta, etc.. Tanto características dinâmicas, através de polos e zeros, como estacionárias, através do ganho da planta, são rapidamente analisadas (Åström e Murray 2008).

No final da década de 80, trabalhos foram publicados no desenvolvimento de um controlador empregando FT no domínio discreto, chamado de *Generalized Predictive Controller* (GPC) (Clarke, Mohtadi e Tuffs 1987). Apesar de ter sido publicado posteriormente aos controladores não paramétricos, como o DMC, teve grande repercussão, devido ao reduzido número de parâmetros necessários ao modelo empregado, à herança de características de controle adaptativo, e às técnicas de predição *multi-step*, otimização dinâmica e correção retroalimentada presentes nessa abordagem (Tatjewski 2010).

Em se tratando de ferramentas para simulação de processos dinâmicos, duas escolas se destacam atualmente, uma voltada para o desenvolvimento de técnicas empregando FT e outra se focando na representação em espaços de estados. Para o trabalho em questão, FT não será utilizada, uma vez que a representação em espaços de estado se mostrou mais simples e adequada. É importante salientar que ambas as representações são intercambiáveis, podendo se explorar em trabalhos futuros essa outra vertente.

2.3 Representação em Espaços de Estado

A representação em espaços de estado surge naturalmente da linearização de um modelo matemático que representa um sistema através de equações diferenciais, onde a relação entre variáveis fundamentais, chamadas de estados, e as entradas do sistema são estabelecidas com as saídas do sistema, através de um conjunto de matrizes.

Estados representam a mínima informação dinâmica necessária para a correta determinação das saídas de um processo. O equacionamento necessário à técnica pode ser visto em (2.5) e (2.6), sendo derivado usualmente de um conjunto de equações fenomenológicas $F: \mathbb{R}^{m+s} \rightarrow \mathbb{R}^n$, representando m equações com n entradas e s estados.

$$\dot{x} = Ax(t) + Bu(t) \quad (2.5)$$

$$y(t) = Cx(t) + Du(t) \quad (2.6)$$

Onde x é o vetor de estados, u representa o vetor das entradas e y , as saídas desejadas. Além disso, as matrizes jacobianas desse conjunto de equações, A e B , estabelecem as relações entre os estados do sistema atuais com os estados passados e as entradas, respectivamente. A matriz C expressa a saída do processo como uma combinação linear dos estados. Finalmente, a matriz D expressa o efeito instantâneo das entradas nas saídas, também chamado de *feedthrough*, sendo que a sua utilização se restringe a casos específicos, não contemplados neste trabalho.

Uma vez que essa equação é resolvida para o domínio contínuo, de um instante t_0 até t , a representação em espaços de estado, como função da entrada u , é tal como se segue em (2.7) (Ogata 2003), que é combinada com a expressão em (2.6) para a simulação do sistema.

$$x(t) = e^{A(t-t_0)}x_0 + \int_{t_0}^t e^{A(t-\tau)}Bu(\tau) d\tau \quad (2.7)$$

Onde, sabendo que $A(t-t_0)$ e $A(t-\tau)$ são matrizes, e^A representa a matriz exponencial desses termos. Essa representação em espaços de estado, entretanto, é usualmente realizada no domínio discreto. A partir dessa premissa, algumas transformações devem ser feitas, para que matrizes discretas equivalentes em espaços de estados possam ser determinadas (A_d , B_d , C_d e D_d), conforme se segue em (2.8) e (2.9):

$$A_d = e^{AT_s} \quad (2.8)$$

$$B_d = (e^{AT_s} - I)A^{-1}B \quad (2.9)$$

Onde T_s é o tempo de amostragem e I é a matriz identidade. C e D se mantêm inalterados na representação discreta, i.e., $C_d=C$ e $D_d=D$. Logo, conforme expresso em (2.10) e (2.11):

$$\Delta x_{k+1} = A_d \Delta x_k + B_d \Delta u_k \quad (2.10)$$

$$\Delta y_{k+1} = C_d \Delta x_{k+1} + D_d \Delta u_{k+1} \quad (2.11)$$

Aplicando essas equações ao longo do horizonte de simulação, uma relação de recorrência surge onde as variações dos estados passados afetam a resposta futura da saída do sistema (Camacho e Bordons 2004). Dessa relação, uma matriz dinâmica equivalente pode ser obtida, chamada de matriz Su , representada por (2.12), a qual foi representada para o caso de $D_d=0$. Maiores detalhes podem ser vistos em Duraiski (2001):

$$Su = \begin{bmatrix} 0 & 0 & \dots & 0 \\ C_d B_d & \ddots & \dots & \vdots \\ \sum_{j=1}^2 C_d A_d^{j-1} B_d & \ddots & 0 & 0 \\ \vdots & \ddots & C_d B_d & 0 \\ \sum_{j=1}^{M-2} C_d A_d^{j-1} B_d & \dots & \sum_{j=1}^2 C_d A_d^{j-1} B_d & C_d B_d \\ \sum_{j=1}^{M-1} C_d A_d^{j-1} B_d & \dots & \sum_{j=1}^3 C_d A_d^{j-1} B_d & \sum_{j=1}^2 C_d A_d^{j-1} B_d \\ \vdots & \ddots & \vdots & \vdots \\ \sum_{j=1}^{P-2} C_d A_d^{j-1} B_d & \dots & \sum_{j=1}^{P-M+1} C_d A_d^{j-1} B_d & \sum_{j=1}^{P-M} C_d A_d^{j-1} B_d \\ \sum_{j=1}^{P-1} C_d A_d^{j-1} B_d & \dots & \sum_{j=1}^{P-M+2} C_d A_d^{j-1} B_d & \sum_{j=1}^{P-M+1} C_d A_d^{j-1} B_d \end{bmatrix} \quad (2.12)$$

Onde M é o horizonte de controle. A expressão final para o cálculo saída toma a seguinte forma, de acordo com (2.13):

$$Y_{\begin{bmatrix} k+P \\ k+1 \end{bmatrix}} = Su \cdot \Delta U_{\begin{bmatrix} k+M \\ k+1 \end{bmatrix}} \quad (2.13)$$

Onde se assume que as entradas U só variam até o instante M , sendo a predição realizada até P . Essa representação discreta em espaços de estado (DEE) é diferente do que se propõe para esta dissertação. Como se está tratando da representação das entradas por uma função contínua, a representação discreta em espaços de estado perde o seu propósito. Partindo de *splines* polinomiais cúbicas que representem a entrada, vale-se de (2.7) para a simulação do sistema em espaços de estado no domínio contínuo. Isso representa um número significativamente menor de variáveis. O objetivo principal por trás da SDM é justamente reduzir o número de variáveis do problema, aumentando a velocidade de cálculo sem perda de qualidade.

A representação em RD apresenta claras desvantagens, no que diz respeito ao armazenamento das respostas. É uma forma ineficiente de representação de modelos, em se tratando dos requisitos de armazenamento, especialmente quando se trabalha com sistemas rígidos multivariáveis (Maciejowski 2002). A representação em espaços de estado, por outro lado, por si só traz vantagens, como, por exemplo, não haver necessariamente a necessidade de se perturbar o sistema, além de, evidentemente, haver uma modelagem matemática que permita a associação de funções polinomiais contínuas ao cálculo das saídas. A concepção

da SDM, logo, fundamenta-se em combinar as características da DM com espaços de estado.

2.4 Simulações Através de Parametrização

A ideia de se simular sistemas com um número menor de variáveis, diminuindo o tempo de computação e, conseqüentemente, aumentando a velocidade de simulação, não é nova. Diversos trabalhos já foram realizados, tratando de diferentes maneiras de parametrizar as entradas em um sistema, tentando manter a representatividade dos dados.

Um das técnicas mais simples a ser empregada para redução do número de variáveis de entrada ao longo de um horizonte de controle é o *blocking*. Essa abordagem consiste em manter uma mesma entrada para mais de um instante discretizado, criando blocos de entradas com valores iguais (Maciejowski 2002). Em termos de simulação dinâmica, essa estrutura não é tão aplicada, visto que a representação da tendência das entradas do sistema é prejudicada. É utilizado, entretanto, amplamente em controladores preditivos, onde essa discrepância não é tão significativa (Gondhalekar e Imura 2010, Cagienard et al. 2007).

O *blocking* introduz características de robustez ao sistema, em virtude do menor número de variáveis de decisão no problema de otimização, prejudicando, entretanto, o desempenho do MPC, pois as respostas se tornam mais lentas. A SDM apresenta um comportamento análogo ao *blocking*, onde blocos de entradas seguem um mesmo comportamento. A diferença, entretanto, reside na utilização de uma função *piecewise* cúbica entre cada intervalo, acarretando em um perfil de entradas e não em valores constantes. Isso possibilita a inserção de robustez, com minimização de perda do desempenho do controlador.

Em se tratando da aplicação de *splines* e polinômios para simulação dinâmica, esses conceitos tem sido utilizados amplamente na área de computação gráfica para animação (Smith 1983), assim como mais recentemente na área de automação, para controle preditivo da movimentação de equipamentos, próteses, etc. (De Vito e Scattolini 2007, Faraway, Reed e Wang 2007), atuando geralmente na modelagem não linear de trajetórias tridimensionais.

Em se tratando de aplicações mais específicas no campo da engenharia química, entretanto, poucos trabalhos foram realizados. Na modelagem de processos não lineares, por exemplo, uma aplicação em redes neurais com o auxílio de *B-splines* foi proposta por Lightbody (1997). Um dos trabalhos mais próximos com a metodologia apresentada nesta dissertação, contudo, foi publicado por Sun e Wang (2005), onde foi proposto um controlador preditivo aplicado a sistemas dinâmicos de chamas, cuja distribuição de energia da chama é controlada a partir da velocidade de injeção de combustível no bocal. Para este caso, *B-splines* lineares de segunda ordem são empregadas para a representação do modelo, onde essas *splines* atuam como o modelo do processo, dando origem às matrizes de estado do processo.

Ao analisar esses trabalhos, é importante salientar a diferença no conceito por trás da SDM. O que se deseja para essa abordagem é a parametrização direta das entradas do sistema, que, associada a um modelo fenomenológico ou identificado do processo, permite o cálculo das saídas do mesmo, através da teoria de espaços de estados no domínio contínuo, combinando, conforme já mencionado, essa representação com a DM.

Uma vez que o tipo de representação é escolhida para a simulação do sistema, cabe ainda apresentar o que são *splines*, quais as suas peculiaridades e como elas se encaixam para a formulação da SDM. O próximo capítulo trata das mesmas, apresentando os diferentes tipos, assim como as suas vantagens e desvantagens para o problema proposto nesta dissertação.

Capítulo 3 – Splines

Ao introduzir o conceito da SDM, é necessário um conhecimento prévio da natureza das *splines*: o que são, como se originam, quais os seus fundamentos. Partindo dessa premissa, este capítulo tem por objetivo, essencialmente, apresentar os conceitos fundamentais relacionados a *splines*.

Splines consistem em ferramentas que, através das coordenadas de pontos no espaço, permitem a representação de todo um horizonte através de funções polinomiais de qualquer grau, sem propriedades extrapolativas. Tais funções podem ser globais, quando uma única função representa todo o espaço dos dados, ou podem ser *piecewise*, quando funções distintas representam intervalos distintos do sistema. Juntamente com essa classificação, ainda há a natureza da *spline* em questão, interpolante ou de ajuste. O primeiro tipo gera as funções polinomiais interpolando o conjunto de dados, onde a interpolação pode ser total ou parcial. As *splines* de ajuste, por outro lado, geram funções que somente aproximam os pontos, não os interpolando (Shikin e Plis 1995).

Em se tratando de simulação dinâmica de processos, o objetivo principal com as *splines* é a representação das entradas ao longo do tempo. Os nós, pontos no espaço que servem como base para a construção das *splines*, empregados para a geração das funções polinomiais, logo, devem fazer parte da própria entrada do sistema, o que resulta na necessidade de *splines* interpolantes, e não de ajuste.

Dentre as *splines* interpolantes, as diferenças principais residem no algoritmo empregado para a construção das curvas, que impacta na continuidade da derivada das funções *piecewise*, a interpolação de todos os pontos de controle e a tendência de oscilações na resposta (Foley 2006). Três grandes grupos se destacam: Bézier, B-splines e Hermite.

Em se tratando de simulação de processos, o que se deseja é a representação da entrada u , na ordenada, em função do tempo de simulação t , na abscissa, através de funções polinomiais cúbicas *piecewise*. A característica cúbica da curva proporciona um comportamento mais flexível, permitindo inflexões entre um ponto e outro, sem, entretanto, promover oscilação excessiva.

À exceção da abordagem de Bézier, *splines* interpolantes seguem uma mesma parametrização das variáveis supracitadas em função de uma variável auxiliar paramétrica j , conforme apresentado em (3.1) e (3.2).

$$f_t(j) = aj^3 + bj^2 + cj + d \quad (3.1)$$

$$f_u(j) = ej^3 + fj^2 + gj + h \quad (3.2)$$

Combinando ambas as equações, chega-se a uma estrutura matricial dada por:

$$[f_t \quad f_u] = [j^3 \quad j^2 \quad j \quad 1] \begin{bmatrix} a & e \\ b & f \\ c & g \\ d & h \end{bmatrix} \quad (3.3)$$

Onde, para cada abordagem empregada, a matriz de coeficientes a a h precisa ser desmembrada em uma matriz característica e uma matriz de coordenadas dos pontos, estabelecendo uma relação entre os dados existentes e os dados interpolados desejados (Pakhira 2010). As subseções seguintes explicitam cada grupo de *splines*.

3.1 Curvas de Bézier

A abordagem de Bézier vale-se do Algoritmo de Casteljau para a obtenção das *splines* (Farin e Hansford 2000). Tal algoritmo consiste em utilizar funções paramétricas de primeiro grau para gerar funções paramétricas de ordem superior, conforme apresentado esquematicamente nas figuras 3.1 e 3.2, valendo-se de uma variável paramétrica j .

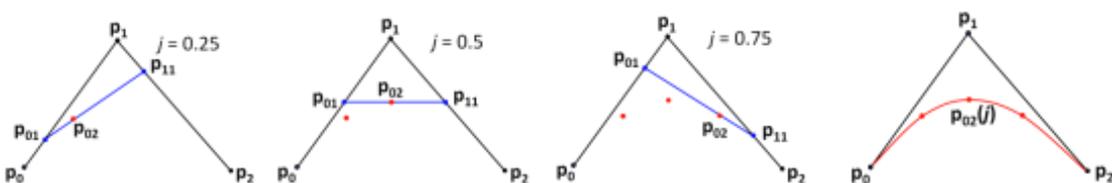


Figura 3.1: Representação esquemática da obtenção de uma função quadrática pelo algoritmo de Casteljau.

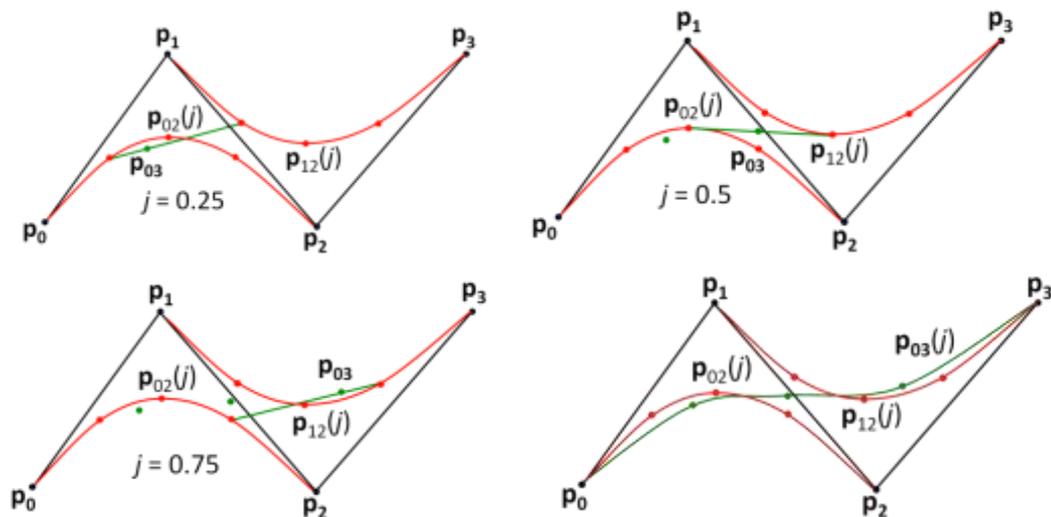


Figura 3.2: Representação esquemática da obtenção de uma função cúbica pelo algoritmo de Casteljau, partindo de funções quadráticas parametrizadas.

A base para o cálculo dessas curvas reside em um algoritmo geométrico, que se vale de funções paramétricas p_{ij} . Tais funções são apresentadas em maiores detalhes no apêndice I.

O que é importante destacar em relação às curvas de Bézier é a sua característica de interpolação parcial de dados. Como fica evidente na figura 3.2, a obtenção de uma função polinomial cúbica está condicionada à existência de quatro pontos no espaço, onde somente dois deles são interpolados de fato, sendo necessários dois pontos auxiliares para a geração da tendência do polinômio. Isso representa uma desvantagem evidente, visto que uma maior gama de dados faz-se necessária para a simulação de processos dinâmicos.

A abordagem por curva de Bézier apresenta ainda outras desvantagens, associadas com a suavidade das curvas formadas. As curvas *piecewise* geradas, valendo-se para tanto de quatro pontos consecutivos no espaço, não influenciam a tendência das curvas subsequentes geradas, sendo, em suma, independentes umas das outras. Não há garantia, logo, de continuidade da derivada das curvas geradas, o que afeta seriamente o desempenho e a confiabilidade da representação do sinal da entrada, o que, conseqüentemente, afeta a qualidade da simulação do modelo.

3.2 B-Splines

B-splines é similar à curva de Bézier, no que diz respeito aos números de pontos a serem escolhidos e o efeito que eles têm na ordem da função polinomial gerada. Quanto mais pontos, maior a ordem da *spline*. A diferença entre as metodologias está em dois pontos principais. A função interpolada cúbica é gerada somente para os pontos intermediários, onde os pontos extremos são auxiliares. Sabendo que a geração da função dá-se somente para os pontos intermediários, as funções subsequentes durante o intervalo de dados são geradas valendo-se ainda de quatro pontos, mas trasladando tal intervalo de um em um

ponto (Shikin e Plis 1995). O resultado de tal algoritmo é esquematizado na figura 3.3.

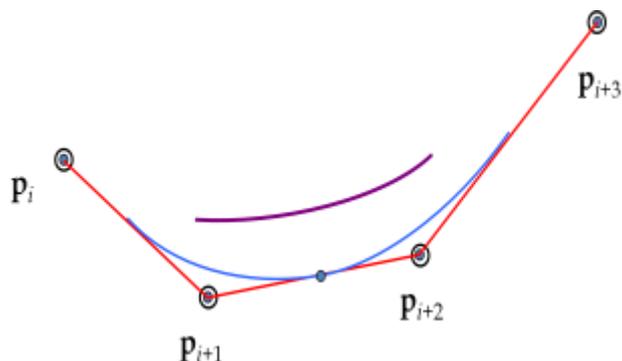


Figura 3.3: Representação esquemática de B-splines.

O algoritmo utilizado para o cálculo das funções leva à garantia de continuidade até a segunda derivada. Tal garantia, entretanto, promove oscilação excessiva das curvas, prejudicando, novamente, a captação da tendência real da curva interpolante desejada. Dessa forma, é uma *spline* inadequada para os propósitos deste trabalho.

3.3 Hermite Splines

Partindo da premissa que as curvas de Bézier incorrem em não suavidade das curvas e que as B-Splines geram funções excessivamente oscilatórias, as *Splines* de Hermite despontam como a solução para a interpolação de dados. Ao contrário das outras técnicas mencionadas, essas *splines* calculam uma função cúbica entre cada dois nós, valendo-se para tanto das tangentes nos mesmos. Assumindo que, logo, cada nó tenha uma tangente, ou um par de tangentes com a mesma direção como será explicado posteriormente, a continuidade da curva gerada é garantida até a primeira derivada e, conseqüentemente, promove-se a suavidade dentro do intervalo de ajuste (Biswal 2008).

As *Splines* de Hermite apresentam ainda outras características, como monotonicidade, o que implica em somente ascensão ou decréscimo do valor da função entre dois nós. Além disso, dependendo do tipo de Hermite *spline*, o ajuste de parâmetros é possível, conferindo diferentes propriedades às curvas geradas (Boor 2001).

A cada dois nós subseqüentes, estabelecem-se condições de contorno específicas, apresentadas esquematicamente na figura 3.4, que são empregadas na equação (3.3) para a construção da matriz característica de Hermite, que serve como base para a geração dos polinômios cúbicos.

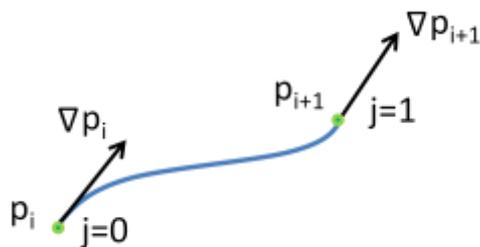


Figura 3.4: Representação esquemática das condições de contorno utilizadas

Onde i varia de 1 a z , sendo z o número de nós. As condições de contorno, para um intervalo de dados onde a variável auxiliar j é normalizada de 0 a 1, são dadas inicialmente pelas coordenadas dos pontos p_i e p_{i+1} no espaço, que são compostas por duas componentes, t e u , como fica claro na . i serve apenas como um indicador, para ilustrar que se trata de nós subsequentes. Supondo que possa se representar simultaneamente ambas as equações paramétricas apresentadas em (3.1) e (3.2) por uma função f que compreenda ambas as variáveis t e u , a fim de simplificar a representação, tem-se, conforme apresentado nas equações (3.4) a (3.5):

$$f(0) = p_i \quad (3.4)$$

$$f(1) = p_{i+1} \quad (3.5)$$

Sabendo que a derivada nos pontos também é necessária ao sistema como condição de contorno, derivam-se as expressões (3.1) e (3.2), conforme apresentado em (3.6) e (3.7):

$$f'_t(j) = 3aj^2 + 2bj + c \quad (3.6)$$

$$f'_u(j) = 3ej^2 + 2fj + g \quad (3.7)$$

Analogamente, valendo-se de uma representação simplificada f' , tem-se que as condições de contorno são tais que, de acordo com (3.8) e (3.9):

$$f'(0) = \frac{dp_i}{dj} \quad (3.8)$$

$$f'(1) = \frac{dp_{i+1}}{dj} \quad (3.9)$$

Ao combinar cada uma das condições com suas respectivas equações, a matriz dos coeficientes a a h pode ser determinada, de acordo com (3.10):

$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix} \begin{bmatrix} a & e \\ b & f \\ c & g \\ d & h \end{bmatrix} = \begin{bmatrix} t_i & u_i \\ t_{i+1} & u_{i+1} \\ \frac{dt_i}{dj} & \frac{du_i}{dj} \\ \frac{dt_{i+1}}{dj} & \frac{du_{i+1}}{dj} \end{bmatrix} \quad (3.10)$$

Isolando a matriz de coeficientes e substituindo em (3.3), tem-se, conforme se segue em (3.11):

$$[f_t \quad f_u] = [j^3 \quad j^2 \quad j \quad 1] \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} t_i & u_i \\ t_{i+1} & u_{i+1} \\ \frac{dt_i}{dj} & \frac{du_i}{dj} \\ \frac{dt_{i+1}}{dj} & \frac{du_{i+1}}{dj} \end{bmatrix} \quad (3.11)$$

Onde a matriz numérica é chamada de matriz característica de Hermite e a matriz referente aos dados e suas respectivas derivadas no espaço é chamada de matriz geométrica de Hermite (Iovine 1995).

Como foi mencionado no início deste capítulo, o objetivo da utilização das *splines* é representar as entradas do sistema em função do tempo, valendo-se de apenas alguns pontos. Partindo de (3.11), pode-se chegar a uma equação só para u , tomando somente a coluna da matriz geométrica referente a essa variável, como pode ser visto em (3.12):

$$u_{interp} = [j^3 \quad j^2 \quad j \quad 1] \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_i \\ u_{i+1} \\ \frac{du_i}{dj} \\ \frac{du_{i+1}}{dj} \end{bmatrix} \quad (3.12)$$

Como tal equação depende somente de j , é necessário estabelecer uma relação entre t e a mesma. Da literatura (Späth 1995) sabe-se que a relação entre essas variáveis é tal que, como se segue em (3.13):

$$j = \frac{t_k - t_i}{t_{i+1} - t_i} \quad (3.13)$$

Onde t_k representa o instante no qual se deseja saber uma determinada entrada. Aliando essa equação à regra da cadeia, para que todos os termos, incluindo as derivadas, sejam dependentes do tempo, chega-se em (3.14), que representa a forma final dos polinômios cúbicos interpolantes *piecewise* de Hermite, onde j foi mantido no vetor que confere a característica cúbica à curva, meramente para simplificar a representação:

$$u_k = [j^3 \quad j^2 \quad j \quad 1] \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_i \\ u_{i+1} \\ (t_{i+1} - t_i)m_i \\ (t_{i+1} - t_i)m_{i+1} \end{bmatrix} \quad (3.14)$$

Onde u_k é a entrada equivalente ao instante t_k e m_i representa a derivada de u em relação a t no nó i . Haverá tantas funções *piecewise* quanto mais pontos houver, seguindo uma regra geral de $z-1$ equações para cada z nós existentes.

Aparentemente, há um número maior de parâmetros quando comparado com as outras abordagens citadas nas subseções anteriores. As curvas de Bézier e as *B-splines* dependem exclusivamente da posição dos pontos, sem haver necessidade de tangente, o que indicaria, em um primeiro momento, um maior número de parâmetros necessários para a representação em *splines* de Hermite. O fato é que as tangentes são calculadas, em função dos próprios nós, ou seja, os parâmetros são, na realidade, somente os nós existentes. Para tanto, vale-se do conceito de diferenças finitas, expresso em (3.15) e na figura 3.5.

$$\Delta_i = \frac{u_{i+1} - u_i}{t_{i+1} - t_i} \quad (3.15)$$

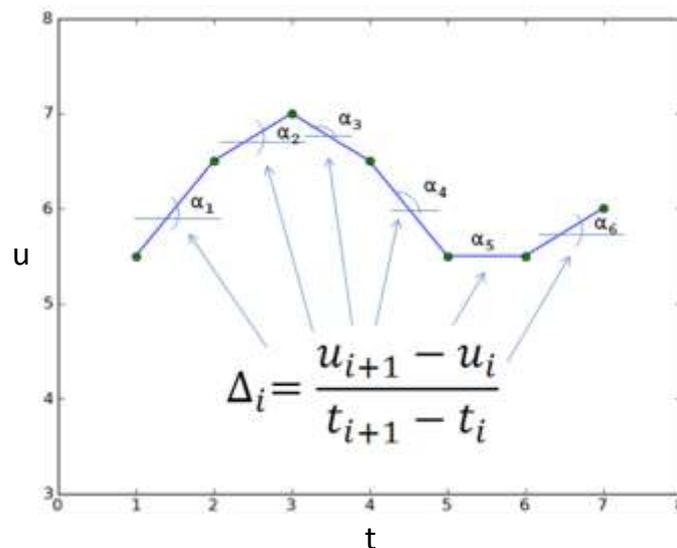


Figura 3.5: Representação esquemática do cálculo dos deltas em cada intervalo.

A diferenciação entre os tipos de Hermite *Splines* reside justamente nas diferentes maneiras de se calcular tal tangente. Dois tipos principais são considerados para esta dissertação, a *Kochanek-Bartels Hermite Spline*, também chamada de TCB Hermite, e a *Piecewise Cubic Hermite Interpolator* (PCHIP). As próximas subseções vão tratar mais especificamente de cada uma delas, apresentando as suas peculiaridades, vantagens e desvantagens.

3.3.1 Piecewise Cubic Hermite Interpolator (PCHIP)

A PCHIP trabalha com a ponderação dos deltas, a fim de obter maior estabilidade e suavização. Tal ponderação impede a existência de vetores tangentes muito grandes, ou com súbita mudança de direção. Além disso, se um nó apresenta deltas adjacentes com sinais contrários, a tangente resultante é dita como sendo zero. O mesmo ocorre se ao menos um dos deltas é igual a zero, indicativo este de dois nós subsequentes com o mesmo valor de u .

É uma *spline* que, devido às suas características, é capaz de representar bem praticamente qualquer curva, dado um número mínimo de nós. Há algumas desvantagens, entretanto, decorrentes da sua implementação. Não há espaço para ajuste de parâmetros, sendo uma formulação fechada. Além disso, como no seu algoritmo, há condicionais a serem avaliadas, dificulta-se a combinação dessa

metodologia com a simulação dos processos, o que ficará claro no próximo capítulo, quando a metodologia SDM for de fato apresentada.

3.3.2 Kochanek-Bartels Hermite Spline (TCB)

A TCB, por outro lado, trabalha com a média entre dois “deltas” adjacentes a um nó, combinado com certos parâmetros, chamados de tensão (τ), continuidade (γ) e bias (ω), que conferem, independentemente, propriedades específicas para a tangente gerada. Esses parâmetros podem variar de -1 a 1, sendo ajustados nó a nó. O valor padrão para todos esses parâmetros é zero. Ainda, decorrente da própria metodologia, a cada nó é conferida não apenas uma, mas duas tangentes, uma de entrada no nó e outra de saída do nó. Cada tangente pode ser calculada de acordo com (3.16) e (3.17):

$$m_i^{in} = \frac{(1-\tau_i)(1+\gamma_i)(1-\omega_i)}{2}(\Delta_{i+1} - \Delta_i) + \frac{(1-\tau_i)(1-\gamma_i)(1+\omega_i)}{2}(\Delta_i - \Delta_{i-1}) \quad (3.16)$$

$$m_i^{out} = \frac{(1-\tau_i)(1-\gamma_i)(1-\omega_i)}{2}(\Delta_{i+1} - \Delta_i) + \frac{(1-\tau_i)(1+\gamma_i)(1+\omega_i)}{2}(\Delta_i - \Delta_{i-1}) \quad (3.17)$$

A Tensão afeta o comprimento do vetor tangente obtido, sendo responsável pela oscilação permitida. Um vetor tangente menor acarreta em maior liberdade de oscilação e vice-versa. A Continuidade afeta, como o próprio nome indica, a continuidade da derivada da função, ao tornar o valor da tangente de saída e de entrada em um nó diferentes. Assim, duas *splines* que se encontram em um nó cujo parâmetro de continuidade é diferente do seu valor padrão, não apresentarão uma transição suave ao se encontrarem. O Bias pondera um delta adjacente em detrimento do outro, valorizando mais a tendência à esquerda ou à direita da curva. A figura 3.6 ilustra os diferentes cenários para os diferentes parâmetros e seus respectivos efeitos na geração das *splines*.

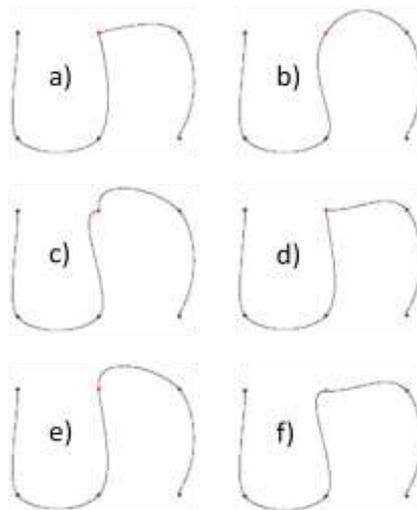


Figura 3.6: Efeito dos parâmetros da TCB na geração das *splines* para o nó vermelho em destaque, para tensão igual a: a) 1 e b) -1; continuidade igual a: c) 1 e d) -1; e bias igual a: e) 1 e f) -1.

Em função da liberdade de ajuste existente na TCB, o que garante resultados mais próximos às curvas de referência do processo e à inexistência de características condicionais associadas a sua implementação, a TCB é considerada a curva mais adequada para a parametrização das entradas de um processo dinâmico.

As figuras 3.7 e 3.8 apresentam uma comparação entre a PCHIP e a TCB para curvas análogas a respostas degrau aplicadas em sistemas de primeira e segunda ordem, respectivamente, onde a entrada U é uma variável arbitrária qualquer. Os sistemas em questão são representados pelas equações expressas em (3.18) e (3.19):

$$G_1(s) = \frac{1}{s+0.5} \quad (3.18)$$

$$G_2(s) = \frac{1}{s^2+0.5s+0.25} \quad (3.19)$$

Como pode ser visto há duas curvas distintas em cada figura associadas à TCB, uma partindo da condição padrão, onde τ , γ e ω são iguais a zero, e outra com os parâmetros manipulados de modo a melhorar a aderência da curva ao modelo em questão. Tais parâmetros podem ser vistos na tabela 3.1. Em se tratando da PCHIP, pode-se perceber que a mesma apresenta dificuldades em capturar a dinâmica inicial de ambas as funções, apresentando erros tão maiores quanto mais complexas são as mesmas. É verdade que em uma primeira análise a TCB não apresenta resultados melhores do que os apresentados pela PCHIP, entretanto isso se deve à curva estar na condição padrão, sem ajuste de parâmetros algum. Ao manipular alguns desses parâmetros, a curva ajusta-se muito bem, havendo pouca discrepância em relação à curva real de fato.

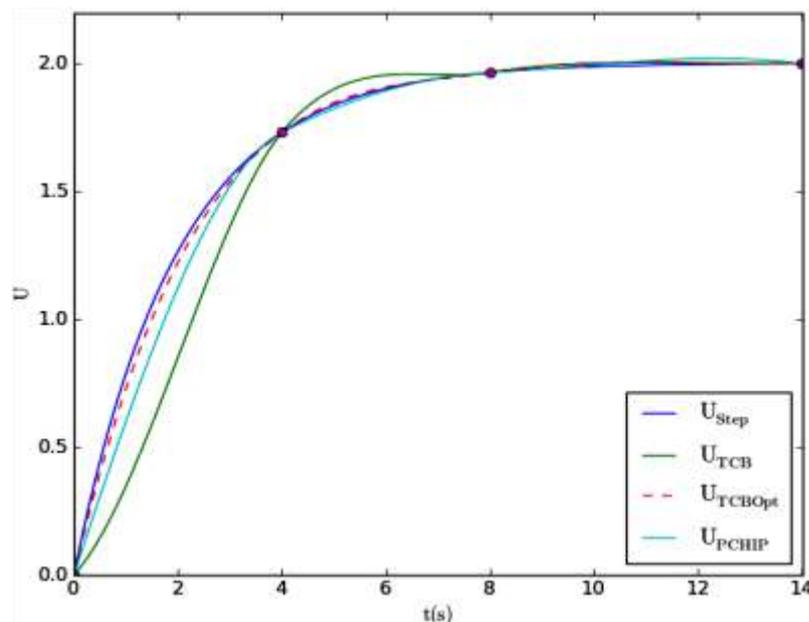


Figura 3.7: Curva correspondente a uma resposta degrau de um sistema de primeira ordem comparando o desempenho da PCHIP com a Hermite TCB, padrão e otimizada, conforme os dados apresentados na tabela 3.1.

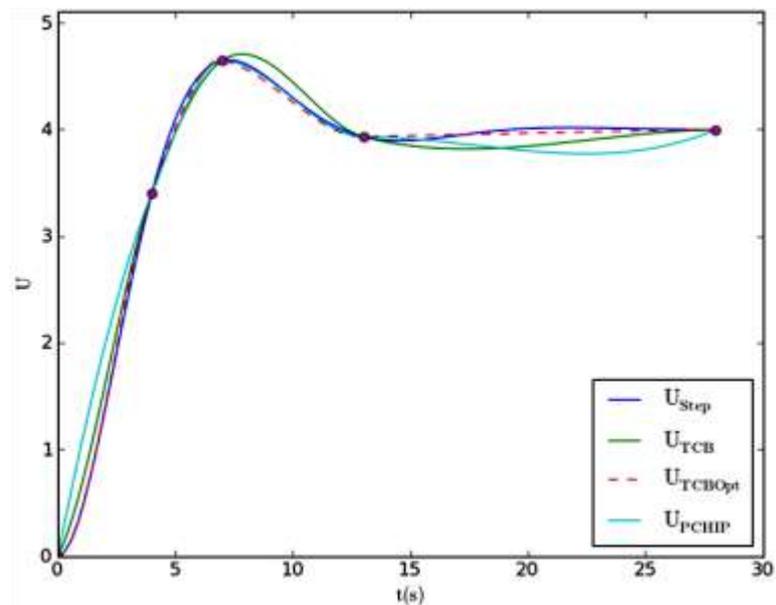


Figura 3.8: Curva correspondente a uma resposta degrau de um sistema de segunda ordem comparando o desempenho da PCHIP com a Hermite TCB, padrão e otimizada, conforme os dados apresentados na tabela 3.1.

Tabela 3.1: Parâmetros da TCB otimizada para as funções, uma análoga a um sistema de primeira ordem e outra, a um de segunda ordem.

Sistema	τ	γ	ω
1ª Ordem (G_1)	[-1;0;0;0]	[0;0;0;0]	[-1;-0,5;0;0]
2ª Ordem (G_2)	[0;0;0;0;0]	[0;0;0;0;0]	[1;0;-0.7;-1;0]

As questões referentes ao posicionamento e ao número de nós serão tratadas posteriormente, na seção 5.5.1, quando da apresentação do MPC baseado na metodologia SDM.

Uma vez desenvolvido o assunto referente a *splines*, é possível, juntamente com o conhecimento apresentado no capítulo 2, introduzir a metodologia completa referente à SDM, tema do próximo capítulo.

Capítulo 4 – Spline Dynamic Matrix - SDM

Uma vez que todas as equações necessárias para a simulação dinâmica de processos lineares foram estabelecidas, juntamente com o tipo de *spline* mais adequada para a parametrização das entradas do sistema, o conceito da SDM pode ser então apresentado.

Uma das maiores motivações por trás da SDM é a redução do número de variáveis necessárias à simulação dinâmica do processo, que consiste na parametrização das entradas do sistema através de *splines*. Parte-se, logo, de um conjunto de nós para cada entrada, representando pontos no espaço para determinados instantes de tempo, recriando-se o perfil de entrada ao longo do tempo, onde, para cada entrenó, há uma função *piecewise* cúbica que representa esse intervalo.

Naturalmente surge o questionamento sobre quais seriam as melhores coordenadas para estes nós serem inseridos. Como uma boa métrica, regiões onde a informação dinâmica é relevante devem ser escolhidas. Em termos de simulação, isso acarreta em nós sendo colocados em instantes que possibilitem a melhor aproximação das entradas reais do sistema ao longo de um dado horizonte de simulação. Alguns métodos foram desenvolvidos para a extração desses nós com maior relevância dinâmica por Silveira (2012), entretanto, para esta dissertação, os pontos foram escolhidos visando simplesmente uma boa aproximação, não havendo uma métrica em particular, focando-se nos resultados decorrente do desenvolvimento da metodologia SDM.

Inspirado na matriz dinâmica (DM) e na representação discreta em espaços de estados (DEE) discutidas no capítulo 2, parte-se do pressuposto que o espaço é discretizado em P instantes de tempo que podem, ou não, ser igualmente espaçados. A partir dessa discretização busca-se uma representação matricial na forma de uma matriz dinâmica que seja dependente somente da posição dos nós no espaço, mas que possa representar cada ponto ao longo do horizonte de simulação. É importante salientar aqui que os nós e os pontos discretizados não

são equivalentes. Os nós representam um conjunto muito menor de informação, colocados em posições específicas, sendo a discretização realizada em um número muito maior. Uma representação dessa ideia pode ser vista na figura 4.1, para um conjunto de u e t quaisquer.

As subseções apresentadas a seguir desmembram a metodologia em duas partes: uma referente à representação matricial das entradas do sistema e outra referente ao cálculo da saída e composição da matriz SDM propriamente dita.

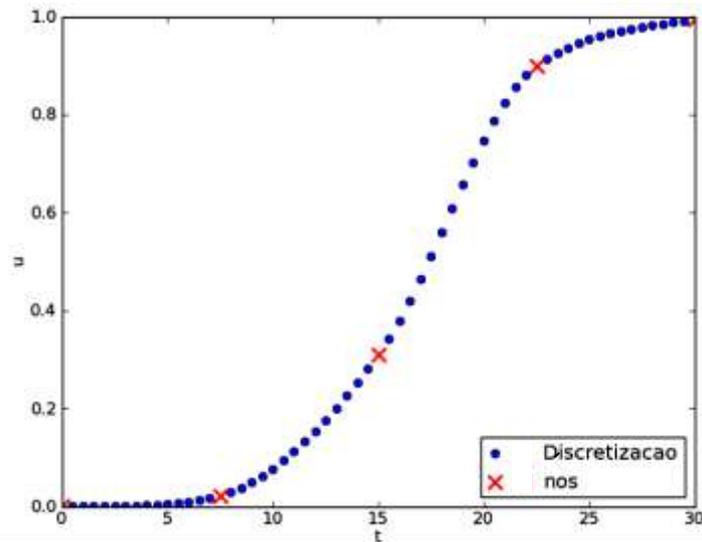


Figura 4.1: Representação esquemática da distinção entre nós e pontos discretizados ao longo do horizonte de simulação para uma entrada sigmoideal.

4.1 Representação Matricial das Entradas

Cada um dos instantes do horizonte discretizado apresenta uma ou mais entradas, correspondentes ao valor da função *piecewise* cúbica calculada para aquele entrenó, aplicado a cada instante. Uma vez estabelecidas essas funções, o cálculo da entrada ao longo de todo o horizonte de simulação é dependente, logo, somente do valor da entrada nos nós.

Da equação (3.14), obtém-se o cálculo de $u(j)$ para um único intervalo. A partir dessa equação, juntamente com a substituição de j por (3.13), multiplicam-se os termos para obter-se uma expressão como a apresentada em (4.1):

$$u_i(t) = T(t) \text{Coef}_i \begin{bmatrix} u_i \\ u_{i+1} \\ m_i^{\text{out}} \\ m_{i+1}^{\text{in}} \end{bmatrix} \quad (4.1)$$

Com i variando de 1 a $z-1$. $T(t)$ é um vetor linha que expressa a natureza cúbica da entrada, como pode ser visto em (4.2):

$$T(t) = [t^3 \quad t^2 \quad t \quad 1] \quad (4.2)$$

É importante salientar que, devido à natureza da *spline* escolhida, há dois tipos de derivadas m por intervalo, uma de saída e outra de entrada. $Coef$ é uma matriz quadrada 4×4 que expressa o efeito que cada ponto no espaço, assim como suas respectivas derivadas, tem em cada termo do polinômio cúbico *piecewise* gerado. A sua estrutura pode ser vista em (4.3):

$$Coef_i = \begin{bmatrix} \left(\frac{2}{\delta_i^3}\right) & \left(\frac{1}{\delta_i^2}\right) & -\left(\frac{2}{\delta_i^3}\right) & \left(\frac{1}{\delta_i^2}\right) \\ \left(-\frac{6t_i}{\delta_i^3} - \frac{3}{\delta_i^2}\right) & \left(-\frac{3t_i}{\delta_i^2} - \frac{2}{\delta_i}\right) & \left(\frac{6t_i}{\delta_i^3} + \frac{3}{\delta_i^2}\right) & \left(-\frac{3t_i}{\delta_i^2} - \frac{2}{\delta_i}\right) \\ \left(\frac{6t_i^2}{\delta_i^3} + \frac{6t_i}{\delta_i^2}\right) & \left(\frac{3t_i^2}{\delta_i^2} + \frac{4t_i}{\delta_i} + 1\right) & \left(-\frac{6t_i^2}{\delta_i^3} - \frac{6t_i}{\delta_i^2}\right) & \left(\frac{3t_i^2}{\delta_i^2} + \frac{2t_i}{\delta_i}\right) \\ \left(1 - \frac{2t_i^3}{\delta_i^3} - \frac{3t_i^2}{\delta_i^2}\right) & \left(-t_i - \frac{t_i^3}{\delta_i^2} - \frac{t_i^2}{\delta_i}\right) & \left(\frac{2t_i^3}{\delta_i^3} + \frac{3t_i^2}{\delta_i^2}\right) & \left(\frac{t_i^3}{\delta_i^2} - \frac{t_i^2}{\delta_i}\right) \end{bmatrix} \quad (4.3)$$

Onde δ_i é expresso por (4.4):

$$\delta_i = t_{i+1} - t_i \quad (4.4)$$

O que se deseja, entretanto, é uma expressão para a entrada ao longo do tempo que seja dependente apenas do valor da mesma em cada nó, sabendo que os instantes de tempo na discretização são conhecidos. Em se tratando da metodologia de *splines*, sabe-se que as derivadas nos pontos são dependentes da própria posição dos nós no espaço. Assim, o vetor coluna apresentado em (4.1) pode ser desmembrado em uma matriz, que contempla o cálculo das derivadas, e um vetor coluna, composto apenas do valor da entrada em cada um dos nós. A essa matriz dá-se o nome de $TCBMat$, fazendo alusão à *spline* escolhida, cuja estrutura é apresentada em maiores detalhes no apêndice II. $TCBMat$ atua integrando o cálculo da derivada na representação matricial, garantindo assim que a entrada em cada nó possa ser representada por um vetor coluna H independente, como mostrado em (4.5):

$$H = \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_z \end{bmatrix} \quad (4.5)$$

Onde h representa o valor das entradas nos nós. É importante salientar que apenas o valor das entradas, e não a sua distribuição no horizonte, representa a variável do problema. Conforme citado anteriormente, a posição dos nós é escolhida de modo arbitrário, visando uma boa representação do perfil de entradas em questão.

Tanto $Coef$ quanto $TCBMat$ são dependentes apenas dos instantes de tempo onde os nós são colocados, à exceção dos parâmetros da tensão, continuidade e bias, que afetam a segunda matriz em questão. Para uma única entrada, então, a representação matricial é tal como se segue em (4.6):

$$u_i(t) = T(t) \text{Coef}_i \text{TCBMat}_i H, \quad i = 1..(z - 1) \quad (4.6)$$

Uma vez estabelecida a expressão para a entrada, pode-se proceder à concepção da matriz SDM propriamente dita, que resultará na simulação do sistema.

4.2 Simulação das Saídas do Sistema / Estrutura SDM

Partindo da equação apresentada em (2.7), que se refere à representação em espaços de estados de um sistema no domínio contínuo, é possível a simulação dos estados de um sistema e, conseqüentemente, da saída do sistema para todos os intervalos existentes ao longo do horizonte de simulação. Esse conceito é expresso pelas equações (4.7) e (4.8):

$$X(t) \begin{bmatrix} 1 \\ \vdots \\ z-1 \end{bmatrix} = e^{A(t-t_0)} x_0 + \int_{t_i}^t e^{A(t-\tau)} BU(\tau) \begin{bmatrix} 1 \\ \vdots \\ z-1 \end{bmatrix} d\tau \quad (4.7)$$

$$Y(t) \begin{bmatrix} 1 \\ \vdots \\ z-1 \end{bmatrix} = C \cdot X(t) \begin{bmatrix} 1 \\ \vdots \\ z-1 \end{bmatrix} \quad (4.8)$$

Para fins didáticos de representação, parte-se do pressuposto que os nós são igualmente espaçados ao longo do horizonte de simulação, tendo cada entrenó r pontos. A partir dessa simplificação, fica mais evidente a transição de cada função polinomial $u_i(t)$ em cada intervalo. Ainda, a fim de explicitar posteriormente a simulação de um sistema sujeito à variação de mais de uma entrada simultaneamente, é apresentado inicialmente apenas o efeito de uma entrada na saída do sistema. Vale salientar que o termo $Y(t)$ pode se referir a mais de uma saída, podendo, logo, ter diversos valores em um único instante.

Nota-se que a representação apresentada em (4.7) é composta de dois termos. O primeiro leva em consideração a condição inicial do sistema e a sua tendência, que corresponde a uma exponencial decrescente que ruma a valores nulos com o passar do tempo, para o caso de se estar lidando com sistemas estáveis, sendo a mesma independente do conceito de intervalos expresso pelas *splines*, *i.e.*, uma resposta livre do sistema. Essa parcela é representada por X_0 , apresentado em (4.9).

$$X_0 \begin{bmatrix} k \\ \vdots \\ k+p \end{bmatrix} = \begin{bmatrix} e^{A(t_k-t_1)} \\ e^{A(t_{k+1}-t_1)} \\ \vdots \\ e^{A(t_{k+p}-t_1)} \end{bmatrix} x_0 \quad (4.9)$$

Em relação a esse termo, é importante salientar que, em virtude da representação em espaços de estado ser linear, as matrizes de estado podem ser obtidas a partir da linearização do sistema em um determinado instante de tempo. Assim, a parcela referente à condição inicial ruma a esse valor estacionário e não a zero. Isso é explicado sabendo que a simulação dá-se em deltas, tomando como base o ponto de linearização.

Esse comportamento pode ser interpretado fisicamente como a tendência do sistema a retornar ao estado estacionário que é coerente com as entradas que lhe foram associadas. Supondo que houvesse, por exemplo, um tanque qualquer com uma única saída e sem entradas, é evidente que a tendência do tanque seria esvaziar, visto que não há entrada nenhuma de fluido no sistema. O seu estado estacionário é atingido quando todo o fluido deixa o sistema, sendo independente da condição inicial estabelecida. Em termos práticos, esse termo é usualmente nulo.

O segundo termo contempla o efeito que a entrada ao longo do tempo tem na saída, valendo-se para tanto de uma integral. Trabalhando com o conceito de discretização para a simulação do problema, a base da SDM consiste em aplicar tal integral para cada instante, lembrando que o vetor coluna H não é afetado pela integração ao longo do tempo, nem pelos intervalos, podendo ser retirado da integral. Esse termo é representado por X_U , conforme pode ser visto em (4.10), onde vale frisar que há distintas equações $u_i(t)$ em cada entrenó:

$$X_U \begin{bmatrix} k \\ k+P \end{bmatrix} = \begin{bmatrix} \int_{t_1}^{t_k} e^{A(t-\tau)} B_1 U_1(t) d\tau \\ \int_{t_1}^{t_{k+1}} e^{A(t-\tau)} B_1 U_1(t) d\tau \\ \vdots \\ \int_{t_1}^{t_{k+r-1}} e^{A(t-\tau)} B_1 U_1(t) d\tau \\ \int_{t_2}^{t_{k+r}} e^{A(t-\tau)} B_1 U_2(t) d\tau \\ \int_{t_2}^{t_{k+r+1}} e^{A(t-\tau)} B_1 U_2(t) d\tau \\ \vdots \\ \int_{t_2}^{t_{k+2r-1}} e^{A(t-\tau)} B_1 U_2(t) d\tau \\ \vdots \\ \int_{t_{z-1}}^{t_{k+(z-1)r}} e^{A(t-\tau)} B_1 U_{z-1}(t) d\tau \\ \int_{t_{z-1}}^{t_{k+(z-1)r+1}} e^{A(t-\tau)} B_1 U_{z-1}(t) d\tau \\ \vdots \\ \int_{t_{z-1}}^{t_{k+P}} e^{A(t-\tau)} B_1 U_{z-1}(t) d\tau \end{bmatrix} \quad (4.10)$$

Onde B_1 é a primeira coluna da matrix B , advinda da representação em espaços de estado, indicando que o efeito é de apenas uma das entradas no sistema. Perceba que há a avaliação da resposta do sistema por intervalos, entretanto ela é realizada a partir de uma concepção de sistema contínuo, diferentemente das representações RD ou DEE.

Complementarmente, é importante levar em consideração também a tendência inercial do sistema intervalo a intervalo, análoga à expressão representada por X_0 , expressa como X_{U0} . Em se tratando da modelagem matemática, entretanto, a analogia não é tão evidente, sendo de difícil concepção em um primeiro momento. Seria equivalente a uma resposta inercial acumulada do sistema.

Cada intervalo apresenta um valor inicial que é utilizado para o cálculo da tendência inercial ao longo do mesmo. Esse valor depende do efeito que a entrada $u_{i-1}(t)$ no intervalo anterior $i-1$ tem sobre o sistema naquele novo instante, o que não é contemplado em X_U , visto que o cálculo de cada equação $u_i(t)$ em (4.10) só afeta o próprio intervalo, não se sobrepondo aos demais. Perceba que a cada novo intervalo há um efeito acumulativo, onde as tendências passadas se somam aos novos valores iniciais de cada intervalo, acarretando em uma maior complexidade da representação. Cada um dos valores iniciais em cada intervalo é expresso em (4.11):

$$Z = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 \\ e^{A(t_{k+r}-t_2)} & 1 & \ddots & \vdots & 0 \\ \prod_2^3 e^{A(t_{k+ri}-t_i)} & e^{A(t_{k+r}-t_2)} & \ddots & 0 & \vdots \\ \vdots & \vdots & \ddots & 1 & 0 \\ \prod_2^{z-1} e^{A(t_{k+ri}-t_i)} & \prod_2^{z-2} e^{A(t_{k+ri}-t_i)} & \dots & e^{A(t_{k+r}-t_2)} & 1 \end{bmatrix} \begin{bmatrix} \int_{t_1}^{t_{k+r}} e^{A(t-\tau)} B_1 U_1(t) d\tau \\ \int_{t_2}^{t_{k+2r}} e^{A(t-\tau)} B_1 U_2(t) d\tau \\ \vdots \\ \int_{t_{z-3}}^{t_{k+(z-3)r}} e^{A(t-\tau)} B_1 U_{z-3}(t) d\tau \\ \int_{t_{z-2}}^{t_{k+(z-2)r}} e^{A(t-\tau)} B_1 U_{z-2}(t) d\tau \end{bmatrix} \quad (4.11)$$

Tendo os valores iniciais em cada intervalo, a expressão para X_{U0} toma a forma apresentada em (4.12).

$$X_{U0} \begin{bmatrix} k \\ k+P \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} e^{A(t_{k+1}-t_1)} \\ e^{A(t_{k+2}-t_1)} \\ \vdots \\ e^{A(t_{k+r-1}-t_1)} \end{bmatrix} \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \\ \begin{bmatrix} e^{A(t_{k+r}-t_2)} \\ e^{A(t_{k+r+1}-t_2)} \\ \vdots \\ e^{A(t_{k+2r-1}-t_2)} \end{bmatrix} Z_1 \\ \begin{bmatrix} e^{A(t_{k+2r}-t_2)} \\ e^{A(t_{k+2r+1}-t_2)} \\ \vdots \\ e^{A(t_{k+3r-1}-t_2)} \end{bmatrix} Z_2 \\ \vdots \\ \begin{bmatrix} e^{A(t_{k+(z-1)r}-t_{z-1})} \\ e^{A(t_{k+(z-1)r+1}-t_{z-1})} \\ \vdots \\ e^{A(t_{k+P}-t_{z-1})} \end{bmatrix} Z_{z-2} \end{bmatrix} \quad (4.12)$$

Onde Z_i é advindo de Z como matrizes compostas de s linhas (Número de Estados) e número de colunas igual à segunda matriz apresentada em (4.11). A soma de X_U e X_{U0} origina a matriz SDM. Assim, a expressão final é tal como se segue em (4.13):

$$Y \begin{bmatrix} k \\ k+P \end{bmatrix} = C(X_0 + (X_U + X_{U0}) H) = C(X_0 + SDM H) \quad (4.13)$$

O efeito de cada parcela pode ser constatado na figura 4.2, onde a resposta a um degrau unitário na entrada é mostrada. O processo em questão é uma planta

laboratorial de três tanques esféricos com uma entrada de líquido F_1 no primeiro tanque, onde L é o nível do terceiro tanque. O sistema é explicado em maiores detalhes na seção 6.2.1.

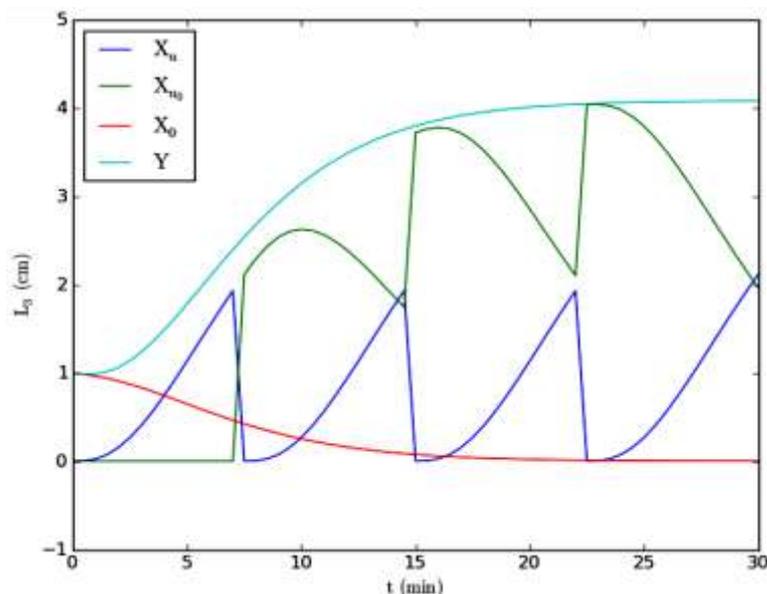


Figura 4.2: Composição das parcelas X_0 , X_U e X_{U0} na simulação de um degrau unitário na vazão de entrada de uma planta laboratorial de três tanques e seu efeito no nível do terceiro tanque.

A expansão para um sistema multivariável é direta. Uma SDM é construída para cada entrada e elas são posteriormente concatenadas lado a lado em uma SDM composta, conforme se segue em (4.14):

$$SDM = [SDM_1 \quad SDM_2 \quad \dots \quad SDM_n] \quad (4.14)$$

Onde n é o número de entradas do sistema. O vetor coluna H passa a ser representado por (4.15):

$$H = \begin{bmatrix} h_{11} \\ h_{12} \\ \vdots \\ h_{1z_1} \\ \vdots \\ h_{n1} \\ h_{n2} \\ \vdots \\ h_{nz_n} \end{bmatrix} \quad (4.15)$$

Onde z_i representa o número de nós para cada entrada. Apesar dessas modificações, a simulação do sistema continua a ser representada pela equação apresentada em (4.13).

A partir dessa representação é evidente que o número de linhas das SDMs devem ser os mesmos, ou seja, a quantidade de pontos discretizados deve ser a

mesma para todas as entradas, podendo ser variados, entretanto, o número de nós e a sua posição ao longo do horizonte.

4.3 Sistemas Variantes no Tempo

Apenas com o objetivo de motivar trabalhos futuros, a SDM poderia ser aplicada para sistemas variantes no tempo, justamente pela natureza analítica do cálculo das saídas. Partindo do pressuposto que as matrizes A , B , C e D advindas da representação em espaços de estado possam ser representadas analiticamente como funções dependentes do tempo, a construção da matriz poderia ser realizada. Esquemáticamente, as equações (4.7) e (4.8) apresentadas na seção 4.2 tomariam a seguinte forma, conforme (4.16) e (4.17):

$$X(t) \begin{bmatrix} 1 \\ \vdots \\ n-1 \end{bmatrix} = e^{A(t)(t-t_0)} x_0 + \int_{t_i}^t e^{A(\tau)(t-\tau)} B(\tau) U(\tau) \begin{bmatrix} 1 \\ \vdots \\ n-1 \end{bmatrix} d\tau \quad (4.16)$$

$$Y(t) \begin{bmatrix} 1 \\ \vdots \\ n-1 \end{bmatrix} = C(t) \cdot X(t) \begin{bmatrix} 1 \\ \vdots \\ n-1 \end{bmatrix} \quad (4.17)$$

Claramente haveria a necessidade da resolução de integrais mais complexas, associada à variação no tempo das matrizes de espaços de estado, acarretando no aumento de carga computacional. As vantagens associadas à metodologia, entretanto, poderiam superar tal desvantagem.

Capítulo 5 – Controlador Preditivo Baseado em Modelo (MPC)

De posse de uma ferramenta que possibilite a simulação de processos dinâmicos através de um modelo linear, tal como visto na SDM, aplicações em controladores preditivos baseados em modelo podem ser vistas como uma aplicação natural da metodologia, visto que a modelagem tem um papel fundamental na simulação do processo e na concepção do problema de otimização presentes no controlador. Este capítulo apresenta uma breve revisão quanto aos conceitos gerais e origem de MPCs, assim como quanto às suas principais formulações, a fim de motivar a apresentação de uma nova abordagem valendo-se da SDM apresentada nesta dissertação.

5.1 Conceitos Gerais

MPCs tiveram suas origens firmadas há aproximadamente cinco décadas, quando foi realizada uma conexão entre o problema ótimo de tempo mínimo e a programação linear (Zadeh e Whalen 1962), assim como se sugeriu pela primeira vez que uma aproximação por horizonte móvel poderia ser utilizado para este problema de controle (Propoi 1963). Estratégias essas que se referem a uma classe de algoritmos computacionais que, utilizando um modelo explícito do processo, prediz as respostas futuras da planta.

O MPC desponta nesse cenário como uma ferramenta altamente empregada. Há basicamente três razões principais para se utilizar MPCs. Primeiro, eles levam em consideração restrições nas entradas e saídas do sistema, garantindo qualidade, eficiência e segurança na produção. Segundo, com a utilização de um modelo do processo, é possível gerar saídas do controlador que levem em consideração interações internas do sistema, podendo operar com processos de difícil dinâmica e cujo número de variáveis manipuladas e controladas não seja igual. Terceiro, salientando uma vantagem do ponto de vista prático da utilização

de MPCs, o princípio de operação dos algoritmos é compreensível e razoavelmente fácil de explicar a um grupo de trabalho de uma instalação industrial, o que é um ponto chave ao introduzir novas técnicas nesse tipo de ambiente (Tatjewski 2010).

Levando em consideração tais vantagens, o MPC pode ser facilmente empregado em sistemas multivariáveis, trabalhando com limitações no atuador e nas variáveis controladas e manipuladas, através de restrições explícitas. Ainda, permite a utilização de *targets*, ou seja, valores desejáveis para as variáveis manipuladas, assim como controle de variáveis tanto por *set point*, como por faixas de operação (Camacho e Bordons 2004). É um controlador que apresenta versatilidade, eficiência, sendo que suas vantagens corroboram a sua tão ampla utilização na indústria de processos.

Um MPC apresenta genericamente uma estrutura básica como é mostrado na figura 5.1, consistindo basicamente de um modelo, que representa uma abstração matemática do processo que se deseja controlar, e um algoritmo de otimização, cujo objetivo é determinar a ação de controle a ser aplicada no sistema para cada instante de tempo, conforme a amostragem.



Figura 5.1: Representação Esquemática de um MPC.

FONTE: (Paim 2009)

O procedimento empregado pelos algoritmos MPC parte de uma determinada condição inicial, valendo-se de um problema de otimização para a predição do perfil de controle ótimo ao longo de um determinado horizonte de controle M , onde a última ação é mantida até o final do horizonte de predição P . Para tanto, toma-se como base para o cálculo basicamente a minimização do erro entre os valores preditos pelo modelo para as saídas e os valores de referência (*set points*) existentes. Partindo dessa solução ótima de ações de controle, apenas a primeira ação é aplicada na planta, mantendo-a até o próximo instante de amostragem, onde é então calculada a discrepância entre a predição da planta para aquele instante e a saída real da planta. Essa informação, associada juntamente com os *set points*, é realimentado ao problema de otimização, predizendo novas ações de controle e assim por diante. A figura 5.2 ilustra os conceitos de horizonte de predição e horizonte de controle.

5.2 Tipos de MPCs

O desenvolvimento da tecnologia MPC ocorreu através de uma série de diferentes escolas ao longo das décadas. A figura 5.3 mostra uma árvore evolucionária, apresentando os MPCs industriais mais relevantes, ilustrando a sua conexão de uma maneira concisa.

Em meados de 1970, a primeira descrição de uma aplicação industrial foi apresentada, descrevendo o que seria o MPHC, Controle Preditivo Baseado em Modelo Heurístico. A relação entre as entradas e saídas do sistema dá-se de modo direto, a partir de um função Resposta Impulso Finita (FIR) (Richalet et al. 1978). As saídas do modelo são uma combinação linear dos valores de entrada passados, que são truncados em um ponto onde esses valores não tem mais influência nas saídas futuras. O MPHC foi empregado no software IDCOM, *Identification and Command*, sendo este o precursor de uma gama de outros controladores que viriam a ser elaborados.

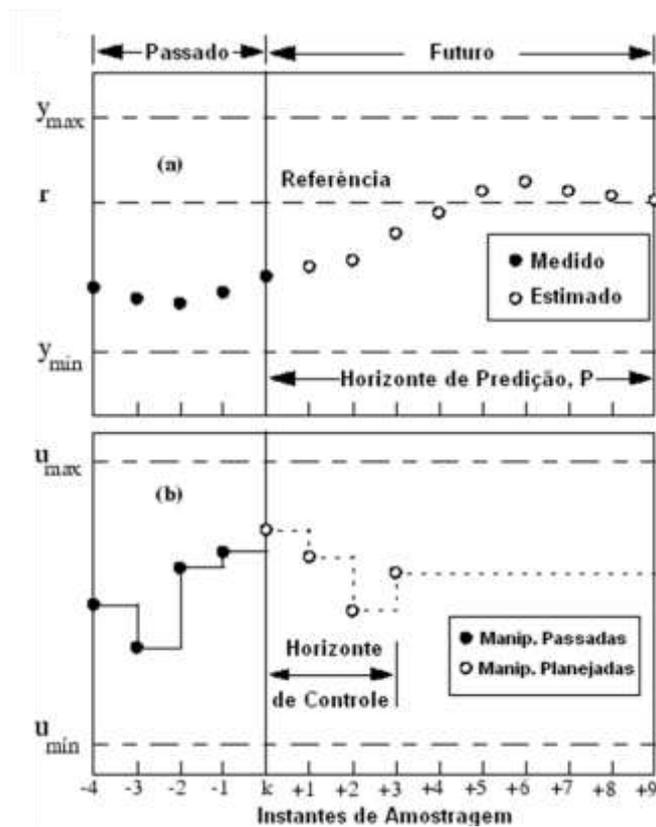


Figura 5.2: Ilustração do horizonte de predição e horizonte de controle. (Figura adaptada de Matlab® Model Predictive Control Toolbox)

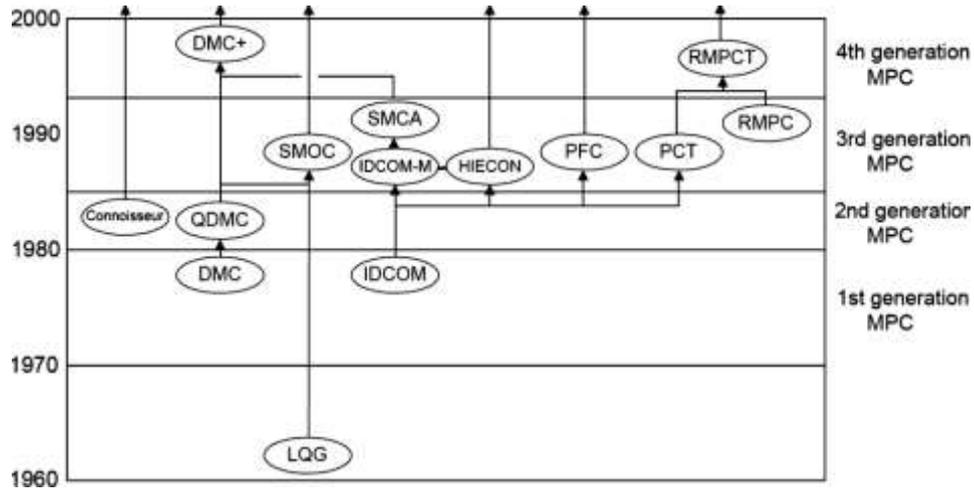


Figura 5.3: Histórico aproximado dos Algoritmos dos MPCs.

FONTES: (Qin e Badgwell 2003)

Uma segunda linha de MPCs foi desenvolvida na mesma época por engenheiros da empresa *Shell Oil*, chamado de DMC, *Dynamic Matrix Control* (Cutler e Ramaker 1980). Diferentemente do MPHC, o DMC trabalha com modelos de resposta degrau lineares, respeitando as equações apresentadas na seção 2.1. Valendo-se do princípio de superposição, possível graças a essa representação, pode-se trabalhar com múltiplas saídas. Utilizando a resposta degrau, o modelo pode prever as mudanças nas saídas do processo como uma combinação linear das entradas futuras no processo. A matriz que relaciona esses dois conjuntos de valores é a chamada matriz dinâmica (DM), conforme apresentada em (2.2), composta pelos coeficientes de resposta degrau das variáveis dentro do horizonte de predição P . Essa tecnologia, em conjunto com o MPHC, caracteriza a primeira geração de MPC desenvolvida.

Conforme mencionado anteriormente na seção 2.3, da linearização de modelos matemáticos que representam sistemas como um conjunto de equações diferenciais, surgiu a representação em espaços de estado. Dentre as aplicações em controladores, os primeiros desenvolvimentos datam da década de 60, tratando de sistemas no domínio discreto representados como modelos em espaço de estado, dando origem ao controlador LQG, Controlador Gaussiano Quadrático Linear (Kalman 1960a, Kalman 1960b). Tal momento representa o advento da teoria que levaria à elaboração do SMOC, Controlador e Otimizador Multivariável Shell, no final da década de 80, considerado como uma ponte entre a tecnologia de espaço de estados e os algoritmos do MPC (Marquis e Broustail 1998, Yousfi e Tournier 1991).

As seções a seguir descrevem em maiores detalhes o MPC baseado na representação discreta em espaços de estado (DEE-MPC), seguido pelo DMC e pela nova abordagem proposta nesta dissertação, *Spline Dynamic Matrix Controller* (SDMC).

5.3 DMC

Conforme expresso na seção 2.1, o modelo do processo é tal como representado em (2.1), composto pelos coeficientes de resposta degrau $S(k)$. Ainda, leva-se em consideração o distúrbio como sendo constante ao longo de todo o horizonte de predição. Os valores preditos tomam, logo, a seguinte forma, conforme é mostrado em (5.1):

$$Y(t+k) = \sum_{i=0}^{\infty} S(i)\Delta U(t+k-i) + d(t+k) \quad (5.1)$$

Onde d é o distúrbio. Sabendo que o mesmo é constante, tem-se que, tal como se segue em (5.2):

$$d(t+k) = Y_d(t) - Y(t) \quad (5.2)$$

Onde Y_d expressa a natureza constante do distúrbio. Substituindo (5.2) em (5.1) e aplicando a predição para m ações de controle, pode-se inferir que, de acordo com (5.3):

$$Y_{[k+P]}^{[k+1]} = DM \cdot \Delta U_{[k+M]}^{[k+1]} + R_{[k+M]}^{[k+1]} \quad (5.3)$$

Onde DM é expresso por (2.2), sendo a matriz dinâmica apresentada na seção 2.1, e R representa a resposta livre do sistema, ou seja, a parte da resposta que independe das ações de controle futuras. Maiores detalhes podem ser vistos em Camacho e Bordons (2004).

Partindo dessa equação, é possível a formulação do problema de otimização que consiste em um problema de programação quadrática, cuja função objetivo pode ser vista em (5.4):

$$F_{obj} = \sum_{j=1}^P [Y(t+j) - Ref(t+j)]^2 + \sum_{j=1}^M \Lambda [\Delta U(t+j)]^2 \quad (5.4)$$

Podendo ser expressa também como em (5.5):

$$F_{obj} = E_y^T E_y + E_u^T \Lambda E_u \quad (5.5)$$

Onde o primeiro termo leva em consideração a discrepância entre o valor de referência e o valor predito da planta ao longo de um horizonte de predição P . O segundo termo penaliza a variação da entrada ao longo de um horizonte de controle M , a fim de evitar variações excessivas nessas variáveis. Λ é um peso que pondera a importância de um termo em detrimento do outro. O problema é sujeito ainda às restrições referentes aos limites de operação das variáveis de entrada e referentes à variação máxima limite das mesmas.

5.4 DEE-MPC

O desenvolvimento de um controlador preditivo baseado em modelo através de espaços de estado parte das equações referentes à linearização do sistema,

culminando nas equações de estado e saída do processo, conforme apresentado no capítulo 2:

$$\Delta x_{k+1} = A_d \Delta x_k + B_d \Delta u_k \quad (2.10)$$

$$\Delta y_{k+1} = C_d \Delta x_{k+1} + D_d \Delta u_{k+1} \quad (2.11)$$

$$Y_{\begin{smallmatrix} [k+P] \\ [k+1] \end{smallmatrix}} = Su \cdot \Delta U_{\begin{smallmatrix} [k+M] \\ [k+1] \end{smallmatrix}} \quad (2.13)$$

Tomando como base um caso mais abrangente, onde distúrbios sejam levados em consideração, a equação (2.10) toma a seguinte forma, conforme visto em (5.6):

$$\Delta x_{k+1} = A \Delta x_k + B_u \Delta u_k + B_d \Delta d_k \quad (5.6)$$

Onde B_u refere-se à matriz que quantifica o impacto das entradas nos estados e B_d é uma matriz análoga, expressando o impacto dos distúrbios medidos nos estados. Partindo dessa representação, é possível se calcular a predição da saída Y ao longo de um horizonte P , conforme expresso por (5.7):

$$Y_{\begin{smallmatrix} [k+1] \\ [k+P] \end{smallmatrix}} = Su \cdot \Delta U_{\begin{smallmatrix} [k+1] \\ [k+M] \end{smallmatrix}} + Sx \cdot \Delta x_k + Sd \cdot \Delta d_k + I_{((P.ny) \times ny)} \cdot y_k + w_k \quad (5.7)$$

Onde a matriz Su é análoga à matriz dinâmica expressa no DMC, Sx e Sd são combinações lineares de A , B e C expressando o efeito da variação dos estados, Δx_k , e dos distúrbios medidos, Δd_k , a cada amostragem na predição e w_k expressa a discrepância entre o valor predito e o valor medido, real, na planta a cada amostragem. Maiores detalhes podem ser vistos em Duraiski (2001).

De posse do cálculo da predição, é possível a construção do problema de otimização, onde a função objetivo é tal conforme se segue em (5.8), para um caso onde *soft constraints* e *targets* (Huang e Kadali 2008) não são contemplados:

$$F_{obj} = \left\| \Gamma_{\begin{smallmatrix} [k+P] \\ [k+1] \end{smallmatrix}} \cdot \left(Y_{\begin{smallmatrix} [k+P] \\ [k+1] \end{smallmatrix}} - Ref_{\begin{smallmatrix} [k+P] \\ [k+1] \end{smallmatrix}} \right) \right\|_2^2 + \left\| \Lambda_{\begin{smallmatrix} [k+M] \\ [k+1] \end{smallmatrix}} \cdot \Delta U_{\begin{smallmatrix} [k+M] \\ [k+1] \end{smallmatrix}} \right\|_2^2 \quad (5.8)$$

Onde Ref representa o valor desejado a ser alcançado pelo controlador, ou a trajetória a ser perseguida e Γ e Λ são pesos a serem arbitrados para ponderação de um termo frente ao outro.

Este problema está sujeito às restrições apresentadas em (5.9) e (5.10):

$$U_{MIN} \leq U \leq U_{MAX} \quad (5.9)$$

$$\Delta U_{MIN} \leq \Delta U \leq \Delta U_{MAX} \quad (5.10)$$

Trata-se de uma minimização de erros quadráticos levando em consideração a discrepância entre o valor predito e a referência, representado pelo primeiro termo, e a variação dos valores de entrada, representado pelo segundo termo.

5.5 Spline Dynamic Matrix Controller (SDMC)

Fazendo o paralelo ao DMC, vale-se de uma matriz dinâmica, no caso a SDM, para a construção do MPC, sendo a mesma responsável pelo cálculo da predição da saída do sistema ao longo das iterações e pela concepção do problema de otimização. O sistema será tratado como sendo discretizado ao longo de um horizonte de predição P , com as entradas variando até um horizonte de controle M . Desse modo, o sistema é equivalente estruturalmente às concepções já mencionadas anteriormente, como DMC e DEE-MPC, o que, para fins de introdução da metodologia, permite uma comparação mais direta dessas abordagens.

Em se tratando dessa representação em particular, é importante salientar que os nós encontram-se fixados nos seus respectivos instantes de tempo, indicando que, na realidade, as únicas variáveis existentes no problema são os valores das entradas nesses pontos. Tendo como base um sistema SISO, a função objetivo do problema de otimização é tal como se segue em (5.11), análoga à expressão apresentada na seção 5.3:

$$F_{obj} = E_y^T \Gamma E_y + E_u^T \Lambda E_u \quad (5.11)$$

Onde E_y representa o erro entre o valor desejado e o valor predito pela planta, ao longo do horizonte de predição, E_u expressa a penalidade quanto à variação da entrada ao longo do tempo, também chamado de *Move Supression*, e Γ e Λ são matrizes peso diagonais quadradas $P \times P$. E_y é dado tal como se segue em (5.12):

$$E_y = Y_{REF} - (Y_{INER} + Y_{BIAS} + w) \quad (5.12)$$

Onde Y_{REF} representa um vetor coluna composto por um valor ou trajetória desejada a ser alcançada pelo MPC ao longo do horizonte de predição. Y_{INER} é uma parcela referente ao impacto que as saídas passadas têm sobre o sistema, sendo obtido pela superposição das simulações dessas respostas a cada amostragem, valendo-se para tanto da equação (4.13), apresentada na seção 4.2. Como o valor de Y_{INER} é calculado em deltas, não em valor absoluto, torna-se necessária a presença de um valor base da saída, referente ao valor do estacionário no início da simulação, para que os cálculos sejam coerentes. A essa parcela dá-se o nome de Y_{BIAS} . Por fim, w expressa a discrepância entre o valor de predição, Y_{PRED_k} , e o valor real na planta em cada amostragem, Y_{PLAN_k} , sendo representado por (5.13):

$$w = Y_{PLAN_k} - Y_{PRED_k} \quad (5.13)$$

Y_{PLAN_k} é obtido por medições na planta a cada amostragem. Y_{PRED_k} deriva de Y_{INER} , sendo igual ao segundo valor desse vetor a cada novo instante, representando a predição, quando no instante k , até $k+1$.

O termo E_u é concebido de um modo particular, dependente da abordagem apresentada até então nesta dissertação, sendo dado por (5.14):

$$E_u = dU * H \quad (5.14)$$

Onde dU é advindo da parametrização das entradas. Como pode ser visto em (4.6), a entrada é uma função analítica. Como a variação da entrada instante a instante pode ser expressa como a derivada de uma função analítica, a matriz dU é construída a partir da derivação da expressão da entrada, aplicada a cada instante de tempo k , para todo o horizonte de predição P . A expressão analítica derivada é tal como se segue em (5.15):

$$du_i(t) = dT(t) dCoef_i TCBMat_i H, \quad i = 1..(z - 1) \quad (5.15)$$

Onde dT e $dCoef$ são representados por (5.16) e (5.17). $dCoef$ é igual à matriz $Coef$, apenas sem a última linha, visto que o termo dT perde um termo na derivação:

$$dT(t) = [3t^2 \quad 2t \quad 1] \quad (5.16)$$

$$dCoef_i = \begin{bmatrix} \left(\frac{2}{\delta_i^3}\right) & \left(\frac{1}{\delta_i^2}\right) & -\left(\frac{2}{\delta_i^3}\right) & \left(\frac{1}{\delta_i^2}\right) \\ \left(-\frac{6t_i}{\delta_i^3} - \frac{3}{\delta_i^2}\right) & \left(-\frac{3t_i}{\delta_i^2} - \frac{2}{\delta_i}\right) & \left(\frac{6t_i}{\delta_i^3} + \frac{3}{\delta_i^2}\right) & \left(-\frac{3t_i}{\delta_i^2} - \frac{2}{\delta_i}\right) \\ \left(\frac{6t_i^2}{\delta_i^3} + \frac{6t_i}{\delta_i^2}\right) & \left(\frac{3t_i^2}{\delta_i^2} + \frac{4t_i}{\delta_i} + 1\right) & \left(-\frac{6t_i^2}{\delta_i^3} - \frac{6t_i}{\delta_i^2}\right) & \left(\frac{3t_i^2}{\delta_i^2} + \frac{2t_i}{\delta_i}\right) \end{bmatrix} \quad (5.17)$$

Ao tratar E_u como uma expressão derivada da SDM, o tamanho da matriz empregada torna-se menor, sendo dependente somente da posição dos nós, assim como a SDM o é.

Ao analisar a função objetivo, percebe-se que cada um dos dois termos expressa uma soma quadrática ponderada ao longo do horizonte de predição, dependente somente da quantidade e da coordenada dos nós empregados para o cálculo das entradas. Sabendo que, em função de ser uma função quadrática, trata-se de um problema convexo de otimização, uma estratégia de programação quadrática (QP) pode ser empregada. A formulação de um QP genérico pode ser vista em (5.18):

$$\min_H \frac{1}{2} H^T Hess H + Q^T H \quad (5.18)$$

Sujeita às restrições apresentadas em (5.19) e (5.20):

$$EH \leq d \quad (5.19)$$

$$GH = f \quad (5.20)$$

Onde $Hess$ é a matriz hessiana e Q expressa a parte linear da função objetivo. As restrições expressas em (5.19) são empregadas para os limites das variáveis de entrada e as expressas em (5.20) são aplicadas para a manutenção de importantes características do MPC. Conforme mencionado anteriormente, a entrada é calculada ao longo de um horizonte de predição P . A entrada, entretanto, só pode

variar até certo horizonte M , menor que P . Para garantir que a partir de M as ações de controle não mudem, o último e penúltimo nós devem ser estrategicamente colocados em M e P , mantendo-os na mesma “altura”, através da restrição de igualdade.

Devido à natureza do MPC, o problema de otimização deve ser resolvido uma vez a cada amostragem, ao longo de um determinado horizonte. A partir de condições iniciais de entrada e saída do sistema, um QP é resolvido para obter um perfil de ações de controle preditas, onde apenas o primeiro valor é aplicado de fato na planta. Até o próximo instante de amostragem, tal ação é aplicada e o comportamento do modelo linear e da própria planta são avaliados, a fim de obter ambos valores no instante seguinte. w é então contabilizado, juntamente com a parcela inercial Y_{INER} . Após isso, procede-se a uma nova etapa de otimização e assim sucessivamente. A característica do problema é, pois, dinâmica.

5.5.1 Ajuste do SDMC

Quando se fala de ajuste de parâmetros, um MPC tradicional trabalha basicamente com o horizonte de predição P , horizonte de controle M , tempo de amostragem T_s e o *Move Supression*, expresso pela ponderação entre os pesos Γ e Λ . A metodologia SDMC, entretanto, disponibiliza parâmetros de ajuste adicionais inexistentes em outras metodologias, relacionados com a disposição e a quantidade de nós no espaço. Essa representação pode ser vista até certo ponto como sendo uma espécie de “*Blocking*”, procedimento utilizado em controladores preditivos visando espaçar e diminuir o número de ações de controle, diminuindo, conseqüentemente, a dimensão do problema de otimização. Para maiores detalhes sobre *blocking*, veja Duraiski (2001).

Diferentemente de outras abordagens, trabalha-se com poucos nós como variáveis de decisão para a determinação do perfil de ações ótimas ao longo do tempo. Isso acarreta, em cada resolução do problema de otimização amostragem a amostragem, uma variação limitada ponto a ponto ao longo do horizonte de predição, visto que o que realmente varia são os nós, ao passo que o restante dos pontos apenas é atualizado de acordo com a *spline* regente naquele intervalo. O impacto disso é, em um primeiro momento, um provável comportamento mais robusto do controlador, sendo este mais acentuado quanto menor for o número de pontos e vice-versa.

Outro fator importante a ser considerado trata da disposição dos nós ao longo de P . Conforme citado em capítulos anteriores, o objetivo principal da metodologia é que os nós sejam postos em regiões que apresentam maior relevância dinâmica. O que isso significa de fato? Em termos práticos, pode-se dizer que são regiões onde haja inflexão ou grande variação em um curto espaço de tempo da variável de entrada. A base dessa escolha reside na qualidade da aproximação dessas entradas pela SDM. Caso a representação não condiga com o sistema de fato, a base para o cálculo das saídas estará alicerçado em uma base incoerente de dados. A própria denominação da SDM indica que o que se deseja é uma matriz dinâmica

do sistema, o que reforça a ideia de que a informação dinâmica deve estar presente e deve ser relevante.

Combinando esses dois fatores com a aplicação em MPC proposta nesta dissertação, alguns pontos importantes devem ser salientados. Inicialmente, vale-se da disposição dos nós para que a teoria do MPC seja aplicada de maneira consistente. Utilizando os parâmetros da Hermite *Spline* TCB, associado à colocação de dois pontos, um em M e outro em P , garante-se que a última ação de controle seja constante de M a P . Outro ponto importante diz respeito ao problema de otimização diretamente. O mesmo deve ser capaz de, se necessário, estabelecer comportamentos característicos de um perfil de ações de controle de um MPC. Ao início de uma simulação onde características servo se mostrem necessárias, por exemplo, um perfil de ações de controle com *overshoot* é frequente, visto que a magnitude das ações de controle iniciais é grande. Sabendo que, para o SDMC, ao menos três nós encontram-se em $t = 0$, $t = M$ e $t = P$, um mínimo de quatro pontos são obrigatórios para que o *overshoot* seja possível.

Juntamente com esses dois fatores, o conceito do *Move Supression* também deve ser levado em consideração. A parcela referente à penalização da variação das entradas é expressa via *splines* através da derivada em cada ponto. Importante salientar, que apesar do horizonte ser discretizado, ele deriva de uma função contínua, ou seja, as derivadas apresentadas não são deltas e sim taxas de variação instantâneas em cada ponto. Apesar de análogo aos outros MPCs, onde a variação da entrada é contabilizada como o delta de ação de controle dado naquele instante, a quantificação desse efeito é distinta, sendo importante avaliar a sensibilidade dos pesos Γ e Λ e da própria formulação frente a diferentes abordagens.

O capítulo 6 vai apresentar os testes realizados para os parâmetros discutidos nessa seção, avaliando a sua sensibilidade e reais efeitos frente à resposta do sistema, comparando com um MPC baseado em RD.

Capítulo 6 – Resultados

A fim de avaliar e validar os conceitos relativos à SDM e sua aplicação no desenvolvimento de um controlador preditivo baseado em modelo, dois estudos de caso foram propostos, um SISO e um MIMO, valendo-se para tanto da modelagem de uma planta laboratorial de três e seis tanques, respectivamente. O trabalho em questão é puramente computacional, onde os dados da planta real são, na realidade, extraídos da resolução do sistema de equações diferenciais que compõem o modelo fenomenológico, não linear, do sistema. As subseções a seguir estruturam a metodologia empregada para a validação das técnicas apresentadas, a ferramenta computacional empregada para este trabalho, seguida da descrição propriamente dita dos dois estudos de caso.

Inicialmente, uma avaliação quanto à simulação dinâmica de processos é apresentada, comparando a resposta do modelo linear obtido através da SDM e do modelo linear obtido através de um algoritmo de referência advinda de um pacote específico em Python, para diferentes perturbações, avaliando a perda de informação. Juntamente com essa análise, o impacto que o posicionamento e o número de nós têm na resposta do sistema é de grande importância, assim como o ajuste dos parâmetros da TCB, merecendo a sua devida discussão.

Ao introduzir o capítulo referente aos resultados do MPC, um cenário base, baseado no que é discutido no mesmo capítulo, deve ser estabelecido, a fim de garantir que a predição das saídas do sistema seja coerente com o sistema em questão e com a teoria de MPC apresentada no capítulo 5.

A fim de validar o MPC proposto, compara-se o mesmo com um já implementado em MatLab®, onde, partindo de um cenário onde ambos os MPCs tenham o mesmo tempo de subida quando submetidos à uma mudança degrau, possa se fazer a análise de seu desempenho comparativamente.

Quesitos como saturação, tempo de amostragem, tempo computacional, *Move Supression* e número de nós, por exemplo, serão abordados para avaliar o

comportamento destes fatores na resposta do controlador. As análises serão realizadas em dois estudos de caso de tanques esféricos, um SISO e outro MIMO.

6.1 Ferramenta Computacional

Em se tratando do desenvolvimento de um trabalho de pesquisa que envolva programação direta e necessidade constante de realização de testes diversos, escolher uma ferramenta versátil e prática é o mais desejado. A metodologia proposta nesta dissertação para a simulação de processos dinâmicos, assim como para o desenvolvimento de um controlador preditivo baseado em modelo alicerçado nessa abordagem, foi realizado a partir de uma linguagem de programação que apresenta tais características: Python.

Python é considerada uma linguagem dinâmica, poderosa, utilizada em uma série de aplicações distintas, desde desenvolvimento de sites, até aplicações em ciência e engenharia (Pérez, Granger e Hunter 2011). Apresenta uma sintaxe simples, intuitiva e de fácil compreensão, com bibliotecas vastas. Trata-se, ainda, de um software livre, o que é uma vantagem considerável frente a softwares comerciais conhecidos, como MatLab®, MAPLE®, entre outros.

A versatilidade da linguagem pode ser constatada ao se perceber que o modo como se programa também pode ser alterado, conforme a preferência do usuário. Python trabalha com linguagem orientada a objeto, para aqueles mais familiarizados com Java, por exemplo, ou com linguagem procedural (*scripts*) para quem está mais acostumado com MatLab®. Além disso, diversas interfaces e pacotes já foram desenvolvidos para permitir a integração dessa linguagem com .NET, PHP, C/C++, Fortran, etc.

Tanto o desenvolvimento do algoritmo que gera a SDM, como toda a implementação do MPC proposto nesta dissertação foi desenvolvido em Python. A versão em questão é a 2.6, onde a redação e interpretação do código foram realizadas através do ambiente de desenvolvimento Eclipse, versão 3.5.2. Eclipse é um ambiente de programação integrado com grandes linguagens, como Java, C/C++ e Python, naturalmente, que permite o desenvolvimento de código de maneira rápida e prática, com uma interface simples e eficiente.

O processador empregado para todos os testes e resultados apresentados nesta dissertação foi um Intel® Core™ i5 CPU 750 @ 2,67 GHz, com 6 GB de RAM.

6.2 Estudos de Caso

6.2.1 Planta Laboratorial de Três Tanques Esféricos

A planta laboratorial de três tanques consiste em uma sequência de três tanques interligados, com uma única alimentação no primeiro tanque, conforme pode ser visto na figura 6.1.

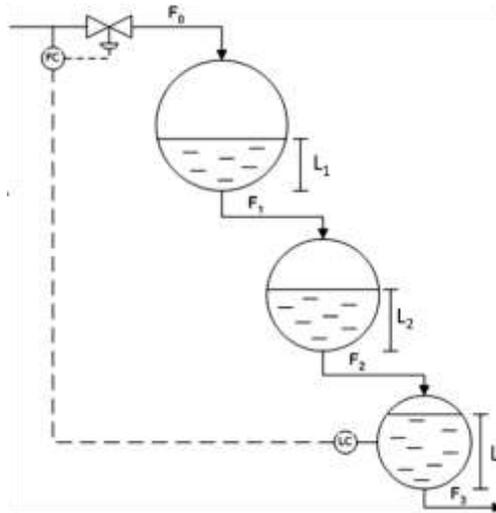


Figura 6.1: Representação esquemática de uma planta laboratorial de três tanques esféricos.

A modelagem desse sistema pode ser visto nas equações de (6.1) a (6.3), tal como se segue:

$$\frac{dL_1}{dt} = \frac{F_0 * 1000 - CD_1 \sqrt{L_1}}{\pi L_1 (D_1 - L_1)} \quad (6.1)$$

$$\frac{dL_2}{dt} = \frac{CD_1 \sqrt{L_1} - CD_2 \sqrt{L_2}}{\pi L_2 (D_2 - L_2)} \quad (6.2)$$

$$\frac{dL_3}{dt} = \frac{CD_2 \sqrt{L_2} - CD_3 \sqrt{L_3}}{\pi L_3 (D_3 - L_3)} \quad (6.3)$$

Onde L_i é o nível de cada tanque, em cm , e F_0 é vazão de entrada no primeiro tanque, em L/min . Os parâmetros são CD_i , o coeficiente de descarga em cada tanque, em $cm^{2.5}/min$, e D_i , diâmetro de cada tanque, em cm . Os valores desses parâmetros são expressos na tabela 6.1.

Tabela 6.1: Parâmetros da planta de três tanques esféricos.

	CD_i ($cm^{2.5}/min$)	D_i (cm)
Tanque 1	1900	30
Tanque 2	2000	30
Tanque 3	2100	30

Em se tratando de controle de processo, o objetivo operacional é o controle do nível L_3 a partir da entrada F_0 .

6.2.2 Planta Laboratorial de Seis Tanques Esféricos

A planta laboratorial de seis tanques consiste de seis tanques de geometria aproximadamente esférica, os quais estão interligados entre si por tubulações e dispostos em dois ramos de três tanques, conforme pode ser visto na figura 6.

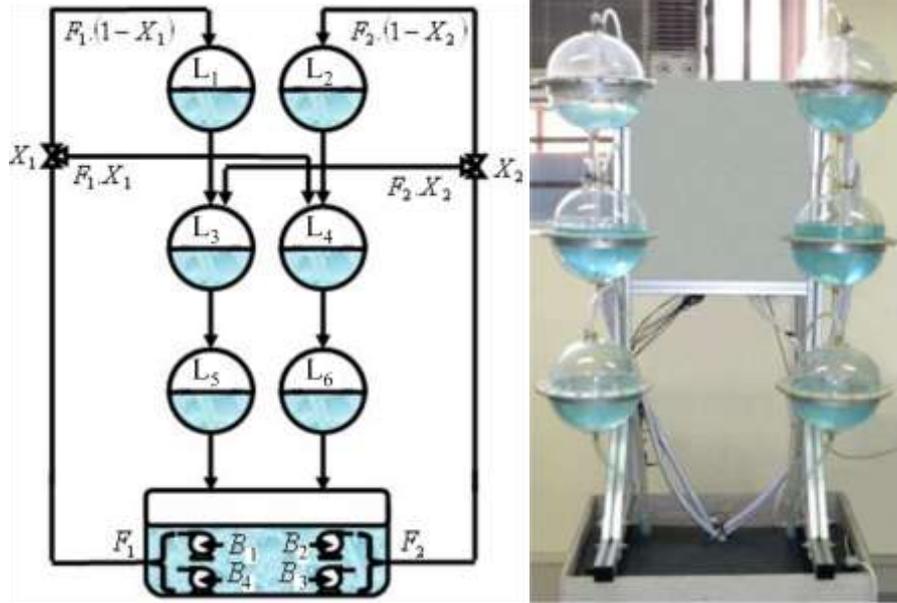


Figura 6.2: Planta laboratorial de seis tanques esféricos.

FONTE: (Paim 2009)

Abaixo destes tanques, encontram-se quatro pequenas bombas submersas em um reservatório de água, cada qual associada com o seu tanque de descarga. As bombas B_1 e B_4 alimentam os tanques 1 e 4 e as bombas B_2 e B_3 alimentam os tanques 2 e 3. Os tanques inferiores 5 e 6 recebem a vazão de saída dos tanques intermediários e suas saídas retornam ao reservatório de origem. As vazões das quatro bombas são modificadas através da variação da tensão de alimentação das mesmas e os níveis dos seis tanques são medidos através de células de carga, que também servem de sustentação para os tanques.

O sistema é modelado conforme se segue, de (6.4) e (6.9):

$$\frac{dL_1}{dt} = \frac{F_1(1-X_1) - CD_1\sqrt{L_1}}{\pi L_1(D-L_1)} \quad (6.4)$$

$$\frac{dL_2}{dt} = \frac{F_2(1-X_2) - CD_2\sqrt{L_2}}{\pi L_2(D-L_2)} \quad (6.5)$$

$$\frac{dL_3}{dt} = \frac{CD_1\sqrt{L_1} + F_2X_2 - CD_3\sqrt{L_3}}{\pi L_3(D-L_3)} \quad (6.6)$$

$$\frac{dL_4}{dt} = \frac{CD_2\sqrt{L_2} + F_1X_1 - CD_4\sqrt{L_4}}{\pi L_4(D-L_4)} \quad (6.7)$$

$$\frac{dL_5}{dt} = \frac{CD_3\sqrt{L_3} - CD_5\sqrt{L_5}}{\pi L_5(D-L_5)} \quad (6.8)$$

$$\frac{dL_6}{dt} = \frac{CD_4\sqrt{L_4} - CD_6\sqrt{L_6}}{\pi L_6(D-L_6)} \quad (6.9)$$

Onde L_i é o nível de cada tanque, em cm , F_1 e F_2 são as vazões de entrada no primeiro e segundo tanques, respectivamente, em L/min . Os parâmetros são CD_i , o coeficiente de descarga em cada tanque, em $cm^{2.5}/min$, e D_i , diâmetro de cada tanque, em cm . Os valores desses parâmetros são expressos na tabela 6.15.

Tabela 6.2: Parâmetros da planta de seis tanques esféricos.

	CD_i ($cm^{2.5}/min$)	D_i (cm)
Tanque 1	0,8179	22,5
Tanque 2	0,8185	22,5
Tanque 3	0,9573	22,5
Tanque 4	0,9595	22,5
Tanque 5	1,0680	22,5
Tanque 6	1,1280	22,5

Os comportamentos dinâmicos desta planta devem ser levados em consideração, pois há a possibilidade do sistema ser de fase mínima e não mínima, decorrentes da existência de um zero de transmissão que se move ao longo do eixo real. Dependendo dos valores de X_1 e X_2 , cada três tanques verticais do sistema representam uma malha distinta, que interagem entre si, a partir da alimentação cruzada nos terceiro e quarto tanques.

Em se tratando de controle do processo, o objetivo operacional é o controle dos níveis intermediários, podendo os níveis restantes ser tratados como variáveis monitoradas (*soft constraints*) através das vazões F_1 e F_2 . Sistemas de fase não mínima apresentam restrições intrínsecas a sua natureza, resultando em resposta inversa, o que dificulta o controle do sistema. Ainda, é importante salientar que quanto mais próximo de 1, tão mais difícil será o controle da unidade, devido à alta interação entre as malhas. Para o caso limite de somatório igual à unidade, torna-se impossível o controle. Aragão (2006) apud Paim (2009) estudou os efeitos de X_1 e X_2 na planta de seis tanques, cujos resultados são apresentados de acordo com a tabela 6.3.

Tabela 6.3: Caracterização de fases do sistema.

Somatório das Frações X_1 e X_2	Característica do Sistema
$X_1 + X_2 > 1$	Sistema de Fase Mínima (Alimentação direta Predominante)
$X_1 + X_2 < 1$	Sistema de Fase Não Mínima (Alimentação indireta Predominante)
$X_1 + X_2 = 1$	Ganho Multivariável Nulo (Não há controle)
$X_1 = X_2 = 1$ ou $X_1 = X_2 = 0$	Sistema Desacoplado

Para os testes realizados nesta dissertação, X_1 e X_2 são fixos em 0,4, garantindo um sistema de fase não mínima. Foi escolhida tal condição em virtude da

dificuldade de controle inerente a esse tipo de condição operacional, o que abre espaço para uma maior gama de análises.

6.3 Resultado do Caso SISO - Planta de Três Tanques Esféricos

Inicialmente, questões relacionadas à simulação dinâmica foram avaliadas. A planta de três tanques esféricos foi submetida a três diferentes perturbações, avaliando, assim, a capacidade de aderência dos modelos lineares a tais entradas. A comparação da qualidade desses modelos foi realizada utilizando uma função referência para simulação de modelos lineares originária do pacote de processamento de sinais do Python (Pacote `scipy.signal.lsim2`).

Inicialmente o sistema foi submetido a um degrau unitário na entrada, conforme pode ser visto na figura 6.3. A fim de avaliar a qualidade de resposta foi variado o número de nós, assim como a sua posição, conforme se segue na tabela 6.4. É importante salientar que cada alteração quanto à característica das *splines*, que está intimamente ligada à colocação dos nós no espaço, acarreta na reconstrução da SDM. Ainda, dentro do horizonte estabelecido por P , os pontos são igualmente distribuídos, com um espaçamento igual a T_s .

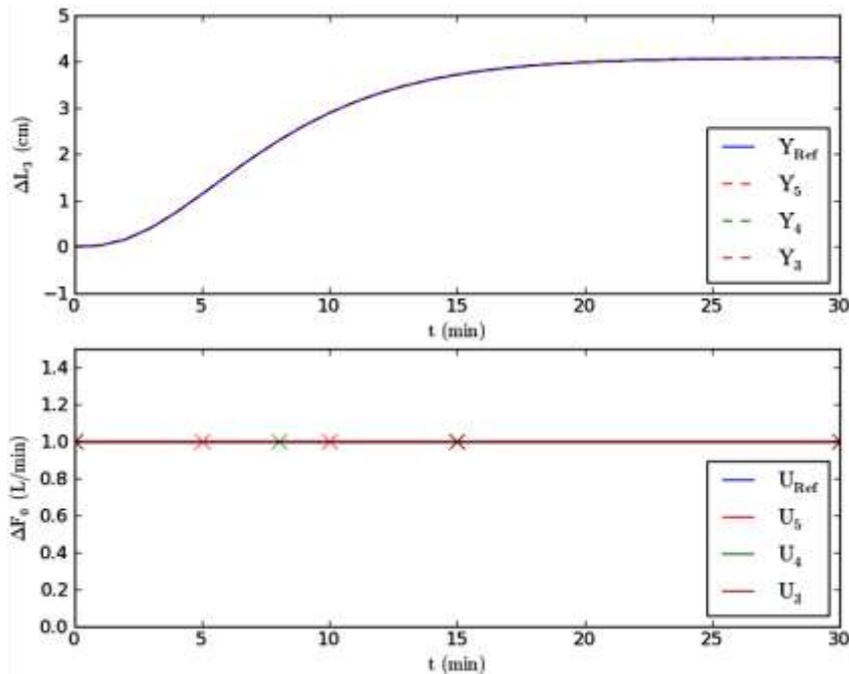


Figura 6.3: Resposta a um degrau unitário na entrada F_0 para três curvas distintas, avaliando o impacto do número e posição dos nós na qualidade da resposta.

Tabela 6.4: Parâmetros das SDMs geradas para uma perturbação degrau.

	Número de Nós (z)	Disposição dos Nós	Parâmetros da TCB		
			Tensão (τ)	Continuidade (γ)	Bias (ω)
U_3	3	[0; 15; 30]	[0; 0; 0]	[0; 0; 0]	[0; 0; 0]
U_4	4	[0; 8; 15; 30]	[0; 0; 0; 0]	[0; 0; 0; 0]	[0; 0; 0; 0]
U_5	5	[0; 5; 10; 15; 30]	[0; 0; 0; 0; 0]	[0; 0; 0; 0; 0]	[0; 0; 0; 0; 0]

Sabendo que a parametrização dos dados dá-se na entrada e não na saída, pode-se perceber que para uma perturbação degrau é irrelevante o número de nós, pois dois apenas são suficientes para a representação exata do degrau. Para o caso da resposta degrau, logo, a característica interpolativa da SDM não é aproveitada.

A figura 6.4 ilustra uma perturbação do tipo sigmoide que segue a expressão em (6.10). Analogamente ao primeiro caso, curvas com diferentes números e posição dos nós foram empregadas, como pode ser visto na tabela 6.5, se focando no efeito do deslocamento e da redundância dos nós.

$$\Delta F_0(t) = \frac{1 + \tanh\left(\frac{t-10}{5}\right)}{2} \quad (6.10)$$

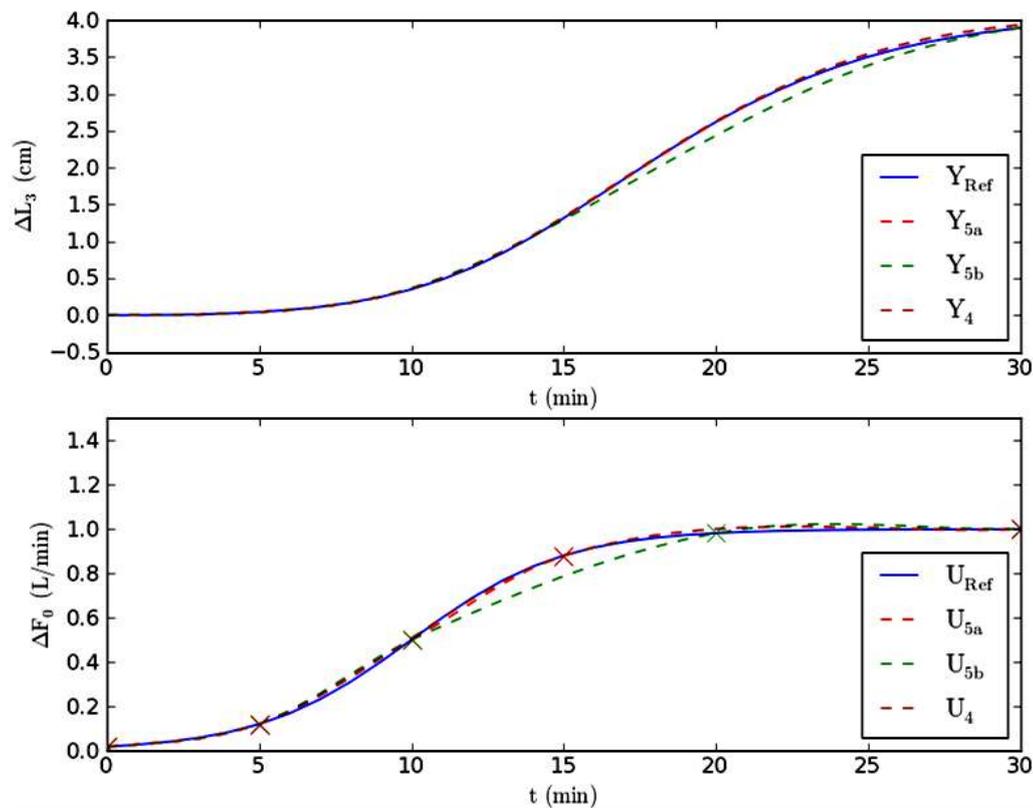


Figura 6.4: Resposta a um perfil de entrada F_0 sigmoide para três curvas distintas, avaliando o impacto do número e posição dos nós na qualidade da resposta.

Tabela 6.5: Parâmetros das SDMs geradas para a reconstrução de uma perturbação do tipo sigmoide.

	Número de Nós (z)	Disposição dos Nós	Parâmetros da TCB		
			Tensão (τ)	Continuidade (γ)	Bias (ω)
U_4	4	[0; 5; 15; 30]	[0; 0; 0; 0]	[0; 0; 0; 0]	[0; 0; 0; 0]
U_{5a}	5	[0; 5; 10; 15; 30]	[0; 0; 0; 0; 0]	[0; 0; 0; 0; 0]	[0; 0; 0; 0; 0]
U_{5b}	5	[0; 5; 10; 20; 30]	[0; 0; 0; 0; 0]	[0; 0; 0; 0; 0]	[0; 0; 0; 0; 0]

Ao comparar a qualidade dos modelos gerados, pode-se ver que, em se tratando da parametrização das entradas, a curva U_{5b} apresentou o pior resultado, mesmo apresentando um número de pontos maior que a curva U_4 , por exemplo. Isso se deve à escolha errônea de colocação de um nó em particular, o quarto, reforçando a ideia do quão importante é o conhecimento da dinâmica das entradas, a fim de uma representação correta do sistema. Por outro lado, ao observar o comportamento da saída do sistema, é possível dizer que, mesmo com discrepâncias, a saída “ruma” à direção certa e apresenta pouca perda de qualidade. Claro que o efeito será tanto maior quanto maior for a discrepância entre as curvas, mas a indicação apenas do valor inicial e final força o sistema a uma resposta satisfatória.

Juntamente com o conhecimento dinâmico do processo, saber reconhecer pontos redundantes à representação também é de suma importância, visto que isso acarreta em igual representação do sistema ganhando em velocidade de computação. A curva U_4 apresenta resultados similares à curva U_{5a} , apesar de apresentar um nó a menos. A tabela 6.6 confirma isso ao apresentar o somatório do erro quadrático para todas as curvas frente à referência.

Tabela 6.6: Somatório do erro quadrático para as perturbações do tipo sigmoide reconstruídas a partir da SDM.

	$\sum_{i=0}^P (Y_{Ref_i} - Y_i)^2$
U_4	0,018
U_{5a}	0,013
U_{5b}	0,242

Qual o impacto, entretanto, dos parâmetros da TCB na simulação das saídas do sistema? Para avaliar esse fator, vale-se de mais uma perturbação, que apresenta diversas oscilações, o que acarreta em um maior número de pontos para a representação. A figura 6.5 apresenta curvas distintas, onde os parâmetros foram trabalhados para garantir melhor qualidade da resposta do sistema. Os parâmetros utilizados para a geração das SDMs pode ser visto na tabela 6.7.

A tabela 6.8 ilustra a qualidade dos modelos frente à referência:

Tabela 6.8: Somatório do erro quadrático para as perturbações oscilatórias reconstruídas a partir da SDM.

	$\sum_{i=0}^P (Y_{Ref_i} - Y_i)^2$
U ₁₉	0,010
U ₁₁	0,260
U _{11Opt}	0,030

Fica evidente pelos resultados apresentados que a representação de um sistema não depende apenas dos nós e de sua localização. Os parâmetros derivados da metodologia TCB quando ajustados corretamente atuam melhorando a aderência das curvas ao padrão original, acarretando na necessidade de menos nós, o que representa ganho computacional.

Em relação aos parâmetros envolvidos no ajuste da TCB, é importante salientar as suas características e aplicabilidades. A Continuidade (γ), em virtude de afetar a suavidade da curva, só é aplicável quando há variações bruscas na entrada, diferentemente dos exemplos aqui citados. Em situações onde tal variação ocorra, entretanto, frequentemente prefere-se preterir esse parâmetro por um maior número de nós na região de maior variação. Manter a suavidade da curva é importante por diversas razões, como, por exemplo, se obter a expressão da derivada contínua ao longo do horizonte, o que pode ser interessante, dependendo da aplicação. Como foi visto na seção 5.5, o SDMC se vale da derivada de $U(t)$ para compor um dos termos do problema de otimização, sendo necessária uma função $U'(t)$ contínua no intervalo.

A Tensão (τ) atua criando ao redor do nó uma determinada zona onde a curva deve manter a direção da tangente naquele ponto. Quanto mais próximo de +1, a zona se expande, havendo uma maior região com aquela tendência, como pode ser visto na seção 3.3.1 na figura 3.6, e vice-versa. Isso impacta na curvatura das splines geradas, o que possibilita, sem afetar a suavidade, um ajuste ao perfil de entradas de referência. O conceito, entretanto, é de difícil aplicação, sendo preterido pelo Bias.

O Bias (ω) é o que apresenta o ajuste mais intuitivo e de fácil aplicação dentre os demais. Como se trata simplesmente de uma ponderação entre as secantes à esquerda e à direita do ponto, a direção da tangente pode ser facilmente modificada, o que afeta diretamente o comportamento da *spline* gerada e permite o ajuste de modo mais simples. Como uma boa métrica, pode-se dizer que, inicialmente procede-se ao ajuste do Bias. Caso esse não seja suficiente para a aderência do modelo, procede-se ao ajuste dos outros parâmetros ou mudam-se o posicionamento e/ou quantidade de nós no espaço.

A utilização da SDM como ferramenta é válida, uma vez que a informação dinâmica relevante seja representada corretamente, valendo-se para tanto da disposição e número de nós no espaço. É importante reforçar o quão isso é importante para que a resposta tenha aderência ao padrão original. Aliado a esses graus de liberdade, o ajuste dos parâmetros derivados da metodologia TCB para geração de Hermite *splines* atua de duas maneiras distintas, melhorando a qualidade do modelo e reduzindo o número total de nós necessários. Pode-se dizer que a elaboração da SDM trata, no final das contas, de um compromisso entre os nós, ligados a velocidade e robustez da resposta, e a qualidade final do modelo frente a diferentes tipos de entrada.

O trunfo existente por trás da SDM, entretanto, não está na simulação dinâmica apenas, mas sim na aplicação desta ferramenta em outras técnicas, como na formulação de um novo tipo de controlador, SDMC, em cujo núcleo a SDM reside. Para a planta laboratorial apresentada na figura 6.1, um MPC linear SISO foi desenvolvido utilizando o modelo SDM, cujos pontos discretizados são igualmente espaçados com base em T_s ao longo de P . O fluxo F_0 atua como entrada, controlando o nível L_3 do terceiro tanque, que foram empregados para a avaliação do desempenho em geral e robustez deste novo MPC, quando comparado com um MPC linear baseado em resposta degrau. A figura 6.6 mostra uma resposta degrau frente ao modelo linearizado para os três casos.

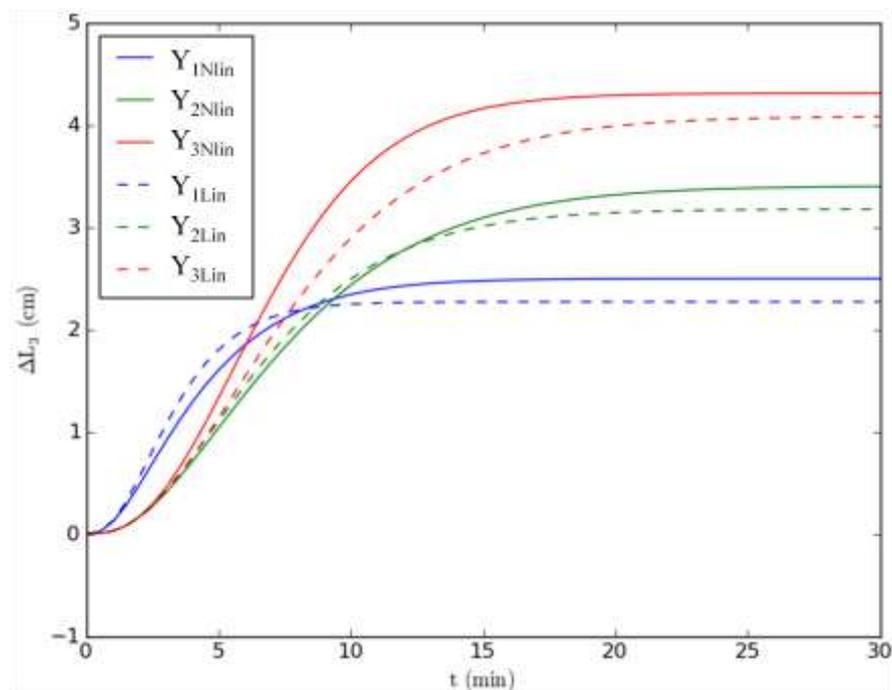


Figura 6.6: Resposta a um degrau unitário na entrada F_0 para três diferentes pontos de operação em uma planta laboratorial, mostrando a diferença entre o modelo linear (linhas tracejadas) e não linear (linhas sólidas).

A tabela 6.9 e as equações (6.11) a (6.13) mostram os valores em estado estacionário e as funções de transferência equivalentes a W_i , respectivamente.

Tabela 6.9: Valores estacionários das alturas em cada tanque e da vazão de entrada para os três pontos de operação.

	L_1 (cm)	L_2 (cm)	L_3 (cm)	F_0 (L/min)	Ponto de Operação (PO)
W_1	6,92	6,25	5,67	5,0	BAIXO
W_2	13,57	12,25	11,11	7,0	MÉDIO
W_3	22,44	20,25	18,37	9,0	ALTO

$$W_1 = \frac{1,423}{s^3 + 2,595s^2 + 2,222s + 0,628} \quad (6.11)$$

$$W_2 = \frac{0,234}{s^3 + 1,264s^2 + 0,530s + 0,073} \quad (6.12)$$

$$W_3 = \frac{0,201}{s^3 + 1,100s^2 + 0,403s + 0,049} \quad (6.13)$$

Partindo desses dados, propõe-se projetar um controlador que possa controlar o sistema em todos os pontos de operação, comparando-o com um controlador similar baseado em resposta degrau (RD) projetado para a mesma condição. É importante salientar que a planta real é representada pela simulação não linear das equações diferenciais que compõem o modelo do sistema.

Como o objetivo é se escolher uma dinâmica para a qual o projeto do controlador permita que o mesmo venha a trabalhar sem instabilização nos demais pontos de operação, optou-se pelo terceiro ponto de operação (PO), visto que é a resposta que apresenta, conjuntamente, a dinâmica mais lenta e o maior ganho, como se pode verificar através da figura 6.6.

A fim de poder estabelecer um critério de comparação, visto que são técnicas com abordagens distintas para a elaboração do MPC, parte-se do pressuposto que, de acordo com a resposta do sistema ao controlador projetado para o PO 3 aplicado no PO 3, os MPCs serão ditos equivalentes quando o tempo de subida referente à mudança de *set point* for similar. Tal resposta pode ser vista na figura 6.7.

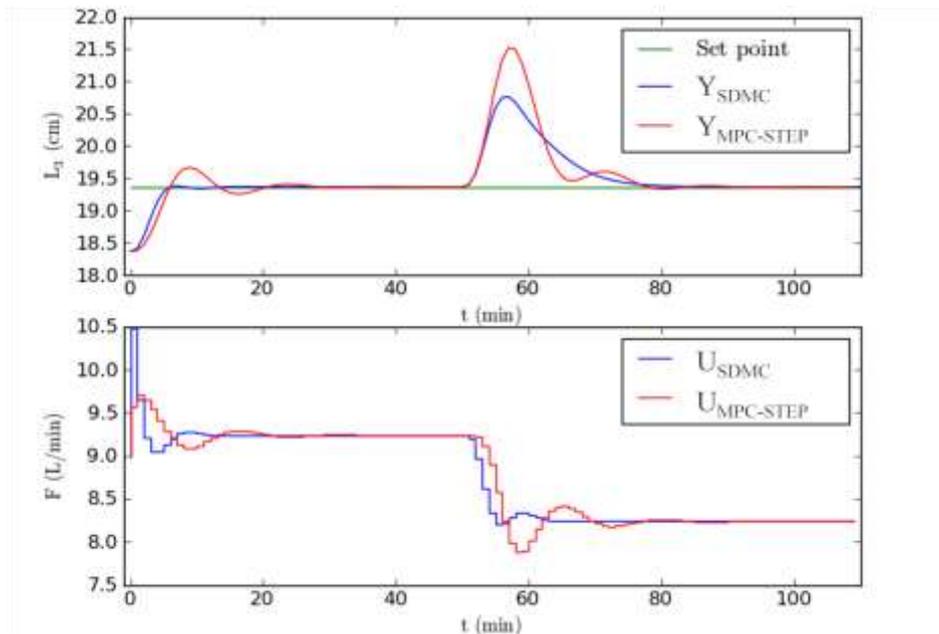


Figura 6.7: Simulação da planta de três tanques expressando o comportamento de dois controladores projetados para o PO 3 aplicados no PO 3, quando submetidos à mudança de *set point* e distúrbio na carga unitários.

Onde os parâmetros empregados para cada controlador podem ser vistos nas tabelas 6.10.

Tabela 6.10: Parâmetros base empregados para o SDMC e o MPC-STEP.

	P (min)	M (min)	T_s (min)	z	Disposição dos Nós	Γ	Λ
SDMC	30	8	1	5	[0;3;6;8,30]	1	0,5
MPC-STEP	30	10	1	-	-	1	1,2

Em relação ao posicionamento dos nós, a figura 6.8 ilustra bem esse conceito, mostrando a cada amostragem a previsão do controlador, juntamente com a posição dos nós no sistema, que respeitam o horizonte de previsão P e horizonte de controle M determinados.

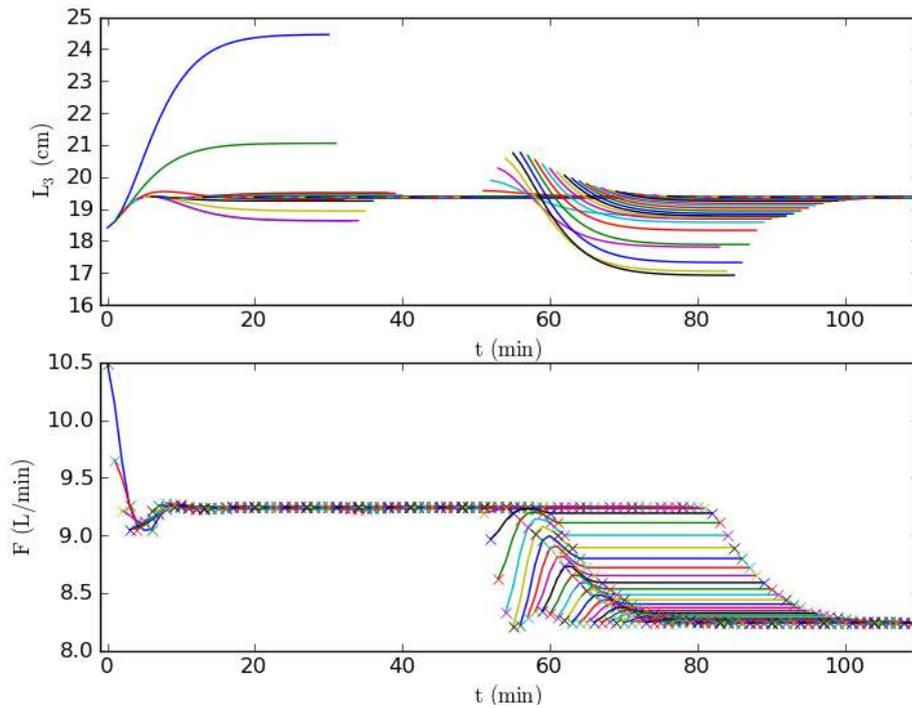


Figura 6.8: Predição do MPC amostragem a amostragem, indicando, para a entrada, a posição otimizada dos nós em cada iteração.

Como ferramenta didática, a figura 6.8, especialmente na região após o distúrbio na entrada ter sido aplicado, destaca que a variação na entrada, a cada vez que o problema de otimização é resolvido, ocorre apenas até o penúltimo nó, em M , sendo coerente com a teoria de MPC apresentada até então.

De posse dos controladores em questão, o desempenho dos mesmos nos pontos de operação 1 e 2 pode ser avaliado, conforme pode ser visto nas figuras 6.9 e 6.10, respectivamente. Percebe-se que o SDMC, quando comparado com o MPC-STEP, apresenta uma resposta menos oscilatória, atingindo o *set point* rapidamente. Além disso, rejeita bem distúrbios, sendo equivalente ou melhor que o MPC em questão.

Quais seriam as razões para este melhor desempenho, em um primeiro momento? Ao pensar sobre a natureza da SDM, percebe-se que, ainda que haja um horizonte discretizado de pontos que representam a entrada, apenas uma pequena quantidade de nós confere a variação de fato à curva, assim garantindo um comportamento menos oscilatório e, conseqüentemente, mais robusto. Os pontos discretizados ao longo do horizonte apenas são utilizados para o cálculo da soma do quadrado dos erros que compõem a função objetivo expressa em (5.11).

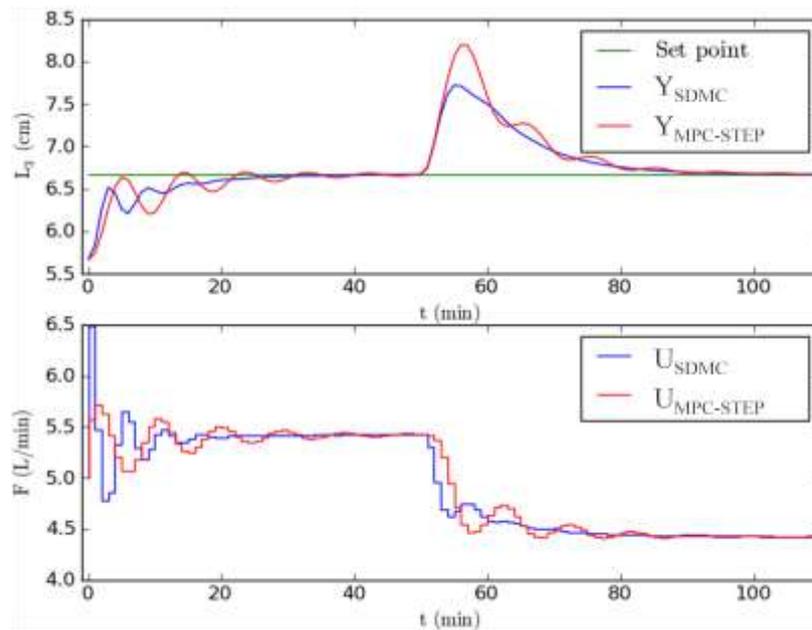


Figura 6.9: Simulação da planta de três tanques expressando o comportamento de dois controladores projetados para o PO 3 aplicados no PO 1, quando submetidos à mudança de *set point* e distúrbio na carga unitários.

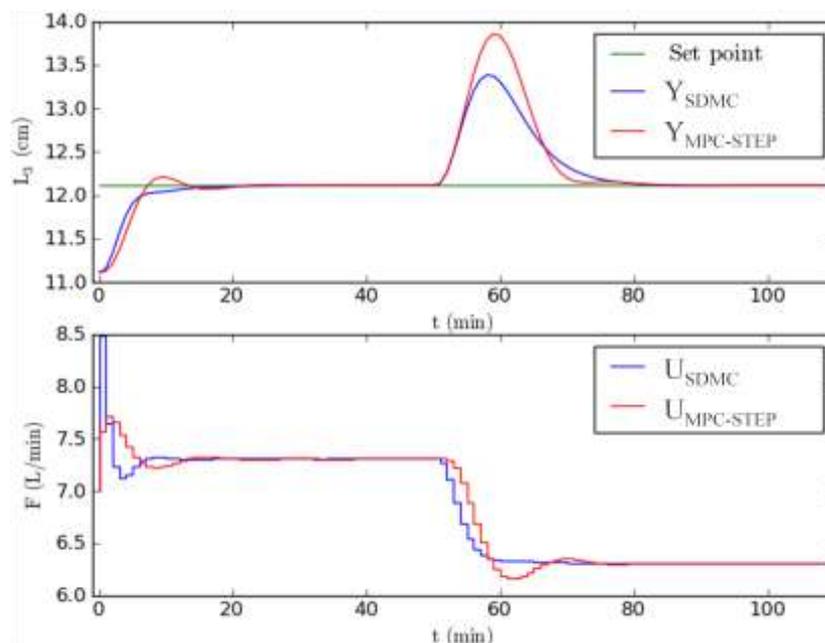


Figura 6.10: Simulação da planta de três tanques expressando o comportamento de dois controladores projetados para o PO 3 aplicados no PO 2, quando submetidos à mudança de *set point* e distúrbio na carga unitários.

Seguindo ainda os mesmos parâmetros, ao avaliar o desempenho de um MPC questões relacionadas à sua operabilidade vem à tona, como saturação, por exemplo. As figuras 6.11 e 6.12 ilustram o comportamento do SDMC quando submetido a restrições operacionais de máximo e mínimo, respectivamente.

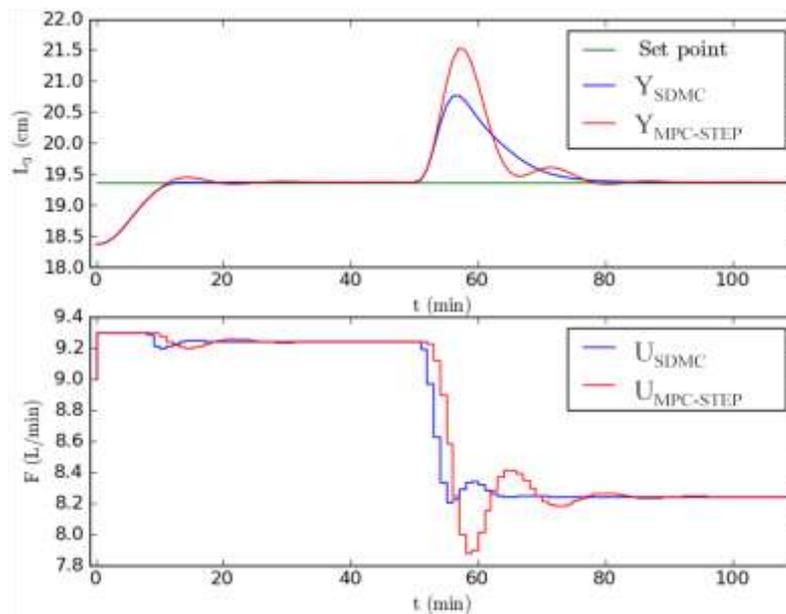


Figura 6.11: Simulação da planta de três tanques expressando o comportamento de dois controladores projetados para o PO 3 aplicados no PO 3, quando submetidos à mudança de *set point* e distúrbio na carga unitários com restrição superior na entrada de 9,3 L/min.

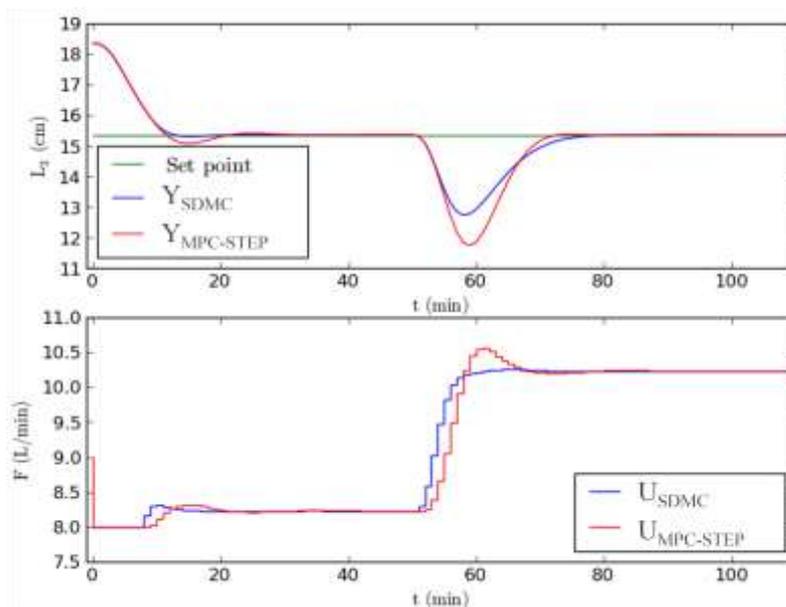


Figura 6.12: Simulação da planta de três tanques expressando o comportamento de dois controladores projetados para o PO 3 aplicados no PO 3, quando submetidos à mudança de *set point* e distúrbio na carga unitários com restrição inferior na entrada de 8 L/min.

Como pode ser observado, o SDMC se comporta bem no caso de saturação, independentemente do tipo de restrição, sendo mais rápido e menos oscilatório que o MPC-STEP, o que valida a sua atuação em sistemas restritos.

Quais são os fatores de fato, entretanto, que afetam o comportamento do SDMC? Além de questões aplicáveis a outros MPCs, como variação do tempo de amostragem e dos horizontes, questões referentes à construção da SDM são

relevantes, como o número de nós, principal fator responsável pela robustez do sistema, e o *Move Supression*, visto que a sua estrutura difere da aplicação tradicional. As subseções seguintes discorrem sobre a avaliação desses quesitos, tratando também de questões relacionadas ao impacto do tempo de amostragem e dos nós no tempo computacional envolvido em cada aplicação.

6.3.1 Número de Nós

Ao avaliar o número de nós e o seu efeito no desempenho do MPC, foi comparada a resposta de duas situações extremas, onde duas concepções de SDM distintas foram utilizadas, de acordo com os parâmetros apresentados na tabela 6.11, mostrando o contraste entre o maior número de nós possível para aquela amostragem e o menor número de nós possível que mantenha o comportamento característico das ações de controle ao longo do tempo. O PO escolhido para a análise dessa característica, ao contrário das outras análises, foi o ponto 1, visto que é a região que apresenta a menor estabilidade em termos de controle. Os resultados são comparados com um MPC-STEP de condições semelhantes às apresentadas na tabela 6.10, tendo apenas o tempo de amostragem alterado para 0,5 min. Vale salientar que o *Move Supression* foi reduzido a um quinto do seu valor original, de modo a pouco influenciar a oscilação da resposta obtida. Os resultados podem ser vistos na figura 6.13.

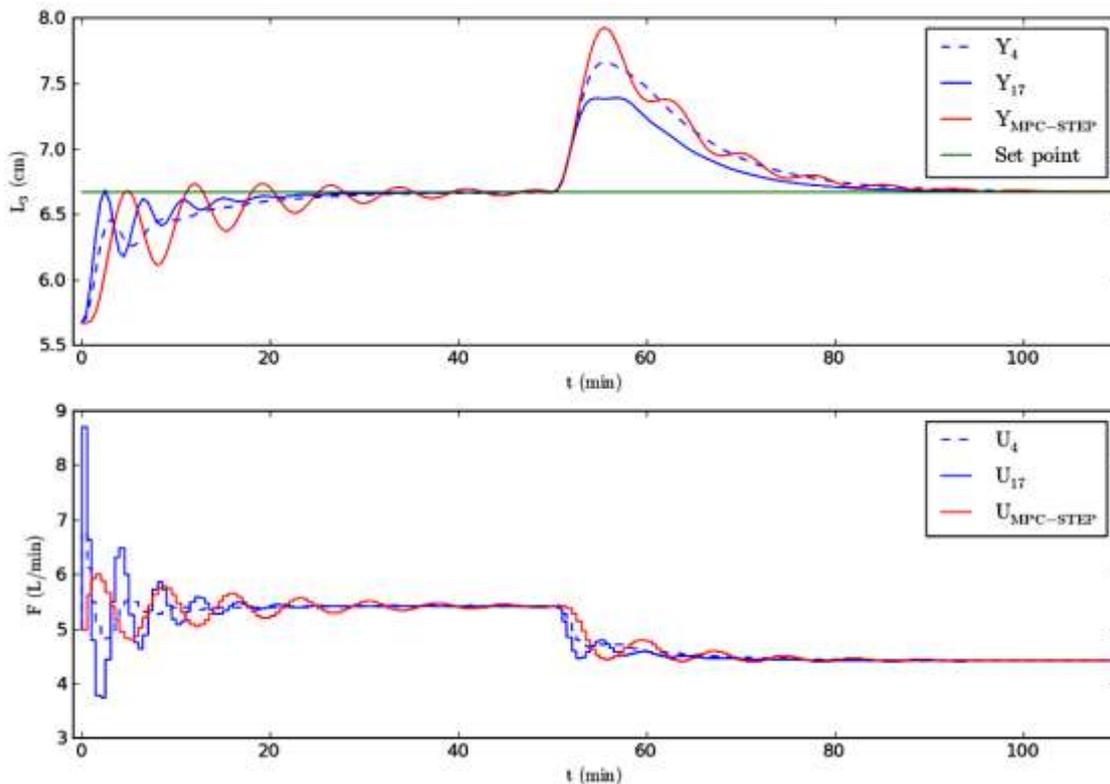


Figura 6.13: Simulação da planta de três tanques expressando o comportamento de dois controladores projetados para o PO 3 aplicados no PO 1, onde dois SDMCs com número de pontos distintos são empregados.

Tabela 6.11: Parâmetros empregados para o SDMC.

	P (min)	M (min)	T_s (min)	z	Disposição dos Nós	Γ	Λ
U_4	30	7,5	0,5	4	[0;4;7,5;30]	1	0,02
U_{17}	30	7,5	0,5	17	[0;0,5;1;1,5;2;2,5;3;3,5;4;4,5; 5;5,5;6;6,5;7;7,5;30]	1	0,02

O número de nós afeta a resposta do sistema, corroborando o que foi citado anteriormente quanto à liberdade que é dada para o sistema. U_{17} teve uma maior liberdade de variação, o que acarretou em maior oscilação e respostas mais rápidas do que as obtidas pela U_4 , justamente pela diferença no número de nós. Ainda assim, a curva mais lenta gerada pelo SDMC apresenta, através de uma curva robusta, um desempenho superior à curva gerada pelo MPC-STEP.

6.3.2 Move Supression

A característica do *Move Supression* implementado no SDMC é diferente das formulações convencionais, sendo necessária a análise e validação do seu comportamento quando na simulação de sistemas submetidas a perturbações do tipo servo e regulatório. A figura 6.14 ilustra esse conceito, tomando como base os parâmetros expressos na tabela 6.12 para os controladores aplicados no PO 3, onde o desempenho dos controladores foi testado frente a valores de *Move Supression* vinte vezes maiores e vinte vezes menores para ambos os MPCs.

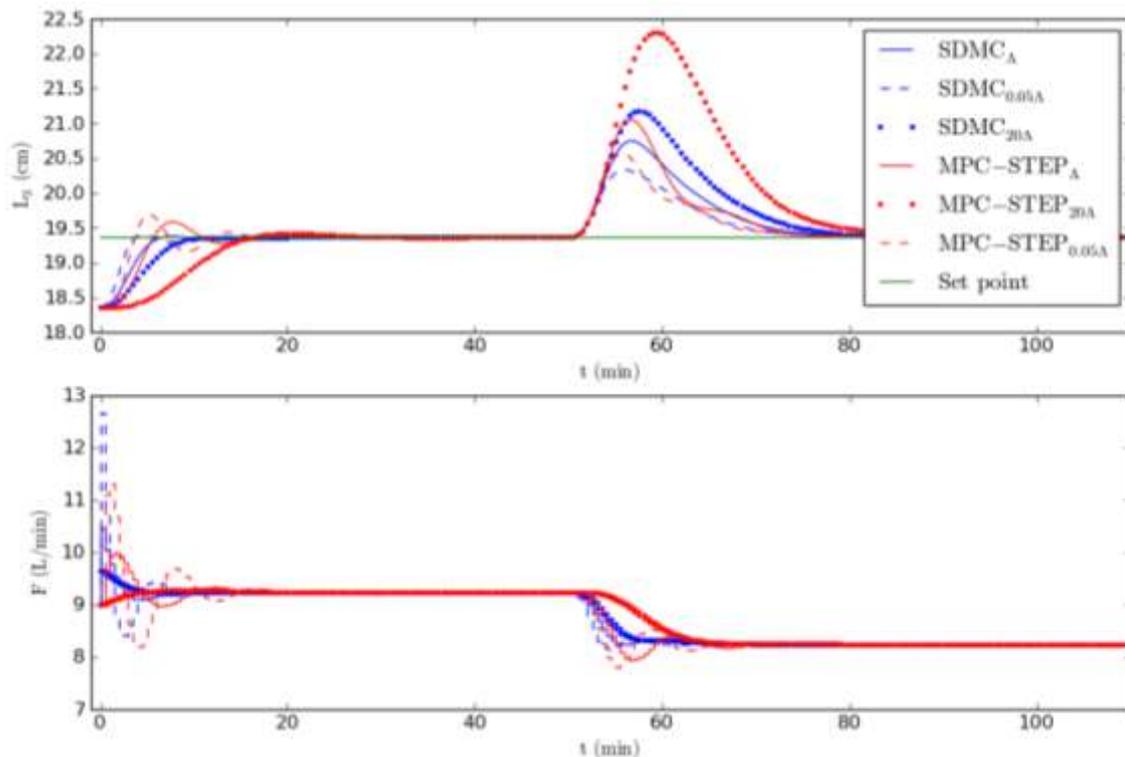


Figura 6.14: Simulação da planta de três tanques expressando o comportamento de controladores projetados para o PO 3 aplicados no PO 3, variando o *Move Supression*.

Tabela 6.12: Parâmetros base empregados para o SDMC e o MPC-STEP.

	P (min)	M (min)	T_s (min)	z	Disposição dos Nós	Γ	Λ
SDMC	30	7,5	0,5	8	[0;1,5;2,5;4; 5;6,5;7,5; 30]	1	0,5
MPC-STEP	30	10	1	-	-	1	1,2

O *Move Supression* afeta a resposta de forma análoga ao número de nós, visto que o termo justamente lida com a variação da entrada ao longo do horizonte de controle, representado pela derivada das *splines*. O efeito desse parâmetro, quando comparado com o MPC-STEP não apresenta grande sensibilidade, em virtude do número de nós também influenciar na robustez da resposta final. Para o MPC-STEP, entretanto, como somente o *Move Supression* atua restringindo a variação de U , a variação do mesmo afeta significativamente o tipo de resposta obtida. Importante notar que o MPC-STEP poderia se valer do *blocking* para um comportamento similar ao número de nós. Isso, entretanto, não foi analisado.

Para o SDMC, logo, o número de nós e o *Move Supression* devem ser analisados em conjunto no momento da definição dos parâmetros do MPC, a fim de garantir a robustez do sistema, mas com respostas rápidas.

6.3.3 Tempo Computacional

Em se tratando da simulação do sistema com a atuação do SDMC, o tamanho da SDM gerada e o tempo de amostragem do sistema representam os dois fatores que afetam a velocidade do sistema.

As dimensões da SDM dependem do número de nós, que afetam o número de colunas da matriz, e do tempo de amostragem, que influencia no número de linhas. Em se tratando do algoritmo executado pelo MPC, a cada amostragem há a predição das saídas para todo o horizonte de predição, que consiste na multiplicação da SDM pelo vetor H das entradas, e a resolução do problema de otimização, que depende basicamente do tamanho da Hessiana empregada, que é proporcional ao tamanho das variáveis de decisão, os nós.

O número de nós é menor que o número de pontos ao longo do horizonte de controle M . A resolução do problema, logo, deve ser mais rápida que a das formulações tradicionais, visto que essas empregam todos os pontos como variáveis de decisão, o que acarretaria em um tempo computacional total menor para a metodologia SDMC. A tabela 6.13 ilustra os resultados referentes à velocidade de simulação apresentados para o caso base apresentado na tabela 6.10, para diferentes tempos de amostragem.

Tabela 6.13: Tempo de simulação total e da parcela referente à otimização para dois controladores, em função do tempo de amostragem.

T_s (min)	Tempo SDMC (s)	Tempo Otimização SDMC (s)	Tempo MPC-STEP (s)
1	2,594	0,096	0,610
0,5	9,656	0,361	1,2500
0,25	38,45	0,545	2,45

Os resultados mostram que a simulação através do SDMC é consideravelmente mais lenta do que a realizada pelo MPC-STEP. Quais as razões para isso, visto que o tempo levado para otimização é de fato menor? A realimentação dos dados, assim como o cálculo da parcela inercial amostragem a amostragem é provavelmente deficiente, em termos de implementação. À exceção do problema de otimização, o restante da implementação apresenta problemas quanto à execução dos comandos de modo eficiente, sendo necessário em um segundo momento, a otimização do código para um melhor aproveitamento da técnica.

A tabela 6.14 analogamente ilustra a diferença de velocidade na resolução, variando, entretanto, o número de nós, quando mantido o tempo de amostragem em 0,5 min.

Tabela 6.14: Tempo de simulação total e da parcela referente à otimização para dois controladores em função do número de nós.

z	Tempo SDMC (s)	Tempo Otimização SDMC (s)
5	9,719	0,282
11	15,87	0,315
17	22,34	0,331

Pode-se perceber que o tempo total de simulação, assim como o tempo gasto somente para o problema de otimização aumenta à medida que o número de variáveis de decisão aumenta, devido a uma hessiana maior, o que confirma a proposição de ganho de velocidade com a redução da quantidade de variáveis de decisão disponíveis. A discrepância não é tão acentuada, pois, em virtude do sistema em questão ser simples, provavelmente o tempo mínimo de computação do algoritmo de otimização, inerente a sua implementação, não difere muito do tempo necessário para a resolução do problema aqui apresentado.

6.4 Resultado do Caso MIMO - Planta de Seis Tanques Esféricos

A análise de sistemas multivariáveis permite, além da validação da metodologia SDM por si só, o estudo de determinadas características inexistentes em sistemas monovariáveis (SISO). Ao longo desta seção, questões relacionadas à simulação dinâmica e à concepção do SDMC serão levadas em consideração, tratando mais

especificamente das peculiaridades inerentes a uma representação multivariável e o impacto que a metodologia SDM tem no comportamento desse sistema.

Para a planta de seis tanques, o objetivo operacional é o controle do nível dos tanques intermediários 3 e 4, através das vazões F_1 e F_2 , que apresentam alimentação direta nos tanques superiores 1 e 2 e alimentação cruzada nos tanques 3 e 4, como já foi apresentado na seção 6.2.2. Os testes nesta seção são realizados no ponto de operação apresentado na tabela 6.15. A resposta degrau do sistema, simultaneamente para perturbações em ambas as vazões de entrada, pode ser vista na figura 6.15, para o modelo linear, a partir da SDM, e o modelo não linear.

Tabela 6.15: Valores estacionários das alturas em cada tanque e da vazão de entrada.

L_1 (cm)	L_2 (cm)	L_3 (cm)	L_4 (cm)	L_5 (cm)	L_6 (cm)	F_1 (L/min)	F_2 (L/min)
6,59	6,58	13,37	13,31	10,74	9,63	3,5	3,5

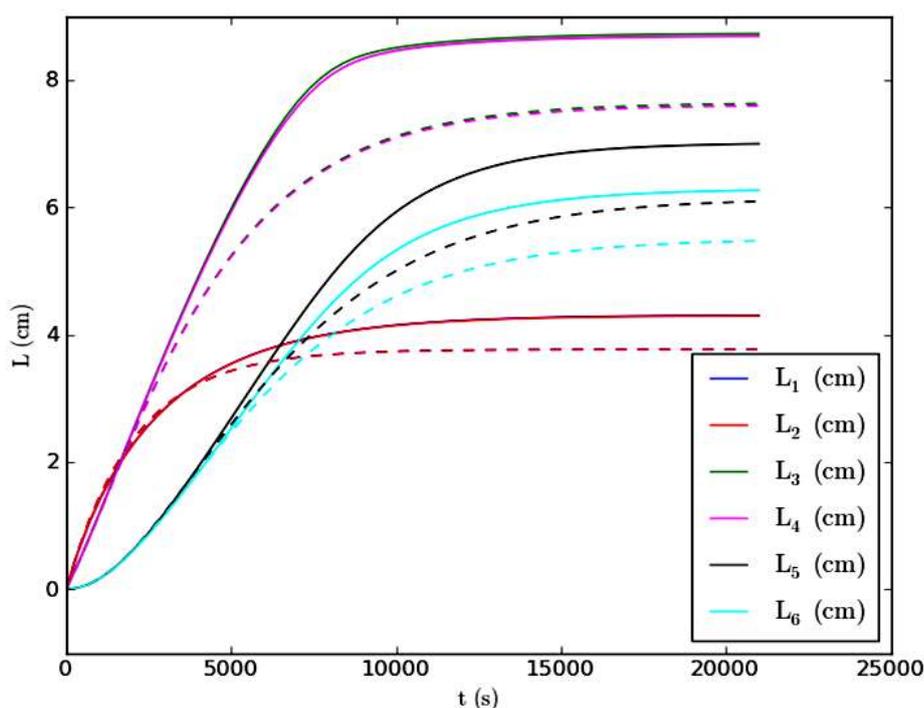


Figura 6.15: Resposta degrau na planta de seis tanques, comparando o modelo linear com o não linear.

Em se tratando de simulação de processos dinâmicos, uma das maiores vantagens da SDM está relacionada à representação independente de cada conjunto de entradas. É interessante que cada entrada tenha o seu grupo de nós que a represente, visto que isso abre espaço à representação de sistemas rígidos, i.e., quando a dinâmica das entradas é discrepante. A figura 6.16 ilustra esse comportamento, onde fica evidente a vantagem da utilização da SDM. Ambas as entradas são representadas com um número reduzido de nós, mantendo as suas características, sem que o número e a posição dos nós sejam coincidentes.

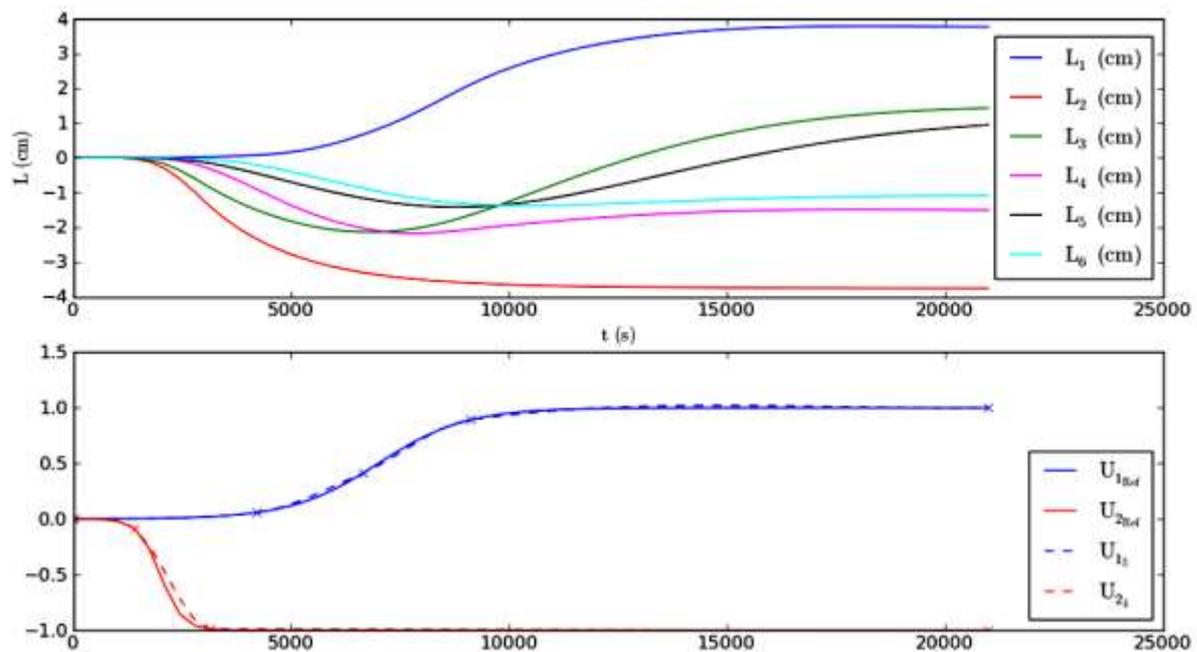


Figura 6.16: Parâmetros das SDMs gerada para a reconstrução de perturbações do tipo sigmoide, com dinâmicas diferentes.

Os parâmetros utilizados para a geração dessas entradas podem ser visto na tabela 6.16:

Tabela 6.16: Parâmetros das SDMs geradas para a reconstrução de uma perturbação do tipo sigmoide.

	Número de Nós (z)	Disposição dos Nós	Parâmetros da TCB		
			Tensão (τ)	Continuidade (γ)	Bias (ω)
U_{15}	5	[0; 4200; 6650; 9100; 21000]	[0; 0; 0; 0; 0]	[0; 0; 0; 0; 0]	[0; 0,5; 0; -0,5; 0]
U_{24}	4	[0; 1400; 3150; 21000]	[0; 0; 0; 0]	[0; 0; 0; 0]	[0; 0; -1; 0]

Para o MPC proposto nesta dissertação, os testes aqui apresentados comparam o desempenho do SDMC a um MPC baseado na representação em espaços de estado no domínio discreto (DEE-MPC). Esse controlador foi implementado em Python pelo mestrando Eng. Rafael Cardoso Dias Costa, podendo ser aplicado para sistemas mono e multivariáveis, contemplando, caso desejado, *soft constraints* e *targets*. Para os resultados apresentados, entretanto, essas características não serão levadas em consideração.

O controlador foi sintonizado através da metodologia RPN, *Robust Performance Number*, que quantifica a dificuldade que o sistema tem para atingir o desempenho desejado de forma robusta, visto em maiores detalhes no trabalho de Trierweiler e Engell (1997). A sintonia foi realizada através do *RPN Toolbox*, ferramenta desenvolvida pelo Grupo de Intensificação, Modelagem, Simulação,

Controle e Otimização de Processos da UFRGS, GIMSCOP. Através dessa metodologia obtém-se matrizes de escalonamento do sistema, R e L , que associadas aos pesos Γ e Λ das variáveis controladas e manipuladas, respectivamente, fornecem os parâmetros de sintonia do MPC propriamente dito, U_m e Y_m , como pode ser visto em (6.14) e (6.15):

$$U_m = R^{-1} \Lambda \quad (6.14)$$

$$Y_m = \Gamma L \quad (6.15)$$

Para o sistema estudado, os valores desses parâmetros, assim como os valores do tempo de amostragem, horizonte de predição e horizonte de controle podem ser vistos na tabela 6.17.

Tabela 6.17: Parâmetros base para o DEE-MPC.

R	L	Γ	Λ	$P(s)$	$M(s)$	$T_s(s)$
$\begin{bmatrix} 1 & 0 \\ 0 & 1,004 \end{bmatrix}$	$\begin{bmatrix} 0,3237 & 0 \\ 0 & 0,3282 \end{bmatrix}$	$\begin{bmatrix} 0,7654 & 0 \\ 0 & 1,848 \end{bmatrix}$	$\begin{bmatrix} 0,6089 & 0 \\ 0 & 1,470 \end{bmatrix}$	21000	5250	350

Para o SDMC, adotou-se como métrica neste caso estabelecer os mesmos parâmetros do DEE-MPC quanto ao horizonte de predição, horizonte de controle e tempo de amostragem, ajustando os demais parâmetros de modo arbitrário. Os parâmetros podem ser vistos na tabela 6.18.

Tabela 6.18: Parâmetros base empregados para o SDMC.

$P(s)$	$M(s)$	$T_s(s)$	z	Disposição dos Nós	Γ	Λ
21000	5250	350	[5,5]	$[0;1750;3500;5250,21000]$ $[0;1750;3500;5250,21000]$	1	0,5

Em se tratando de um sistema multivariável, a direção da perturbação das entradas é de grande importância. Quando se fala em controle de planta, a perturbação na direção do zero de transmissão representa o pior cenário possível, onde o controle é, pois, mais difícil. De acordo com os parâmetros vistos na tabela 6.17, pode-se observar que a pior perturbação no sistema ocorre na variação positiva do *set point* de L_3 e na variação negativa de L_4 , ambos com a mesma magnitude. A figura 6.17 ilustra essa situação, onde o desempenho dos dois controladores é comparado.

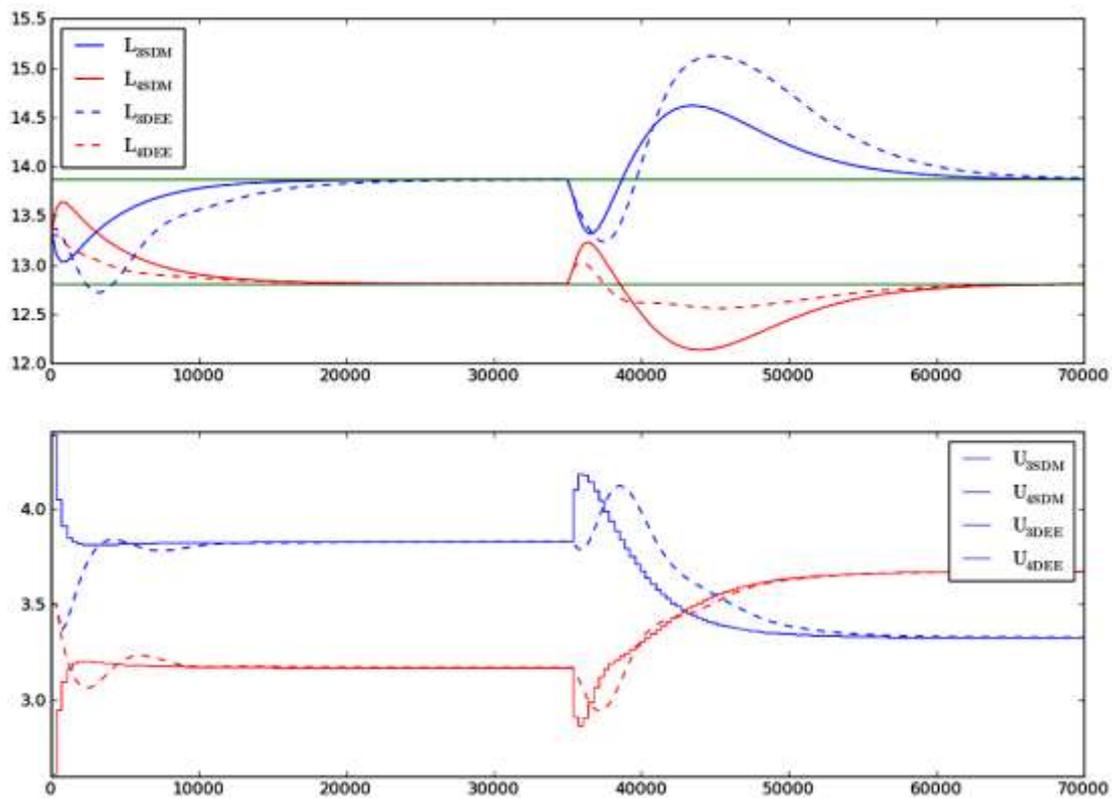


Figura 6.17: Simulação da planta de seis tanques expressando o comportamento de dois controladores projetados para o PO em questão: o DEE-MPC (linha tracejada) e o SDMC (linha sólida).

O SDMC se comporta de maneira semelhante ao DEE-MPC, apresentando características menos oscilatórias e uma satisfatória rejeição a distúrbios.

É importante citar aqui a capacidade do SDMC em controlar um sistema de forma equivalente a um DEE-MPC sintonizado por uma metodologia conceituada na literatura. Para este trabalho, os resultados se focaram na exploração das características únicas associadas a esse MPC, onde aspectos referentes a fatores “clássicos” de ajuste por assim dizer, como horizonte de predição, horizonte de controle, foram deixados de lado. Em trabalhos futuros, é interessante explorar uma metodologia de ajuste que englobe todas essas características, atingindo resultados ainda melhores.

Capítulo 7 – Conclusão e Trabalhos Futuros

Nesta dissertação foi apresentada uma nova ferramenta para simulação de processos dinâmicos lineares que, combinando a teoria de *splines* e a representação de modelos em espaços de estado, vale-se de uma pequena quantidade de nós em um dado horizonte, representando as entradas do sistema, para a simulação das saídas de um processo ao longo do tempo, independentemente do número de variáveis de entrada e de saída.

A SDM foi concebida inspirada na DM, juntamente com a representação em espaços de estado, cujo potencial foi apresentado, corroborando a sua equiparação a essas técnicas já estabelecidas na literatura. A natureza inerente à SDM dá ao usuário a liberdade de representar com grande acurácia qualquer conjunto de entradas que lhe for necessário, desde que os parâmetros associados a essa abordagem sejam corretamente ajustados.

A caracterização e captura da informação dinâmica relevante ao processo é de fundamental importância, para a concepção da SDM de maneira correta. O posicionamento e quantidade de nós afeta diretamente a qualidade das respostas, assim como os parâmetros associadas à construção das *splines* intervalo a intervalo representados por τ , γ e ω .

A sua aplicação, entretanto, não se resume à simulação dinâmica pura e simplesmente. Alicerçado na metodologia SDM, um novo tipo de controlador preditivo baseado em modelo foi proposto, o SDMC. Com a assimilação das características da SDM na predição das saídas ao longo do tempo e na formulação do problema de otimização, novas características foram introduzidas a esse controlador.

Da maneira como foi abordado nesta dissertação, a representação da saída de um processo qualquer ao longo de um horizonte de predição foi realizada a partir da discretização deste horizonte, partindo de funções *piecewise* contínuas, onde os

nós são posicionados em instantes de tempo pré-determinados. Apesar de todos os pontos no horizonte serem considerados para o cálculo da função objetivo, as variáveis de decisão do problema de otimização são somente esses nós, o que limita a variação das ações de controle ótimas. Características de robustez são introduzidas naturalmente a um controlador que, via de regra, apresentaria um desempenho pobre quanto a isso. As respostas tomam características menos oscilatórias e mais estáveis, quando comparado com outros MPCs, como pôde ser visto no capítulo 6. Ainda, em virtude dessas características, o termo relacionado à variação das entradas, *Move Supression*, perde parte de sua significância, podendo ser até mesmo suprimido, visto que o seu efeito pode ser substituído pela disposição dos nós ao longo do horizonte.

Ao longo desta dissertação, o objetivo principal foi a apresentação da metodologia SDM, motivando trabalhos futuros a serem realizados. Inicialmente, em se tratando da aplicação em MPC apresentada nesta dissertação, diversas melhorias na metodologia podem ser empregadas, não somente relacionadas à estruturação do código por si só, mas também à concepção do algoritmo como um todo. A formulação do problema de otimização, por exemplo, apresenta diversas alternativas a serem abordadas. Neste trabalho, referente à matriz dinâmica em particular, um conceito análogo às metodologias já existentes foi proposto, onde a discretização faz-se necessária. Por que não, em vez disso, entretanto, trabalhar com as funções *piecewise* analíticas na íntegra para a concepção da função objetivo, sem haver a necessidade de tal discretização ao longo de um horizonte? Isso traria novas características a serem exploradas pela técnica. Outra possibilidade de melhoria estaria relacionada ao espaçamento da SDM gerada. Apesar dos nós não serem igualmente espaçados, os pontos discretizados o são. Ao avaliar a distribuição variável dos pontos, questões ligadas à representação de sistemas rígidos com dinâmicas muito distintas mostram-se mais tangíveis.

Para sistemas multivariáveis, *soft constraints* e *targets* devem ser integrados nas próximas versões do SDMC. Concomitantemente, é importante a diversificação da aplicação da técnica em processos mais complexos, lidando com sistemas rígidos, sistemas de elevada ordem, tempo morto, multimodelos e a extensão para sistemas variantes no tempo.

Outro ponto importante ligado a essa representação trata do posicionamento dos nós no espaço, o que influencia diretamente na qualidade da simulação obtida. Trabalhar para encontrar uma métrica que extraia somente pontos com características mais relevantes, do ponto de vista dinâmico é de grande importância. Outra possibilidade seria a utilização das técnicas de compressão de dados propostas por Silveira (2012) diretamente como entradas visando a predição das saídas. Isso poderia ser empregado no desenvolvimento de uma metodologia para validação e montagem de modelos.

Seguindo com a ideia de captura de informação dinâmica, identificação e auditoria de modelos também despontam como campos a serem explorados pela metodologia SDM. É uma técnica que apresenta um grande potencial, devido a sua

aplicação intuitiva e conceitualmente de fácil aplicação, onde a combinação de técnicas já existentes, adicionadas com a teoria de *splines*, culmina em uma ferramenta de simulação dinâmica de processos com grande potencial.

Referências

- Aragão, M. E. 2006. Abordagem flexível, robusta e estruturada para o projeto de controladores PID multivariáveis. In *Departamento de Engenharia Química*, 158. Universidade Federal do Rio Grande do Sul.
- Biswal, P. C. 2008. *Numerical Analysis*. Prentice-Hall Of India Pvt. Ltd.
- Boor, C. D. 2001. *A practical guide to splines*. Springer.
- Cagienard, R., P. Grieder, E. C. Kerrigan & M. Morari (2007) Move blocking strategies in receding horizon control. *Journal of Process Control*, 17, 563-570.
- Camacho, E. F. & C. Bordons. 2004. *Model predictive control*. Springer.
- Clarke, D. W., C. Mohtadi & P. S. Tuffs (1987) Generalized predictive control--Part I. The basic algorithm. *Automatica*, 23, 137-148.
- Cutler, C. R. & B. L. Ramaker (1980) DYNAMIC MATRIX CONTROL - A COMPUTER CONTROL ALGORITHM. *Journal of Environmental Science and Health, Part B: Pesticides, Food Contaminants, and Agricultural Wastes*, 1.
- De Vito, D. & R. Scattolini. 2007. A receding horizon approach to the multiobjective control problem. 6029-6034. New Orleans, LA.
- Duraiski, R. G. 2001. Controle preditivo não linear utilizando linearizações ao longo da trajetória (Non-linear predictive control through linearization along the trajectory). In *Chemical Engineering Department*, 289. Porto Alegre: Federal University of Rio Grande do Sul.
- Eberly, D. 1999. Kochanek-Bartels Cubic Splines (TCB Splines). Geometric Tools, LLC.
- Faraway, J. J., M. P. Reed & J. Wang (2007) Modelling three-dimensional trajectories by using Bézier curves with application to hand motion. *Journal of the Royal Statistical Society. Series C: Applied Statistics*, 56, 571-585.
- Farin, G. E. & D. Hansford. 2000. *The essentials of CAGD*. A.K. Peters.
- Foley, J. D. 2006. *Introduction to computer graphics*. Addison-Wesley.
- Gondhalekar, R. & J.-i. Imura (2010) Least-restrictive move-blocking model predictive control. *Automatica*, 46, 1234-1240.
- Huang, B. & R. Kadali. 2008. *Dynamic modeling, predictive control and performance monitoring: a data-driven subspace approach*. Springer.
- Iovine, J. 1995. *Step into virtual reality*. Windcrest/McGraw-Hill.
- Kalman, R. E. (1960a) A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME – Journal of Basic Engineering*, 35-45.
- . 1960b. Contributions to the Theory of Optimal Control.
- Lightbody, G., P. O'Reilly, G. W. Irwin, K. elly & J. McCormick (1997) Neural modelling of chemical plant using MLP and B-spline networks. *Control Engineering Practice*, 5, 1501-1515.
- Liu, T. & F. Gao. 2011. *Industrial Process Identification and Control Design: Step-test and Relay-experiment-based Methods*. Springer.
- Luyben, W. L. 1990. *Process modeling, simulation, and control for chemical engineers*. McGraw-Hill.
- Maciejowski, J. M. 2002. *Predictive control: with constraints*. Prentice Hall.

- Marquis, P. & J. P. Broustail. 1998. SMOC, a bridge between state space and model predictive controllers: Application to the automation of a hydrotreating unit. In *IFAC workshop on model based process control*, 37-43. Oxford.
- Ogata, K. 2003. *Modern control engineering*. Prentice Hall.
- Ogunnaike, B. A. & W. H. Ray. 1994. *Process dynamics, modeling, and control*. Oxford University Press.
- Paim, A. C. 2009. Controle Preditivo Retroalimentado por Estados Estimados, Aplicado a uma Planta Laboratorial (Feedback predictive control using state estimation, applied to a laboratorial plant). In *Chemical Engineering Postgraduate Studies Program*, 161. Porto Alegre: Federal University of Rio Grande do Sul.
- Pakhira, M. K. 2010. *Computer Graphics Multimedia And Animation*. Prentice-Hall Of India Pvt. Ltd.
- Propoi, A. I. (1963) Use of LP methods for synthesizing sampled-data automatic systems. *Automation and Remote Control*, 24, 837-844.
- Pérez, F., B. E. Granger & J. D. Hunter (2011) Python: An ecosystem for scientific computing. *Computing in Science and Engineering*, 13, 13-21.
- Qin, S. J. & T. A. Badgwell (2003) A survey of industrial model predictive control technology. *Control Engineering Practice*, 11, 733-764.
- Richalet, J., A. Rault, J. L. Testud & J. Papon (1978) Model predictive heuristic control. Applications to industrial processes. *Automatica*, 14, 413-428.
- Shikin, E. V. & A. I. Plis. 1995. *Handbook on splines for the user*. CRC Press.
- Silveira, R. 2012. Novas Metodologias Para Compressão de Dados de Processos e Sistemáticas Para Ajuste do Sistema PI. In *Departamento de Engenharia Química*. Porto Alegre: Universidade Federal do Rio Grande do Sul.
- Smith, A. R. 1983. Spline Tutorial Notes - Technical Memo Nº 77. In *SIGGRAPH '83 Tutorial Notes: Introduction to Computer Animation*, 64 - 75.
- Späth, H. 1995. *One dimensional spline interpolation algorithms*. A K Peters.
- Sun, X. & H. Wang. 2005. Dynamic system control of gas jet flames distribution by predictive control method. 1458-1463.
- Tatjewski, P. 2010. *Advanced Control of Industrial Processes: Structures and Algorithms*. Springer.
- Trierweiler, J. O. & S. Engell (1997) The Robust Performance Number: A new tool for control structure design. *Computers & Chemical Engineering*, 21, Supplement, S409-S414.
- Yousfi, C. & R. Tournier. 1991. Steady state optimization inside model predictive control. 1866-1870.
- Zadeh, L. A. & B. H. Whalen (1962) On optimal control and linear programming. *IRE Transactions on Automation Control*, 7, 45-46.
- Åström, K. J. & R. M. Murray. 2008. *Feedback systems: an introduction for scientists and engineers*. Princeton University Press.

Apêndice

Apêndice I - Equacionamento do Algoritmo de Casteljaou

O algoritmo de Casteljaou baseia-se em funções paramétricas de primeira ordem para a obtenção de polinômios de ordem superior. Tais funções de primeira ordem representam retas que, a partir de uma variável paramétrica j , percorrem o caminho existente entre dois pontos no espaço. Tomando como base a figura 3.1, essas retas representam o caminho a ser percorrido entre p_0 e p_1 , assim como entre p_1 e p_2 . Tais funções são representadas, respectivamente, por 7.1 e 7.2:

$$p_{01} = (1 - j)p_0 + jp_1 \quad (7.1)$$

$$p_{11} = (1 - j)p_1 + jp_2 \quad (7.2)$$

Essas equações podem ser encaradas como pontos móveis deslizando sobre duas retas distintas. Promovendo a interpolação entre esses pontos, fica evidente que a representação final é uma função de segunda ordem, dependente de três pontos (p_0, p_1 e p_2), conforme fica explícito em 7.3.

$$p_{02} = (1 - j)p_{01} + jp_{11} = (1 - j)^2p_0 + 2j(1 - j)p_1 + j^2p_2 \quad (7.3)$$

Seguindo a mesma analogia para a obtenção de um polinômio cúbico, promove-se a interpolação entre dois pontos deslizantes p_{02} e p_{12} , originando um polinômio cúbico p_{03} , como pode ser visto em 7.4, após algumas simplificações:

$$p_{03} = (1 - j)^3p_0 + 3j(1 - j)^2p_1 + 3j^2(1 - j)p_2 + j^3p_3 \quad (7.4)$$

É importante salientar que uma regra geral é seguida para a geração dessas funções. Assim como para os outros polinômios obtidos, a existência desses polinômios é condicionada a um número de pontos no espaço igual a sua ordem mais um.

Apêndice II - Matriz $TCBMat$

O cálculo da matriz $TCBMat$ é realizada com base no cálculo das tangentes nos nós expresso em (3.16) e (3.17). O objetivo é criar uma matriz que, quando multiplicada por vetor H composto da entrada dos nós, seja equivalente ao vetor coluna apresentado em (4.1).

Inicialmente, é importante salientar que o cálculo da derivada pela metodologia TCB para um nó qualquer é dependente, além do próprio nó, dos nós adjacentes imediatamente antes e depois do mesmo. A razão para tal é a necessidade do cálculo dos deltas, apresentados na figura 3.5, que influenciam diretamente o cálculo das tangentes. Partindo desse pressuposto, pode-se ver que para os nós inicial e final, o cálculo é um pouco distinto, visto que não há nó anterior e posterior, respectivamente. Assim, a estrutura da $TCBMat$ para esses nós é tal conforme expresso em (7.5) e (7.7).

Em se tratando dos pontos intermediários, os mesmos respeitam uma regra geral, onde a cada intervalo, devido à multiplicação pelo vetor H , os termos não nulos sofrem um deslocamento para a direita. A representação fica clara em (7.6). Tomando como base todas essas equações percebe-se que, independentemente do nó, cada $TCBMat_i$ é dada por uma matriz $4 \times (\text{Número de nós})$. Em se tratando de (7.6), essa premissa pode ser constatada desmembrando a matriz $TCBMat_{2..n-2}$ em matrizes menores, seguindo a lógica supracitada acima.

$$TCBMat_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & 0 & \dots & 0 \\ -\frac{(1-\tau_1)(1-\gamma_1)(1-\omega_1)}{2*\delta_1} & \frac{(1-\tau_1)(1-\gamma_1)(1-\omega_1)}{2*\delta_1} & 0 & 0 & \dots & 0 \\ -\frac{(1-\tau_2)(1-\gamma_2)(1+\omega_2)}{2*\delta_1} & \frac{(1-\tau_2)(1-\gamma_2)(1+\omega_2)}{2*\delta_1} & -\frac{(1-\tau_2)(1+\gamma_2)(1-\omega_2)}{2*\delta_2} & \frac{(1-\tau_2)(1+\gamma_2)(1-\omega_2)}{2*\delta_2} & 0 & \dots & 0 \end{bmatrix} \quad (7.5)$$

$$TCBMat_{n-1} = \begin{bmatrix} 0 & \dots & 0 & 0 & 1 & 0 \\ 0 & \dots & 0 & 0 & 0 & 1 \\ 0 & \dots & 0 & -\frac{(1-\tau_{n-1})(1+\gamma_{n-1})(1+\omega_{n-1})}{2*\delta_{n-2}} & \frac{(1-\tau_{n-1})(1+\gamma_{n-1})(1+\omega_{n-1})}{2*\delta_{n-2}} & -\frac{(1-\tau_{n-1})(1-\gamma_{n-1})(1-\omega_{n-1})}{2*\delta_{n-1}} & \frac{(1-\tau_{n-1})(1-\gamma_{n-1})(1-\omega_{n-1})}{2*\delta_{n-1}} \\ 0 & \dots & 0 & 0 & -\frac{(1-\tau_n)(1-\gamma_n)(1+\omega_n)}{2*\delta_{n-1}} & \frac{(1-\tau_n)(1-\gamma_n)(1+\omega_n)}{2*\delta_{n-1}} \end{bmatrix} \quad (7.6)$$

$$TCBMat_{2..n-2} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & -\frac{(1-\tau_2)(1+\gamma_2)(1+\omega_2)}{2*\delta_1} & \frac{(1-\tau_2)(1+\gamma_2)(1+\omega_2)}{2*\delta_1} & -\frac{(1-\tau_2)(1-\gamma_2)(1-\omega_2)}{2*\delta_2} & \frac{(1-\tau_2)(1-\gamma_2)(1-\omega_2)}{2*\delta_2} & 0 & \dots & 0 \\ 0 & -\frac{(1-\tau_3)(1-\gamma_3)(1+\omega_3)}{2*\delta_2} & \frac{(1-\tau_3)(1-\gamma_3)(1+\omega_3)}{2*\delta_2} & -\frac{(1-\tau_3)(1+\gamma_3)(1-\omega_3)}{2*\delta_3} & \frac{(1-\tau_3)(1+\gamma_3)(1-\omega_3)}{2*\delta_3} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & \dots & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & \dots & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & \dots & 0 & -\frac{(1-\tau_{n-2})(1+\gamma_{n-2})(1+\omega_{n-2})}{2*\delta_{n-3}} & \frac{(1-\tau_{n-2})(1+\gamma_{n-2})(1+\omega_{n-2})}{2*\delta_{n-3}} & -\frac{(1-\tau_{n-2})(1-\gamma_{n-2})(1-\omega_{n-2})}{2*\delta_{n-2}} & \frac{(1-\tau_{n-2})(1-\gamma_{n-2})(1-\omega_{n-2})}{2*\delta_{n-2}} & 0 \\ 0 & \dots & 0 & -\frac{(1-\tau_{n-1})(1-\gamma_{n-1})(1+\omega_{n-1})}{2*\delta_{n-2}} & \frac{(1-\tau_{n-1})(1-\gamma_{n-1})(1+\omega_{n-1})}{2*\delta_{n-2}} & -\frac{(1-\tau_{n-1})(1+\gamma_{n-1})(1-\omega_{n-1})}{2*\delta_{n-1}} & \frac{(1-\tau_{n-1})(1+\gamma_{n-1})(1-\omega_{n-1})}{2*\delta_{n-1}} & 0 \end{bmatrix} \quad (7.7)$$