

BERNARDO GIORI AMBRÓSIO

**MODELAGEM DA FASE DE REQUISITOS EM PROCESSOS DE
DESENVOLVIMENTO DE SOFTWARE: UMA ABORDAGEM
UTILIZANDO DINÂMICA DE SISTEMAS**

Dissertação apresentada à
Universidade Federal de Viçosa, como
parte das exigências do Programa de
Pós-Graduação em Ciência da
Computação, para obtenção do título de
Magister Scientiae.

**VIÇOSA
MINAS GERAIS - BRASIL
2008**

**Ficha catalográfica preparada pela Seção de Catalogação e
Classificação da Biblioteca Central da UFV**

T

A496m
2008

Ambrósio, Bernardo Giori, 1984-

Modelagem da fase de requisitos em processos de desenvolvimento de software: uma abordagem utilizando dinâmica de sistemas / Bernardo Giori Ambrósio. – Viçosa, MG, 2008.

xiv, 155f. : il. (algumas col.) ; 29cm.

Inclui apêndice.

Orientador: José Luis Braga.

Dissertação (mestrado) - Universidade Federal de Viçosa.

Referências bibliográficas: f. 146-155.

1. Engenharia de software - Modelos matemáticos.
2. Software - Desenvolvimento. 3. Avaliação de riscos.
4. Dinâmica. I. Universidade Federal de Viçosa. II. Título.

CDD 22.ed. 005.1

BERNARDO GIORI AMBROSIO

**MODELAGEM DA FASE DE REQUISITOS EM PROCESSOS DE
DESENVOLVIMENTO DE SOFTWARE: UMA ABORDAGEM
UTILIZANDO DINÂMICA DE SISTEMAS.**

Dissertação apresentada à
Universidade Federal de Viçosa, como
parte das exigências do Programa de
Pós-Graduação em Ciência da
Computação, para obtenção do título
de *Magister Scientiae*.

APROVADA: 10 de abril de 2008

Prof. Alcione de Paiva Oliveira
(Co-orientador)

Prof. Moisés de Andrade Resende Filho
(Co-orientador)

Prof. Mauro Nacif Rocha

Prof. Aziz Galvão da Silva Júnior

Prof. José Luis Braga
(Orientador)

*Dedico essa dissertação aos meus pais
Sinval e Fátima*

*Ao meu irmão
Iago*

*E à minha amada noiva
Nayarah*

AGRADECIMENTOS

Em primeiro lugar, agradeço a Deus que me deu força e coragem para realizar esse trabalho. Agradeço por sempre colocar pessoas certas em meu caminho, que sempre me auxiliaram, não só nesse trabalho, como em toda a vida.

Agradeço também aos meus pais, Sinval e Fátima, e ao meu irmão Iago, por todo amor e incentivo concedido ao longo de todo esse tempo. Agradeço por sempre acreditarem em mim e pelo apoio em todos os momentos; sem vocês, nada disso seria possível. Aos meus pais, obrigado por toda a educação que recebi e pela oportunidade que me deram de realizar meus sonhos. Vocês são exemplos de vida para mim. Agradeço também a todos os familiares: avós, tios e primos, pelo grande incentivo.

Agradeço à Nayah, ‘minha Nay, meu amor’, por todos os momentos divididos, pelo companheirismo, pela amizade, pela paciência, pelo amor e pelo carinho. Você também faz parte dessa conquista. Obrigado por tudo que sempre fez e faz por mim e por nós. Um enorme obrigado também ao José Antônio e à Vicência, que sempre me acolheram em sua casa. Vocês são minha segunda família.

Agradeço ao Professor José Luis Braga, por acreditar em mim, tanto na graduação quanto no mestrado. Você, além de orientador, é também um amigo que sempre me mostrou a direção correta dos passos a serem tomados. Agradeço também aos meus co-orientadores, Professores Alcione de Paiva Oliveira e Moisés de Andrade Resende Filho, pelos conselhos e opiniões sobre o trabalho.

A todos do corpo docente do Departamento de Informática, da Universidade Federal de Viçosa, pela formação e conhecimento recebidos. Ensinos não só para o crescimento profissional, como para a vida toda. Muito obrigado.

Ao Altino Alves de Souza Filho, secretário da Pós-graduação, pelas ótimas conversas e por sempre se prontificar a ajudar perante as burocracias do mestrado. À Eliana Ferreira Rocha, secretária da Graduação, por também ajudar diante de eventuais problemas.

A todos os colegas, do mestrado e da graduação, que me incentivaram, ajudaram e contribuíram direta ou indiretamente para a realização desse trabalho. Um agradecimento especial ao Pedro e ao Estevão, companheiros de trabalho e de estudo.

Ao pessoal da república, pelos vários anos de convivência e de amizade. Um agradecimento especial à Maria Júlia, ‘Dona Maria’, secretária da república, que me acompanhou durante a graduação e o mestrado. Na correria do dia-a-dia, sempre declarava palavras e conselhos que acalmavam os ânimos.

A Universidade Federal de Viçosa.

BIOGRAFIA

BERNARDO GIORI AMBRÓSIO, filho de Sinval Ambrósio e Nadir de Fátima Giori, brasileiro nascido em 15 de outubro de 1984 na cidade de Cachoeiro de Itapemirim, no estado do Espírito Santo.

No ano de 2002, após concluir o ensino médio na cidade de Cachoeiro de Itapemirim, ingressou no curso de graduação em Ciência da Computação na Universidade Federal de Viçosa, concluído no ano de 2005.

Em 2006, foi aprovado na seleção do programa de pós-graduação do Departamento de Informática – DPI, onde, em maio de 2006, iniciou o mestrado em Ciência da Computação na Universidade Federal de Viçosa – UFV, defendendo sua dissertação em abril de 2008.

Trabalhou como Analista de Sistemas na Atan Ciência da Informação de janeiro de 2006 a agosto de 2007. Atualmente, é docente na Faculdade Ubaense Ozanam Coelho – FAGOC, desde março de 2007.

SUMÁRIO

LISTA DE FIGURAS	viii
LISTA DE TABELAS	x
LISTA DE QUADROS	xi
RESUMO	xii
ABSTRACT	xiv
1 INTRODUÇÃO	1
1.1 Motivação.....	3
1.2 Objetivos	5
1.3 O fluxo de trabalho utilizado no desenvolvimento dessa pesquisa.....	6
1.4 Organização desse documento	7
2 REFERENCIAL TEÓRICO.....	9
2.1 Requisitos de sistemas de software	9
2.2 Dinâmica de sistemas.....	12
2.3 Análise e pensamento sistêmicos no gerenciamento de projetos de software	16
2.4 Trabalhos correlatos	18
3 CONSTRUÇÃO DE UM MODELO DINÂMICO PARA A FASE DE REQUISITOS.....	24
3.1 O processo de construção do modelo.....	24
3.2 Relação entre as medidas pontos de função e homens-dia	27
3.3 Visões para análise do modelo	28
3.4 Elementos do Modelo Nível 2.....	29
3.5 Elementos do Modelo Nível 1.....	33
3.5.1 Estado dos requisitos a serem entregues na próxima liberação	33
3.5.2 Alocação de esforço para as atividades.....	35
3.5.3 Requisitos entregues em liberações anteriores.....	42
3.5.4 Quantidade de esforço necessário e definição do tamanho da equipe.....	46
3.5.5 Controle do tamanho da equipe.....	49
3.5.6 Uso de esforço extra da equipe e ajustes no prazo.....	52
3.5.7 Quantidade de esforço disponibilizado pela equipe.....	56
3.5.8 Geração de erros.....	59
3.5.9 Custo para a realização da especificação dos requisitos	60
4 USO DO MODELO PARA APOIO À TOMADA DE DECISÃO	62
4.1 Painel de controle	62

4.1.1	Variáveis de configuração.....	62
4.1.2	Variáveis de análise.....	64
4.2	Variáveis-gráfico.....	68
4.3	Modelagem de cenários.....	71
4.4	Análises gerenciais suportadas pelo modelo.....	74
5	SIMULAÇÕES DO MODELO.....	78
5.1	Ajustando o modelo com dados disponibilizados na literatura.....	78
5.1.1	Definição e simulação de um cenário base	78
5.1.2	Aumento na quantidade de erros cometidos	86
5.1.3	Aumento na volatilidade dos requisitos	89
5.1.4	<i>Turnover</i> de pessoal	91
5.1.5	Comparação entre cenários: otimista, mais provável e pessimista	95
5.1.6	Erro na estimativa da quantidade de requisitos.....	99
5.1.7	Modificações no tamanho da equipe.....	102
5.1.8	Alocação de esforço para as atividades de garantia de qualidade....	104
5.1.9	Uso de esforço extra.....	106
5.2	Ajustando o modelo com dados coletados em uma empresa de desenvolvimento de software.....	108
5.2.1	Definição e simulação de um cenário base	110
5.2.2	Aumento na quantidade de erros cometidos	115
5.2.3	Aumento na volatilidade dos requisitos	117
5.2.4	<i>Turnover</i> de pessoal	118
5.2.5	Comparação entre cenários: otimista, mais provável e pessimista ..	123
5.2.6	Erro na estimativa da quantidade de requisitos.....	125
5.2.7	Modificações no tamanho da equipe.....	128
5.2.8	Alocação de esforço para as atividades de garantia de qualidade....	130
5.2.9	Uso de esforço extra.....	132
5.3	Considerações sobre as simulações.....	134
6	CONCLUSÕES	136
6.1	Trabalhos futuros	138
	APÊNDICE A.....	140
A.1	Glossário com a definição dos elementos do Modelo Nível 1	140
	REFERÊNCIAS BIBLIOGRÁFICAS	146

LISTA DE FIGURAS

Figura 2.1. Diagrama de influência.....	13
Figura 2.2. Símbolos dos elementos contidos em um modelo estoque-fluxo da Dinâmica de Sistemas	15
Figura 2.3. Modelo estoque-fluxo da Dinâmica de Sistemas.....	16
Figura 2.4. Relacionamentos entre variáveis envolvidas no gerenciamento de projetos de software	17
Figura 2.5. Diagrama de influência que descreve a estrutura do modelo dinâmico para o gerenciamento de projetos de software	20
Figura 3.1. Diagrama de atividades para a construção do modelo.....	25
Figura 3.2. Relação entre tamanho do trabalho e esforço necessário	28
Figura 3.3. Visões para estudo e análise do modelo	29
Figura 3.4. Estrutura do Modelo Nível 2	31
Figura 3.5. Estados dos requisitos.....	34
Figura 3.6. Alocação de esforço para as atividades	36
Figura 3.7. Cálculo do valor da variável <i>densidade de erros</i>	37
Figura 3.8. Relação entre a densidade de erros e o potencial de detecção de erros da equipe	38
Figura 3.9. Correção de erros	39
Figura 3.10. Alteração de requisitos	40
Figura 3.11. Estoque <i>Requisitos Re-especificados</i>	40
Figura 3.12. Especificação de novos requisitos	41
Figura 3.13. Fluxos que representam os erros cometidos	41
Figura 3.14. Estados dos requisitos entregues em liberações anteriores.....	43
Figura 3.15. Cálculo do valor da variável <i>densidade de erros nos requisitos entregues</i>	45
Figura 3.16. Relação entre a densidade de erros nos requisitos entregues e a taxa com que os erros são descobertos pelos usuários	45
Figura 3.17. Quantidade de esforço necessário e definição do tamanho da equipe... 46	
Figura 3.18. Controle do tamanho da equipe	50
Figura 3.19. Uso de esforço extra da equipe e ajustes no prazo	53
Figura 3.20. Cálculo do valor da variável <i>tempo necessário para terminar</i>	54
Figura 3.21. Exaustão da equipe devido à sobrecarga de trabalho	55
Figura 3.22. Quantidade de esforço disponibilizado por um membro experiente	57
Figura 3.23. Relação entre a produtividade dos novatos e a dos experientes	58
Figura 3.24. Cálculo do valor da variável <i>total de homens-dia disponíveis diariamente</i>	59
Figura 3.25. Geração de erros	60
Figura 3.26. Custo para a realização da especificação dos requisitos.....	60
Figura 4.1. Relacionamentos envolvendo a variável-gráfico <i>relação entre esforço extra utilizado e quantidade de erros cometidos</i>	70
Figura 4.2. Esboço do gráfico da variável <i>relação entre esforço extra utilizado e quantidade de erros cometidos</i>	71

Figura 4.3. Aumento na produtividade devido ao esforço extra e queda na produtividade devido à fadiga e estresse da equipe	76
Figura 5.1. Painel de controle – entrada de valores	81
Figura 5.2. Painel de controle – saída de valores	82
Figura 5.3. Qualidade e custo obtidos com o cenário base	84
Figura 5.4. Aumento na quantidade de erros cometidos	87
Figura 5.5. Aumento na volatilidade dos requisitos	90
Figura 5.6. Impactos da ocorrência de <i>turnover</i>	92
Figura 5.7. Efeitos de manter uma pessoa de reserva na equipe	94
Figura 5.8. Comparação entre cenários	96
Figura 5.9. Tendência das curvas de custo	98
Figura 5.10. Impactos do erro na estimativa da quantidade de requisitos	100
Figura 5.11. Impactos de modificações no tamanho da equipe	103
Figura 5.12. Variação na quantidade de esforço alocado para as atividades de garantia de qualidade	106
Figura 5.13. Variação na quantidade de esforço extra utilizado	108
Figura 5.14. Painel de controle – entrada de valores	111
Figura 5.15. Painel de controle – saída de valores	112
Figura 5.16. Qualidade e custo obtidos com o cenário base	113
Figura 5.17. Aumento na quantidade de erros cometidos	116
Figura 5.18. Aumento na volatilidade dos requisitos	118
Figura 5.19. Impactos da ocorrência de <i>turnover</i> quando o tempo decorrido do início da especificação for igual à metade do prazo planejado	119
Figura 5.20. Impactos da ocorrência de <i>turnover</i> quando o tempo decorrido do início da especificação for igual a 40% do prazo planejado	121
Figura 5.21. Efeitos de manter uma pessoa de reserva na equipe	122
Figura 5.22. Comparação entre cenários	124
Figura 5.23. Impactos do erro na estimativa da quantidade de requisitos	126
Figura 5.24. Impactos do erro na estimativa da quantidade de requisitos	127
Figura 5.25. Impactos de modificações no tamanho da equipe	129
Figura 5.26. Variação na quantidade de esforço alocado para as atividades de garantia de qualidade	131
Figura 5.27. Variação na quantidade de esforço extra utilizado	133

LISTA DE TABELAS

Tabela 3.1. Nível de competência dos profissionais	49
Tabela 4.1. Intervalos de valores para algumas variáveis de análise	72
Tabela 5.1. Valores utilizados para configurar o cenário base	79
Tabela 5.2. Tempo gasto para terminar a especificação em função do prazo planejado	101
Tabela 5.3. Questionário para coleta de dados acerca da fase de requisitos	109
Tabela 5.4. Tempo gasto para terminar a especificação em função do prazo planejado	114
Tabela 5.5. Tempo gasto para terminar a especificação em função do prazo planejado	128

LISTA DE QUADROS

Quadro 4.1. Variáveis de configuração.....	63
Quadro 4.2. Variáveis de análise para simular a ocorrência de riscos.....	65
Quadro 4.3. Variáveis de análise para determinar as decisões e as políticas gerenciais	66
Quadro 4.4. Variáveis de análise para verificação após a simulação.....	68
Quadro 4.5. Variáveis-gráfico do modelo.....	69

RESUMO

AMBRÓSIO, Bernardo Giori, M.Sc., Universidade Federal de Viçosa, abril de 2008.

Modelagem da fase de requisitos em processos de desenvolvimento de software: uma abordagem utilizando dinâmica de sistemas. Orientador: José Luis Braga. Co-Orientadores: Alcione de Paiva Oliveira e Moisés de Andrade Resende Filho.

Os problemas de decisão na área de gerenciamento de projetos de software são complexos, com muitas variáveis dinâmicas interconectadas que se relacionam de maneira não linear. Diante da quantidade de variáveis e relacionamentos envolvidos nesse processo, torna-se muito difícil, senão impossível, para os gerentes entenderem e reproduzirem mentalmente o comportamento do processo, sendo indispensável a utilização de ferramentas que permitam enxergar todo o contexto. A dinâmica de sistemas é uma dessas ferramentas, consistindo em uma técnica de modelagem que permite a análise, compreensão e simulação de problemas e situações que envolvam comportamento dinâmico. Alguns trabalhos prévios foram realizados com o objetivo de definir um modelo de dinâmica de sistemas para descrever o processo de gerenciamento de projetos e tentar aumentar a compreensão sobre esse processo. Entretanto, esses trabalhos não abordaram a modelagem das variáveis e dos relacionamentos envolvidos na fase de requisitos. O objetivo desse trabalho é construir um modelo de dinâmica de sistemas que abrange as principais variáveis envolvidas na fase de requisitos de projetos de software e descreve como essas variáveis se relacionam umas com as outras. As simulações realizadas com o modelo permitiram verificar e antever os impactos da materialização de riscos, tais como *turnover* de pessoal, alta volatilidade dos requisitos e outros. Também foi possível antecipar, utilizando a simulação, os possíveis efeitos das decisões gerenciais planejadas antes que elas sejam implementadas em um projeto real. O modelo pode ser utilizado como uma ferramenta para apoiar os gerentes na tomada de decisão durante a fase de requisitos. Para configurar o modelo e realizar as simulações, foi definido um painel de controle contendo os parâmetros que são utilizados para

ajustar o modelo de acordo com o cenário que se pretende simular. São apresentados vários cenários simulados com o modelo. A análise e a discussão dos resultados obtidos com as simulações mostram que o comportamento do modelo está de acordo com o conhecimento comum da área de Engenharia de Software. Isso comprova a consistência do modelo e a viabilidade do seu uso no apoio à tomada de decisão.

ABSTRACT

AMBRÓSIO, Bernardo Giori, M.Sc., Universidade Federal de Viçosa, April, 2008.

Modeling of the requirements phase of software development lifecycles: an approach using system dynamics. Adviser: José Luis Braga. Co-Advisers: Alcione de Paiva Oliveira and Moisés de Andrade Resende Filho.

Decision problems in software project management are complex, with many interconnected dynamic variables related in a non-linear way. Given the great number of variables and their relationships involved in this process, it becomes very difficult, if not impossible, for managers to understand and manage the whole process without the help of tools that allow them to see the forest for the trees. System Dynamics is one of these tools, providing a modeling technique that allows the analysis, understanding and simulation of problems and situations with dynamic behaviour. Some previous studies have been conducted to define a system dynamics model for describing parts of software project management problems, aiming at understanding the problems involved. However, these studies haven't addressed the modeling of variables and their relationships involved in the requirements phase of software development lifecycles. The goal of this research was to build a system dynamics model focusing on the key variables involved in the requirements phase of software development lifecycles, establishing their mutual influence. The simulations carried on using the model let us foresee the consequences of risks materialization, such as people turnover, high requirements volatility and others. It was also possible to anticipate, using the simulation, the effects of decision making before implementing the decisions in the real world. The model may be used as a tool for supporting managers in decision making during the requirements phase. For setting up the model and running simulations, we built a control panel that contains the parameters used for adjusting the model variables according to the scenarios to be used in the simulations. The analysis and discussion of results obtained from the simulations let us conclude that the models are consistent with reality, and so they can be used in actual project management decision making.

1 INTRODUÇÃO

O avanço tecnológico e científico e a queda no preço do hardware vêm contribuindo para a popularização dos computadores e, conseqüentemente, para a disseminação do uso de sistemas de software. Os softwares possuem objetivos variados e atuam em diversas áreas: aplicativos pessoais; automação comercial, industrial e bancária; controle de equipamentos médicos; controle de aviões e do tráfego aéreo; controle de armas e equipamentos militares; etc. Diante da imensa gama de aplicações, a ocorrência de falhas durante a execução do software pode provocar desde uma simples perda de tempo, até grandes prejuízos financeiros e a perda de vidas humanas. Por esse motivo, é imprescindível que a qualidade seja uma característica dos sistemas do software.

Com a evolução do hardware no final da década de 60, tornou-se possível a construção de sistemas de software maiores e mais complexos que os precedentes (SOMMERVILLE, 2000). Os sistemas construídos nesse período possuíam a característica de não serem confiáveis, serem entregues com atraso e apresentarem um custo maior que o previsto inicialmente. Essas dificuldades demonstraram a necessidade de se utilizar um método disciplinado para o desenvolvimento de software (SOMMERVILLE, 2000).

Como um esforço para trazer disciplina para a atividade de desenvolvimento de software, muitas organizações, na década de 70, começaram a aplicar os rigores da ciência e da engenharia no processo de desenvolvimento de software, dando origem à disciplina de Engenharia de Software (ABDEL-HAMID; MADNICK, 1991). Sommerville (2000) define a Engenharia de Software como: “uma disciplina da engenharia que se ocupa de todos os aspectos da produção de software, desde os estágios iniciais de especificação do sistema até a manutenção desse sistema, depois que ele entrou em operação”.

Com a Engenharia de Software, surgiram novos métodos e técnicas que foram aplicadas pela indústria, tais como: projeto e programação estruturada, verificação formal, linguagens amigáveis, programação orientada a objetos, reutilização e outras

que permitiram aumentar a produtividade. No entanto, em parte devido ao enorme número de problemas com a produção de software de qualidade, pouca atenção foi dada pela comunidade científica aos aspectos gerenciais do processo de desenvolvimento de software (ABDEL-HAMID; MADNICK, 1991), que ainda constituem um enorme desafio. A carência de investimentos em pesquisa para aprofundar o conhecimento nessa área é considerada por alguns pesquisadores a principal causa do fracasso de projetos e a responsável pela persistente dificuldade em produzir software com qualidade (KAPPELMAN; MCKEEMAN; ZHANG, 2006; KING; BURGESS, 2006). Segundo (HOOD; HOOD, 2006), a disciplina trazida pela Engenharia de Software não é suficiente para lidar com os desafios gerenciais.

Os problemas de decisão na área de gerenciamento de projetos de software são complexos, com muitas variáveis dinâmicas interconectadas. A análise de decisões e de seus impactos sistêmicos escapa dos limites humanamente gerenciáveis, sendo indispensável a utilização de ferramentas que permitam enxergar todo o contexto. A dinâmica de sistemas (STERMAN, 1992; STERMAN, 2000; BRAGA *et al.*, 2004) é uma dessas ferramentas, consistindo em uma técnica de modelagem que permite a análise, compreensão e simulação de problemas e situações que envolvam comportamento dinâmico.

Alguns trabalhos, como o descrito em (ABDEL-HAMID; MADNICK, 1989) e (ABDEL-HAMID; MADNICK, 1991), foram desenvolvidos com o objetivo de definir um modelo de dinâmica de sistemas para descrever o processo de gerenciamento de projetos de software e tentar aumentar a compreensão sistêmica sobre esse processo. Entretanto, esses trabalhos não incluíram a fase de requisitos em seu escopo, o que contribui para que continuem existindo dificuldades para entender o comportamento dinâmico de algumas variáveis envolvidas na fase de requisitos.

Os requisitos estabelecem as funcionalidades e a abrangência do sistema, definem restrições sobre sua operação e implementação, e representam as expectativas dos usuários quanto ao sistema a ser desenvolvido (SOMMERVILLE, 2000). Uma especificação e uma gestão de requisitos bem planejadas e executadas permitem: alcançar estimativas aceitáveis de tempo e de custo para o projeto, acompanhar o seu progresso, descobrir, o quanto antes, os erros de requisitos, e minimizar os impactos dos reparos e das alterações de requisitos que podem ser necessários ao longo do desenvolvimento.

De acordo com os modelos de maturidade CMMI (*Capability Maturity Model Integration*) (SEI, 2008) e mpsBR (Melhoria do Processo do Software Brasileiro) (SOFTEX, 2007), ser capaz de gerenciar os requisitos corretamente é o primeiro passo para garantir qualidade e melhorar a competência em desenvolvimento de software. A maneira como os requisitos são levantados e gerenciados influencia diretamente no sucesso de um projeto de software e determina a qualidade dos sistemas entregues ao cliente (NUSEIBEH; EASTERBROOK, 2000).

Apesar da importância do trabalho realizado durante a fase de requisitos, a comunidade científica e gerentes de projetos enfrentam um problema: há uma falta de conhecimento acerca do comportamento dinâmico de algumas variáveis envolvidas e dos relacionamentos existentes entre essas variáveis.

1.1 Motivação

A falta de entendimento acerca das interações dinâmicas entre as variáveis envolvidas na fase de requisitos pode levar os gerentes a tomarem decisões reativas, considerando apenas o problema presente, sem relacioná-lo com o seu ambiente, suas variáveis e demais problemas correlacionados. As decisões reativas, em geral, são apenas paliativas e podem solucionar o problema em curto prazo, no entanto, podem agravá-lo em longo prazo.

As decisões equivocadas e as falhas cometidas durante a especificação e gestão dos requisitos podem acarretar inúmeros prejuízos para os projetos de software. Os requisitos formam uma *baseline*¹ que irá direcionar todas as fases seguintes do desenvolvimento. Um erro de requisitos ou uma alteração de requisitos em um estágio avançado do desenvolvimento irá exigir que vários artefatos sejam modificados para que a consistência seja mantida (SAYÃO; LEITE, 2006). Esse retrabalho pode ser responsável por atrasos no projeto e pela introdução de novas falhas. O problema é agravado se os requisitos não forem rastreáveis (RAMESH; JARKE, 2001), pois será mais trabalhoso identificar e obter todos os artefatos que precisam ser alterados.

Em muitas organizações, o processo de requisitos não é realizado com o esforço e com o empenho necessários, e os recursos são concentrados na codificação do sistema, que geralmente é iniciada precocemente. Jones (1996b) relata que o

¹ baseline: conjunto de artefatos associados a um marco do projeto, consistentes entre si e conformes com os padrões do processo (FILHO, 2003).

processo de requisitos esteve deficiente em mais de 75% das empresas de software. A falta de dedicação à fase de requisitos contribui para que erros de requisitos persistam ao longo do processo de desenvolvimento. Segundo (BLACKBURN; BUSSER; NAUMAN, 2001), defeitos em requisitos são responsáveis por 50% das falhas encontradas na fase de testes.

A partir de um estudo realizado, KAPPELMAN; MCKEEMAN; ZHANG (2006) afirmam que o trabalho deficiente realizado durante a fase de requisitos é uma característica de grande parte dos projetos que fracassam. DAMIAN; CHISAN (2006) relatam que estudos realizados pelo *Standish Group*¹, em 2003, descobriram que 28% dos projetos de software foram completamente cancelados. De acordo com esse estudo, os problemas relacionados com requisitos foram as principais causas do fracasso dos projetos.

Apesar da disponibilidade de inúmeras ferramentas e técnicas para apoiar o trabalho realizado durante a fase de requisitos e a posterior gestão dos requisitos levantados, estas atividades continuam não sendo executadas de modo efetivo, apresentando falhas e comprometendo a qualidade do processo e do software desenvolvido. Isso reforça a hipótese de que esse problema não ocorre por motivos técnicos ou tecnológicos.

A falta de entendimento acerca da maneira como as variáveis envolvidas interagem entre si pode ser um fator determinante para a baixa qualidade do trabalho realizado durante a fase de requisitos. A impossibilidade de antever e compreender como as decisões gerenciais afetam o comportamento do sistema dinâmico, formado por essas variáveis, contribui para a tomada de decisões reativas, sem analisar seus impactos sobre as demais variáveis correlacionadas.

Para suprir essa deficiência, há uma necessidade de se estudar o comportamento dinâmico dos requisitos e de algumas variáveis envolvidas na fase de requisitos, e disponibilizar um modelo descritivo que permita a análise e a compreensão desse comportamento. O modelo deve apresentar as variáveis envolvidas e mostrar como essas variáveis se relacionam umas com as outras, permitindo aos gerentes verificar os impactos de determinadas políticas gerenciais antes de implementá-las em um projeto real.

¹ *Standish Group*, <http://www.standishgroup.com>

1.2 Objetivos

O objetivo do trabalho é estudar, compreender e descrever o comportamento dinâmico dos requisitos e as interações dinâmicas entre as principais variáveis envolvidas na fase de requisitos em processos de desenvolvimento de software.

Os objetivos específicos desse trabalho são:

- Identificar as principais variáveis envolvidas na fase de requisitos;
- Investigar como as variáveis interagem entre si ao longo do tempo, produzindo um comportamento dinâmico;
- Aumentar a compreensão acerca dos aspectos gerenciais envolvidos na fase de requisitos;
- Compreender o comportamento dinâmico dos requisitos e de algumas variáveis envolvidas na fase de requisitos;
- Construir um modelo descritivo para a fase de requisitos em processos de desenvolvimento de software, usando dinâmica de sistemas;
- Usar o modelo como um veículo de experimentação para avaliar os impactos da materialização de riscos e analisar os efeitos das decisões gerenciais antes que elas sejam implementadas em um projeto real;
- Tornar possível o uso do modelo como uma ferramenta para apoiar os gerentes na tomada de decisão durante o gerenciamento das atividades realizadas no decorrer da fase de requisitos;
- Apresentar situações que comprovem: a consistência do modelo, a viabilidade do seu uso no apoio à tomada de decisão e a sua capacidade de reproduzir o comportamento dinâmico gerado pelas interações entre as variáveis envolvidas na fase de requisitos.

O quinto desafio daqueles propostos pela SBC - Sociedade Brasileira de Computação, identificado no seminário “Grandes Desafios da Pesquisa em Computação no Brasil – 2006 – 2016” (SBC, 2006), é o “Desenvolvimento tecnológico de qualidade: sistemas disponíveis, corretos, seguros, escaláveis, persistentes e ubíquos”. O aumento do conhecimento sobre as variáveis envolvidas na fase de requisitos contribui para a resolução do quinto desafio, no que tange ao desenvolvimento de sistemas corretos e seguros.

De acordo com a descrição do estado da arte das pesquisas em Engenharia de Requisitos apresentado em (CHENG; ATLEE, 2007), esse trabalho enquadra-se na

atividade de Gerência de Requisitos, apoiando o gerenciamento das atividades realizadas durante a fase de requisitos. Cheng; Atlee (2007) destacam também algumas tendências para as pesquisas futuras na área, concentrando-se nos aspectos técnicos e tecnológicos. O único aspecto gerencial abordado trata da globalização do desenvolvimento de software, em que as equipes de desenvolvimento estão distribuídas geograficamente. Os modelos de dinâmica de sistemas aqui apresentados podem ser expandidos para apoiar o mesmo tipo de decisão no gerenciamento do desenvolvimento distribuído de software.

1.3 O fluxo de trabalho utilizado no desenvolvimento dessa pesquisa

Desde os estágios iniciais da pesquisa, buscou-se realizar um trabalho fortemente embasado em informações comprovadas e publicadas na literatura. Isso garante uma maior confiabilidade aos resultados alcançados e às conclusões obtidas, e também torna explícito o contexto em que o trabalho está inserido e o problema que o mesmo pretende resolver.

Visando a alcançar os resultados que satisfaçam os objetivos do trabalho, foram realizadas as atividades relatadas a seguir.

1. Estudar e entender a modelagem de processos utilizando dinâmica de sistemas: foram obtidos o conhecimento e os fundamentos necessários para aplicar essa técnica na modelagem da fase de requisitos em processos de desenvolvimento de software.

2. Investigar trabalhos relacionados: inicialmente foi realizada a leitura do livro “*Software Project Dynamics: an Integrated Approach*” (ABDEL-HAMID; MADNICK, 1991) que apresenta a construção, a descrição e a avaliação de um modelo de dinâmica de sistemas para o processo de gerenciamento de projetos de software. Além do livro, foram investigados também diversos artigos científicos que abordam assuntos relacionados ao trabalho desenvolvido. Esses artigos são referenciados ao longo do texto.

3. Construir um modelo de dinâmica de sistemas (modelo estoque-fluxo) para a fase de requisitos em processos de desenvolvimento de software: para construir o modelo foi utilizado um processo que será descrito na seção 3.1, constituído pelas seguintes atividades:

3.1. Identificar as variáveis envolvidas na fase de requisitos.

- 3.2. Definir as variáveis principais que são investigadas e incluídas no modelo.
- 3.3. Identificar os relacionamentos existentes entre as variáveis principais.
- 3.4. A partir das variáveis e dos relacionamentos identificados, construir o modelo de dinâmica de sistemas propriamente dito.
- 3.5. Simular e refinar o modelo para verificar se há inconsistências e realizar as correções necessárias.
- 3.6. Quantificar os relacionamentos entre os elementos presentes no modelo.
- 3.7. Validar o modelo para assegurar que os resultados produzidos estão coerentes com o conhecimento comum da área de Engenharia de Software.

4. Possibilitar o uso do modelo como uma ferramenta gerencial: foi definido um painel de controle contendo os parâmetros do modelo que devem ser configurados antes de realizar as simulações. O painel de controle possibilita aos gerentes ajustar o modelo de acordo com o cenário que se pretende simular, permitindo que os resultados das simulações sejam utilizados para verificar os impactos da materialização de riscos e os efeitos das decisões gerenciais planejadas.

5. Verificar a aplicabilidade do modelo: foram elaborados cenários para verificar a aplicabilidade do modelo e a viabilidade do seu uso como uma ferramenta de apoio à tomada de decisão no decorrer da fase de requisitos.

1.4 Organização desse documento

Os resultados alcançados nesse trabalho estão fortemente embasados em dados e informações publicadas na literatura. Para isso, foi realizada uma extensa pesquisa bibliográfica buscando trabalhos científicos que estivessem inseridos no contexto do problema enunciado e da solução proposta nesse trabalho. O Capítulo 2 apresenta o referencial teórico sobre requisitos de sistemas de software e dinâmica de sistemas, e descreve a aplicação do pensamento sistêmico no gerenciamento de projetos de software. São apresentados também trabalhos correlatos a este, que relatam a aplicação da dinâmica de sistemas na modelagem do processo de desenvolvimento e de gerenciamento de projetos de software.

O Capítulo 3 descreve o processo utilizado para construir o modelo de dinâmica de sistemas e explica como as medidas pontos de função e homens-dia são utilizadas no modelo para medir, respectivamente, o tamanho do trabalho e a quantidade de esforço necessário para terminar a especificação. Apresenta também o modelo de dinâmica de sistemas construído e descreve as variáveis e os

relacionamentos que formam a sua estrutura. Ao longo da descrição do modelo, são apresentadas as referências aos trabalhos publicados que serviram como base para a modelagem e a quantificação dos relacionamentos entre as variáveis.

O Capítulo 4 apresenta o painel de controle do modelo, constituído pelas variáveis de configuração e pelas variáveis de análise, que permite aos gerentes ajustar o modelo antes de realizar as simulações. Apresenta também as variáveis-gráfico presentes no modelo e discute o propósito dessas variáveis. Esse capítulo disponibiliza uma classificação, embasada na literatura, dos valores de algumas variáveis visando a direcionar a definição de cenários para a realização das simulações. Por fim, são relatadas as principais análises gerenciais suportadas pelo modelo.

No Capítulo 5 são apresentados os resultados de simulações realizadas com o objetivo de mostrar a viabilidade e a aplicabilidade do modelo em ser utilizado como uma ferramenta de apoio à tomada de decisão durante a fase de requisitos. As simulações apresentadas mostram como o modelo pode ser utilizado para verificar os impactos da materialização de alguns riscos e os efeitos de algumas decisões gerenciais que podem ser implementadas. Para ajustar o modelo e realizar as simulações foram utilizados dados disponibilizados na literatura e dados coletados em uma empresa de desenvolvimento de software.

Finalmente, o Capítulo 6 apresenta as conclusões obtidas e as principais contribuições disponibilizadas por esse trabalho, destacando as decisões gerenciais suportadas pelo modelo construído. São disponibilizadas também sugestões para trabalhos futuros, ressaltando a possibilidade de novas variáveis e relacionamentos serem adicionados no modelo e a viabilidade da implementação de um simulador gerencial para a fase de requisitos, a partir do modelo construído.

Ao final do documento é apresentado um apêndice contendo um glossário com a definição de todos os elementos do modelo de dinâmica de sistemas construído, que foram descritos no Capítulo 3.

2 REFERENCIAL TEÓRICO

2.1 Requisitos de sistemas de software

Requisitos de software são as características, funções e recursos que os usuários solicitam que sejam implementados no software. Segundo (SOMMERVILLE, 2000), os requisitos para um sistema de software estabelecem as funcionalidades que devem ser disponibilizadas para os usuários e definem as restrições sobre a implementação e o comportamento do sistema. Os requisitos são a base para a definição da arquitetura do sistema, para a implementação propriamente dita, para a geração dos casos de testes e para a validação do sistema junto ao usuário (SAYÃO; LEITE, 2006).

Extrair os requisitos de sistemas, conversar com o usuário e compreender suas necessidades não são tarefas triviais. Os usuários expressam naturalmente os requisitos em seus próprios termos e com o conhecimento implícito de sua área de atuação (SOMMERVILLE, 2000). Os engenheiros de requisitos, que não têm experiência no domínio do cliente, necessitam compreender esses requisitos para que o sistema que será desenvolvido atenda às necessidades dos usuários.

Em geral, os usuários entendem o domínio do problema e não o processo de desenvolvimento de software, enquanto que os analistas de sistemas compreendem pouco o domínio do problema. São grandes os riscos de ocorrerem distúrbios na comunicação e os requisitos não serem compreendidos corretamente pelos analistas.

Para evitar os erros de entendimento, os analistas devem compreender o domínio do problema e as rotinas e procedimentos que serão implementados no software. Esse conhecimento pode ser obtido via reuniões com o cliente, leituras de manuais de procedimentos e até mesmo pelo emprego da técnica de etnografia (HUGHES *et al.*, 1995) em que o analista realiza visitas ao ambiente e às instalações dos usuários para acompanhamento de suas rotinas. A etnografia pode fornecer um entendimento detalhado acerca de como o trabalho é realmente realizado (VILLER; SOMMERVILLE, 1999), refletindo os processos reais em que as pessoas estão envolvidas.

Quando não são detectadas pelas atividades de garantia de qualidade, as falhas cometidas durante o levantamento e especificação dos requisitos podem acarretar a implementação de um produto que não irá atender completamente às necessidades dos usuários. Alguns trabalhos (SOMMERVILLE; SAWYER, 1997; KOTONYA; SOMMERVILLE, 1998) consideram a deficiência nas atividades realizadas durante a fase de requisitos como uma das principais causas de fracassos em projetos de software.

É necessário empenho para que os requisitos sejam compreendidos e especificados corretamente desde o início do projeto. Quanto mais tarde, durante o desenvolvimento, um erro de requisito for descoberto, maior serão o custo e o tempo gastos para remover esse erro. Segundo (BLACKBURN; BUSSEY; NAUMAN, 2001), corrigir um erro de requisitos quando o software estiver em testes pode ser até 100 vezes mais caro que descobrir e corrigir o problema na fase de requisitos.

Após serem extraídos e especificados, os requisitos podem evoluir ao longo do desenvolvimento do projeto (ANDERSON; FELICI, 2002). A evolução dos requisitos é inerente ao processo de desenvolvimento de software porque os desenvolvedores e usuários não são capazes de prever todas as maneiras como o sistema será utilizado (ZOWGHI; OFFEN, 1997).

As alterações nos requisitos podem ser necessárias devido a novas necessidades dos usuários, mudanças no ambiente em que o sistema será utilizado ou maior entendimento do domínio do problema por parte dos analistas e usuários. As solicitações de alteração podem ocorrer após o software ser entregue, sendo atendidas por meio de manutenções; ou durante o desenvolvimento do sistema, o que, segundo (BOEHM; PAPACCIO, 1988), pode comprometer os custos e prazos do projeto.

DeMarco; Lister (2003) classificam a alteração de requisitos como um dos cinco tipos de risco que mais contribuem para o aumento do prazo e do custo planejados em projetos de software. Por ser algo intrínseco ao domínio do problema em que muitos sistemas irão atuar, a evolução dos requisitos não pode ser controlada ou impedida pela gerência. O ambiente onde o software está situado pode sofrer evoluções frequentes, o que exige modificações nos limites do software e nas regras de negócio que governam o seu uso (ZOWGHI; OFFEN, 1997).

No entanto, é necessário que a gerência adote um plano para lidar com as alterações de requisitos. Uma ação gerencial que pode diminuir os impactos sobre o

orçamento do projeto consiste em identificar e monitorar os requisitos instáveis, ou seja, que possuem maiores chances de serem alterados, visando a antecipar as modificações. Segundo (KOTONYA; SOMMERVILLE, 1998), a tentativa de antecipar as mudanças dos requisitos é considerada uma boa prática de gerência de requisitos.

Gerenciar as mudanças de requisitos é essencial para o sucesso dos projetos de software (JONES, 1996a), e esse trabalho é melhor desempenhado se os requisitos forem rastreáveis. Empregar a rastreabilidade de requisitos possibilita um gerenciamento das correções e das alterações de requisitos, permitindo verificar os seus efeitos sobre o orçamento do projeto. Segundo (SAYÃO; LEITE, 2006), a rastreabilidade de requisitos pode ser vista como a habilidade de acompanhar e descrever a vida de um requisito, em ambas as direções. A rastreabilidade de um requisito consiste em identificar e manter os artefatos que o originam, como os planos de negócios, e os artefatos originados a partir de cada requisito, como os artefatos de desenho, de implementação e de testes.

Diante da necessidade de alterações ou reparos, um conjunto de requisitos rastreáveis permite identificar os artefatos que devem ser alterados, possibilitando obter uma estimativa mais precisa do esforço e do tempo necessários (SAYÃO; LEITE, 2006; RAMESH; JARKE, 2001). A rastreabilidade também agiliza a investigação das causas de falhas nos testes. Ao deparar-se com uma falha durante o teste de um módulo, é possível obter os respectivos documentos e artefatos produzidos nas fases de requisitos e design (PINHEIRO, 2004).

Como uma forma de diminuir os riscos quanto à implementação dos requisitos e obter um *feedback* mais cedo durante o processo de desenvolvimento, muitas organizações estão adotando um processo de desenvolvimento iterativo e incremental, como o modelo de ciclo em espiral (BOEHM, 1988) e o RUP (*Rational Unified Process*) (KRUCHTEN, 2000). Esses processos consistem em obter, em um primeiro momento, uma visão geral, sem muitos detalhes, de todos os requisitos do sistema. Posteriormente são planejados os ciclos de desenvolvimento, de modo que em cada iteração, uma determinada porção dos requisitos é especificada, implementada, integrada à porção previamente implementada e entregue aos usuários. Ao aplicar um processo iterativo, as funcionalidades são testadas logo que são implementadas, os clientes percebem o avanço do projeto e as alterações e/ou correções que forem necessárias podem ser realizadas antes do fim do projeto.

A importância da gestão dos requisitos é enfatizada por modelos de maturidade do processo de desenvolvimento de software como o CMMI (*Capability Maturity Model Integration*) (SEI, 2008) e o mpsBR (Melhoria do Processo do Software Brasileiro) (SOFTEX, 2007). O CMMI possui em seu segundo nível de maturidade a área chave de processo “Gestão de Requisitos”. O mesmo ocorre com o mpsBR que em seu nível de maturidade G avalia o processo “Gerência de Requisitos” da organização.

Apesar de ser essencial para o desenvolvimento de sistemas corretos, o trabalho realizado durante a fase de requisitos e a gestão de requisitos são deixados em plano secundário. Isso ocorre em parte devido às pressões de mercado, à alta competitividade do setor e ao tamanho das empresas, que não dispõem de recursos humanos em quantidade e qualidade suficientes. A partir de um estudo realizado, Capers Jones (JONES, 1996b) descobriu que o processo de requisitos esteve deficiente em mais de 75% das organizações. Zahedi (1995) relata que erros na fase de requisitos consomem 82% dos recursos gastos em correção de erros nos projetos, enquanto que erros no design consomem 13%, erros na codificação consomem 1%, e nas demais fases consomem 4%.

2.2 Dinâmica de sistemas

A dinâmica de sistemas é uma técnica descritiva, utilizada para modelagem e simulação de sistemas (BRAGA *et al.*, 2004), baseada no pensamento e análise sistêmicos e na teoria matemática dos Sistemas Dinâmicos (STERMAN, 2000), que permite levar em consideração as variações dinâmicas dos problemas. Sua grande utilidade decorre de permitir a compreensão de como as políticas adotadas, ou a própria estrutura do sistema, afetam ou determinam o seu comportamento dinâmico (STERMAN, 2000), antecipando colapsos. A utilização dessa técnica permite a análise e compreensão de problemas e situações de maneira integrada e interconectada, deixando claras as relações existentes entre as variáveis de decisão.

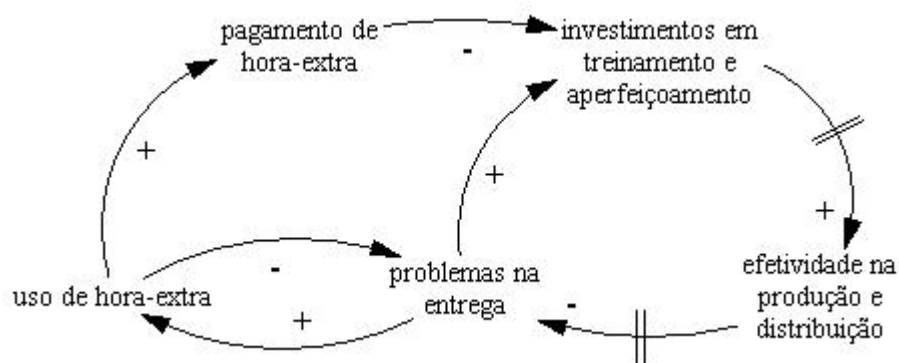
A dinâmica de sistemas foi introduzida por Jay Forrester (FORRESTER, 1961) como um método para modelar e analisar o comportamento de sistemas complexos, formados por diversas variáveis que se relacionam de forma não linear. Nesses sistemas, as diversas variáveis presentes exercem influência umas sobre as outras, formando laços de realimentação e determinando o comportamento do sistema. Esses laços são críticos, pois alterações em algumas variáveis podem desencadear

comportamentos que podem levar todo o sistema a entrar em colapso. A dinâmica de sistemas permite expressar as influências entre as variáveis e capturar os laços de realimentação responsáveis pelo comportamento do sistema (RODRIGUES; BOWERS, 1996).

Diante da necessidade de realizar alguma intervenção no sistema para resolver algum problema, é fundamental conhecer as verdadeiras causas do comportamento indesejado para que a intervenção não seja apenas uma solução paliativa e agrave ainda mais o problema. A modelagem com dinâmica de sistemas facilita a descoberta das causas do problema e também, por meio de simulações utilizando o modelo (STERMAN, 1992), permite analisar os impactos e os efeitos colaterais das alterações planejadas antes que elas sejam implementadas no sistema real.

Uma fase inicial da modelagem normalmente é desenvolvida utilizando os diagramas de influência (SENGE, 1990). Esses diagramas permitem visualizar graficamente as relações entre as variáveis chave do problema, de modo que os gerentes possam localizar possíveis laços de realimentação e atrasos nos efeitos resultantes das interações entre essas variáveis (BRAGA *et al.*, 2004). O diagrama de influência serve como base para a obtenção de um modelo de dinâmica de sistemas.

Na Figura 2.1 apresenta-se um exemplo de um diagrama de influência. As ligações rotuladas com “+” indicam que as variáveis variam na mesma direção (quando uma aumenta/diminui a outra aumenta/diminui) e as rotuladas com “-” indicam variação em direções opostas (quando uma aumenta/diminui a outra diminui/aumenta). Os dois traços paralelos que cruzam algumas ligações entre as variáveis indicam que o efeito de uma variável sobre a outra não é imediato, sendo observável somente com uma defasagem de tempo.



Fonte: (GOODMAN *et al.*, 1997)

Figura 2.1. Diagrama de influência

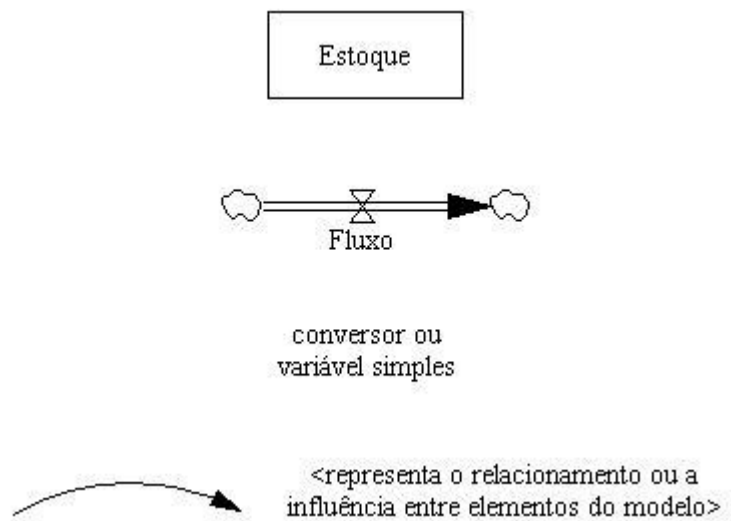
O diagrama de influência apresentado na Figura 2.1 representa uma situação que, segundo (GOODMAN *et al.*, 1997), ocorre freqüentemente nas organizações. Esse diagrama mostra que há um problema na entrega de produtos aos clientes e a organização emprega uma solução paliativa de efeito imediato que é utilizar horas extras dos empregados, enquanto que a solução fundamental e duradoura seria aperfeiçoar o sistema de produção e distribuição. A solução fundamental, por não ter efeito visível imediato, acaba sendo descartada e a solução paliativa agrava o problema com o passar do tempo, pois as despesas com horas extras consomem recursos que poderiam ser investidos em aperfeiçoamentos. Isso causa um laço de realimentação positivo, denominado nesse caso de círculo vicioso, que pode levar todo o sistema a entrar em colapso.

A construção de diagramas de influência permite identificar as variáveis principais e os respectivos relacionamentos existentes entre as mesmas no sistema ou problema que se pretende estudar. O entendimento acerca da estrutura do problema, obtido com esses diagramas, facilita a construção do modelo de dinâmica de sistemas.

Um modelo de dinâmica de sistemas possui dois elementos principais: os estoques, que são os recursos acumuláveis do sistema, e os fluxos, que são funções que representam as decisões ou políticas da empresa com relação aos usos e acúmulos dos estoques ou recursos. São exemplos de estoques: quantidade de requisitos especificados, quantidade de erros cometidos, tamanho da força de trabalho, etc. E são exemplos de fluxos: capacidade de detecção de erros por unidade de tempo, *turnover* de profissionais alocados ao projeto, taxa de especificação de novos requisitos, etc. Há também os conversores, ou variáveis simples, que são os elementos do modelo que exercem influência sobre os valores dos fluxos responsáveis pela variação dos estoques. A Figura 2.2 apresenta os símbolos dos elementos contidos em um modelo de dinâmica de sistemas.

Antes de construir um modelo de dinâmica de sistemas é necessário identificar as variáveis envolvidas no problema ou sistema que se pretende modelar e estudar. Algumas dessas variáveis serão representadas por estoques, outras por fluxos e outras por conversores (ou variáveis simples). Os estoques são utilizados para representar algo que sofre acúmulos ou perdas ao longo do tempo. O valor de um estoque em um tempo t_2 depende do seu valor no tempo t_1 e da variação ocorrida no intervalo $\Delta t = t_2 - t_1$. Os fluxos são os elementos responsáveis pelo crescimento ou

decrescimento dos estoques. No restante do texto será utilizado o termo “variável” para se referir aos conversores do modelo.



Fonte: Elaborado pelo Autor

Figura 2.2. Símbolos dos elementos contidos em um modelo estoque-fluxo da Dinâmica de Sistemas

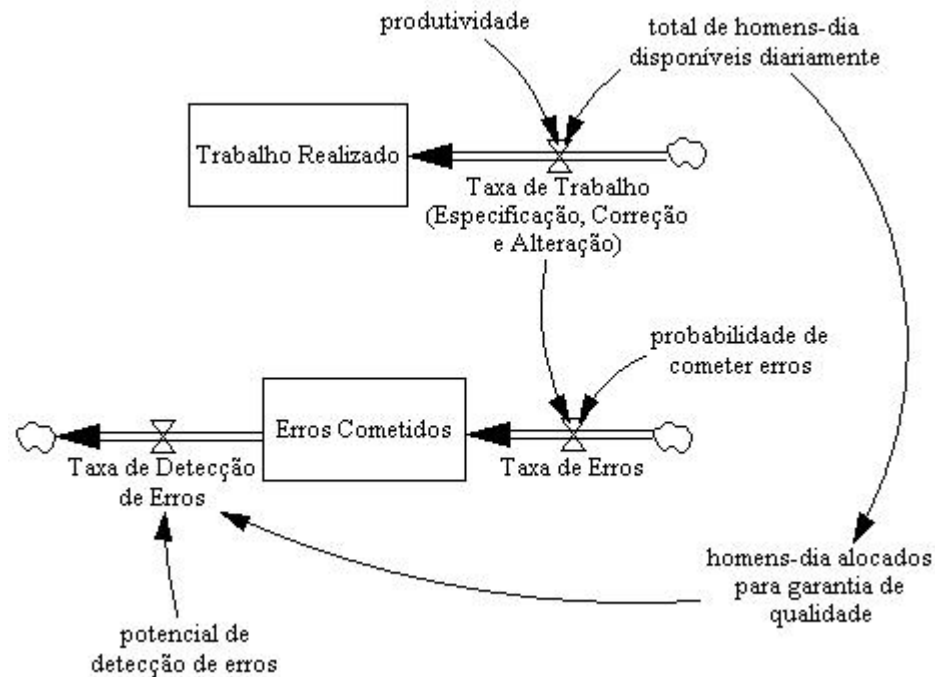
Para ilustrar os elementos (estoques, fluxos e variáveis) de um modelo de dinâmica de sistemas, a Figura 2.3 apresenta uma pequena porção do modelo construído nesse trabalho. O estoque *Trabalho Realizado* representa a quantidade de trabalho realizado pela equipe durante a especificação dos requisitos. E o estoque *Erros Cometidos* representa a quantidade de erros de especificação cometidos que ainda não foram detectados pelas atividades de garantia de qualidade.

O fluxo *Taxa de Trabalho* representa a taxa com que a equipe realiza as atividades e é responsável por aumentar o valor armazenado no estoque *Trabalho Realizado*. O valor desse fluxo é determinado pelos valores das variáveis *total de homens-dia disponíveis diariamente* e *produtividade*.

O fluxo *Taxa de Erros* representa a taxa com que os erros de especificação são cometidos pela equipe durante a realização do trabalho e é responsável por aumentar o valor armazenado no estoque *Erros Cometidos*. O valor desse fluxo é determinado pelos valores do fluxo *Taxa de Trabalho* e da variável *probabilidade de cometer erros*.

O fluxo *Taxa de Detecção de Erros* representa a taxa com que os erros cometidos são detectados pela equipe durante as atividades de garantia de qualidade e é responsável por diminuir o valor acumulado no estoque *Erros Cometidos*. O valor

desse fluxo é determinado pelos valores das variáveis *homens-dia alocados para garantia de qualidade e potencial de detecção de erros*.



Fonte: Elaborado pelo Autor

Figura 2.3. Modelo estoque-fluxo da Dinâmica de Sistemas

A dinâmica de sistemas é uma técnica de modelagem que não está diretamente associada a nenhuma disciplina ou área de conhecimento específica (RICHMOND, 2001), podendo ser utilizada para modelar sistemas com comportamento dinâmico de qualquer domínio de conhecimento. Em problemas que envolvem especialistas de diversas áreas, a dinâmica de sistemas facilita a discussão e a cooperação entre os envolvidos, pois possui um vocabulário comum que permite e incentiva a comunicação.

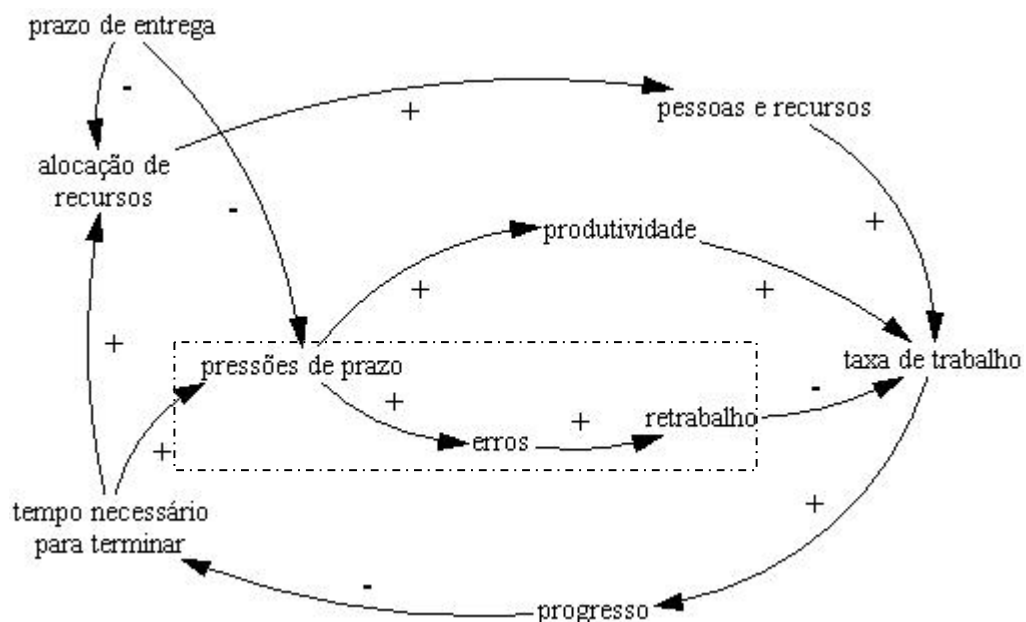
2.3 Análise e pensamento sistêmicos no gerenciamento de projetos de software

Nos últimos anos, as pesquisas em Engenharia de Software contribuíram para diversas melhorias nos aspectos técnicos do desenvolvimento de software. Entretanto, houve uma escassez de trabalhos que investigassem os aspectos gerenciais do processo de desenvolvimento de software. Essa deficiência vem contribuindo para a falta de compreensão e entendimento acerca desse processo e de como as decisões tomadas podem influenciar no sucesso do projeto (ABDEL-

HAMID; MADNICK, 1991; KAPPELMAN; MCKEEMAN; ZHANG, 2006; KING; BURGESS, 2006; HOOD; HOOD, 2006).

A falta de uma visão integradora ou holista do processo de desenvolvimento de software leva os gerentes a tomarem decisões reativas, considerando apenas o problema presente, sem relacioná-lo com o seu ambiente, suas variáveis e demais problemas correlacionados. Essa deficiência de embasamento conceitual foi apresentada e discutida por Peter Senge (SENGE, 1990), que deixou claro que a capacidade de enxergar e compreender o sistema como um todo é fundamental nas decisões modernas.

Um exemplo de decisão reativa no gerenciamento de projetos de software é descrito em (ABDEL-HAMID; MADNICK, 1991) e seu diagrama de influência aparece na Figura 2.4. Diante de pressões de prazo que surgem quando um projeto está atrasado, os gerentes geralmente incentivam os desenvolvedores a se concentrarem mais nas tarefas e a trabalharem mais horas durante vários dias, aumentando a produtividade da equipe. No diagrama de influência, esse relacionamento é representado pela influência que a variável *pressões de prazo* exerce sobre a variável *produtividade* e pela influência que esta variável exerce sobre a variável *taxa de trabalho*.



Fonte: (ABDEL-HAMID; MADNICK, 1991)

Figura 2.4. Relacionamentos entre variáveis envolvidas no gerenciamento de projetos de software

Entretanto, as pressões de prazo podem elevar a quantidade de erros cometidos, devido ao estresse e cansaço dos desenvolvedores, e conseqüentemente podem aumentar a quantidade de retrabalho no projeto, o que muitas vezes não é considerado pelos gerentes. Isso diminui o progresso do projeto e o atrasa ainda mais, levando depois de certo tempo a uma queda na produtividade. Esse relacionamento menos perceptível é apresentado de forma destacada no diagrama de influência.

A análise sistêmica desse problema, considerando os efeitos das pressões de prazo sobre a quantidade de erros e de retrabalho no projeto, mostra que o que aparentemente é um ciclo virtuoso é, de fato, um ciclo vicioso. A modelagem do processo de gerenciamento de projetos de software com dinâmica de sistemas permite aos gerentes obter um maior entendimento sobre o processo e evitar as decisões reativas.

Os modelos de dinâmica de sistemas permitem criar os “laboratórios de aprendizagem” nas organizações (STERMAN, 1992). Diante de algum problema ou situação indesejada, os gerentes podem utilizar os modelos para identificar as verdadeiras causas do problema e determinar as ações efetivas para solucioná-lo. O processo de descoberta da solução fundamental pode ser realizado via a utilização dos modelos como uma ferramenta de previsão e de apoio à tomada de decisão. Após modelar o problema ou sistema, simular o modelo e analisar os resultados gerados, os gerentes podem verificar os impactos das intervenções planejadas antes de implementá-las.

2.4 Trabalhos correlatos

A realização de trabalhos e pesquisas envolvendo a aplicação da técnica de dinâmica de sistemas na modelagem do processo de gerenciamento de projetos não é algo recente. A tese de doutorado do professor Edwards B. Robert com o título “*The Dynamics of Research and Development*” (ROBERTS, 1964), completada em 1962, foi o primeiro trabalho realizado com o objetivo de aplicar a metodologia de dinâmica de sistemas em gerenciamento de projetos. Desde então, ele continuou trabalhando na área gerando conhecimento e novas idéias por meio do estudo da aplicação de dinâmica de sistemas em projetos de pesquisa e desenvolvimento.

Devido a semelhanças existentes entre os estágios do processo de desenvolvimento de projetos de pesquisa e desenvolvimento e dos processos de

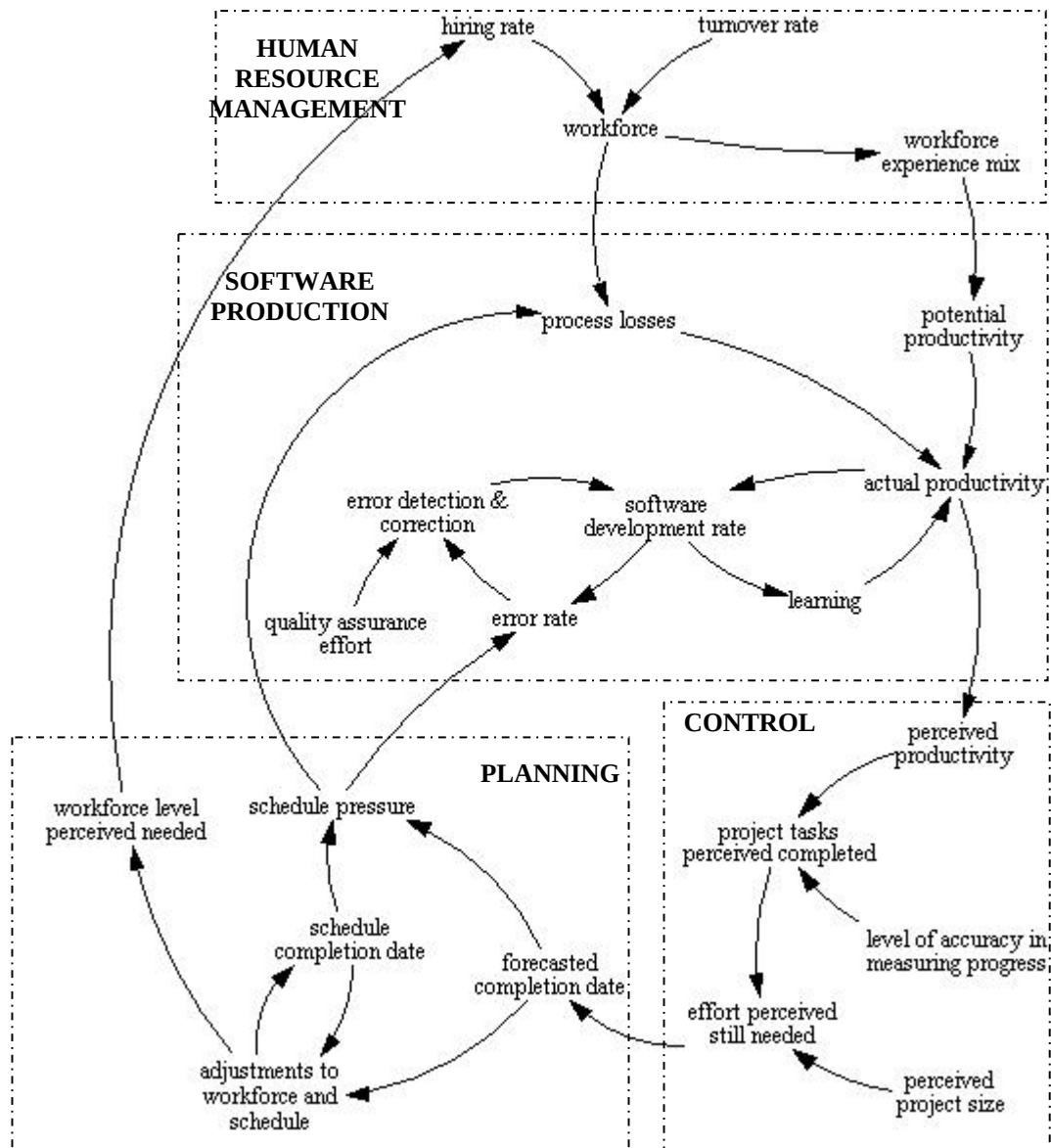
engenharia de software, pesquisadores começaram a investigar a aplicação das técnicas de dinâmica de sistemas na modelagem do processo de gerenciamento de projetos de software.

Um trabalho que se destacou quanto à construção de modelos de dinâmica de sistema para estudo do processo de gerenciamento de projetos de software foi realizado por Tarek Abdel-Hamid e Stuart E. Madnick. Esse trabalho está relatado no livro “*Software Project Dynamics: an Integrated Approach*” (ABDEL-HAMID; MADNICK, 1991) e no artigo “*Lessons Learned from Modeling the Dynamics of Software Development*” (ABDEL-HAMID; MADNICK, 1989). O objetivo do trabalho foi desenvolver e testar uma visão integrada do gerenciamento de projetos de software visando a aumentar a compreensão e permitir a realização de previsões sobre o processo por meio do qual o desenvolvimento é gerenciado.

Abdel-Hamid; Madnick (1991) apresentam um modelo integrado baseado nas técnicas de dinâmica de sistemas para estudo do processo de gerenciamento de projetos de software. O trabalho revela um método para representar o relacionamento entre as diversas variáveis que afetam o desenvolvimento de software. Por meio de simulações, o modelo pode ser utilizado como um *framework* para a realização de experimentos visando a testar o impacto de novas políticas gerenciais. Os efeitos de diferentes suposições podem ser verificados. Com a simulação do modelo, o efeito da alteração de uma variável ou relacionamento sobre o projeto pode ser observado, enquanto o restante do modelo permanece inalterado. A execução desses experimentos em um ambiente real é inviável devido ao custo e tempo necessários para reproduzir os projetos de software.

Para verificar a aplicabilidade do modelo, os pesquisadores realizaram um estudo de caso com um projeto da NASA (*National Aeronautics and Space Administration*). O modelo replicou com precisão o comportamento dinâmico de um projeto real de software. O padrão de variação e o valor das variáveis analisadas na simulação foram muito próximos dos valores reais do projeto. Entre as variáveis verificadas tem-se: o nível da força de trabalho, cronograma e custo do projeto.

Posteriormente, o modelo foi utilizado como uma ferramenta para estudar ou prever as implicações dinâmicas de diversas políticas gerenciais. Os autores focaram em quatro áreas: cronograma, qualidade, controle e recursos humanos. O diagrama de influência que descreve a estrutura do modelo é apresentado na Figura 2.5.



Fonte: (ABDEL-HAMID; MADNICK, 1991)

Figura 2.5. Diagrama de influência que descreve a estrutura do modelo dinâmico para o gerenciamento de projetos de software

Os experimentos realizados com o modelo permitiram enxergar consequências desconhecidas de algumas políticas comuns nas organizações e forneceram novos conhecimentos e orientações para os gerentes. Esses experimentos mostraram que cada estimativa para um projeto de software origina um projeto diferente, ou seja, os prazos disponíveis influenciam diretamente nas decisões e no comportamento dos envolvidos no projeto, aumentando ou diminuindo a produtividade da equipe. Em geral, quanto maior for o fator de segurança para a estimativa do prazo, maior será o custo do projeto porque os desenvolvedores se sentem mais confortáveis, tornando-se menos produtivos.

Os experimentos testaram também a veracidade da Lei do Brooks (BROOKS, 1995) que afirma que adicionar mais pessoas a um projeto que está atrasado irá atrasá-lo ainda mais. Os resultados mostraram que para projetos de tamanho médio, adicionar mais pessoas a um projeto atrasado o torna mais caro, mas nem sempre o atrasa ainda mais. Para que o atraso seja agravado, a alocação de pessoas deve continuar a ocorrer até o fim da fase de testes.

A partir do modelo construído por Abdel-Hamid e Madnick, alguns trabalhos foram realizados visando a aplicar a dinâmica de sistemas no gerenciamento de projetos de software: (ABDEL-HAMID, 1996), (LIN; ABDEL-HAMID; SHERIF, 1997), (COLLOFELLO *et al.*, 1998), (SENGUPTA; ABDEL-HAMID; BOSLEY, 1999), (ZHANG *et al.*, 2006) e outros. Rodrigues (2001) apresenta um *framework* baseado em dinâmica de sistemas para o gerenciamento e modelagem do comportamento dinâmico de riscos em projetos de software.

No Brasil também existem grupos que desenvolvem trabalhos nessa área. Pesquisadores que atuam na COPPE, na Universidade Federal do Rio de Janeiro, vêm trabalhando com modelagem baseada em técnicas de dinâmica de sistemas aplicada ao processo de gerenciamento de projetos de software. Os trabalhos desses pesquisadores vêm focando a construção de modelos para simular situações de risco nos projetos de desenvolvimento de software. O objetivo é oferecer conhecimento, permitir aos gerentes analisar e avaliar os riscos dos projetos e verificar os impactos das intervenções planejadas.

Dentre os trabalhos realizados na COPPE, em (BARROS; WERNER; TRAVASSOS, 1999) é apresentado o uso de modelos dinâmicos para suportar a identificação e análise de riscos em projetos de software. Em (BARROS; WERNER; TRAVASSOS, 2003) é proposto o uso de cenários para separar fatos de suposições nos modelos de dinâmica de sistemas aplicados na modelagem de processos de desenvolvimento de software. Uma técnica que auxilia na análise de modelos de dinâmica de sistemas, permitindo rastrear as mudanças em algumas variáveis do modelo, é apresentada em (BARROS; WERNER; TRAVASSOS, 2001a). Em (BARROS; WERNER; TRAVASSOS, 2001b) é proposta uma técnica para o desenvolvimento de metamodelos de domínios, que podem ser especializados para descrever problemas e situações particulares do domínio. Em (BARROS; WERNER; TRAVASSOS, 2001c) é sugerida a construção de um modelo para representar o comportamento esperado de projetos de software e de modelos complementares para

representar os possíveis cenários diferentes que podem ocorrer durante o desenvolvimento do projeto.

No entanto, durante as pesquisas realizadas, não foram encontrados trabalhos que abordam a modelagem das variáveis e dos relacionamentos envolvidos na fase de requisitos em processos de desenvolvimento de software.

Os processos de desenvolvimento de software podem ser classificados em dois grandes grupos: métodos ágeis e métodos dirigidos por planejamento, também chamados de métodos disciplinados. De acordo com as características do projeto, um método pode ser mais adequado que o outro (BOEHM; TURNER, 2003a). Métodos dirigidos por planejamento são mais adequados para sistemas grandes e complexos, com requisitos críticos de segurança e confiabilidade, requisitos estáveis e ambiente previsível. Métodos ágeis são mais adequados quando a equipe de desenvolvimento é menor, o cliente participa ativamente do projeto esclarecendo as dúvidas durante o desenvolvimento, e os requisitos e o ambiente são voláteis.

No entanto, o desenvolvimento de software requer ambas, agilidade e disciplina. Em muitos projetos, é necessário combinar as práticas dos métodos ágeis com as dos métodos dirigidos por planejamento, formando um processo híbrido. Em seu livro *“Balancing Agility and Discipline: A Guide for the Perplexed”* (BOEHM; TURNER, 2003a), Boehm e Turner disponibilizam um método, baseado na análise de riscos, para auxiliar os gerentes a realizar um balanceamento adequado entre agilidade e disciplina no processo utilizado para o desenvolvimento de software. Os critérios utilizados no balanceamento são baseados nos riscos quanto ao uso de métodos ágeis ou de métodos dirigidos por planejamento, ao analisar várias características dos projetos. Por exemplo, a existência de requisitos críticos de escalabilidade e segurança é um risco para o uso de métodos ágeis; enquanto que a necessidade de atender rapidamente às frequentes solicitações de mudanças no sistema é um risco para o uso de métodos dirigidos por planejamento.

Entre as características dos projetos avaliadas durante a análise de riscos no método proposto por Boehm e Turner, estão: tamanho da equipe, *turnover* de pessoal, competência dos profissionais, volatilidade dos requisitos e existência de requisitos emergentes não previstos inicialmente. Essas características correspondem a riscos que são abordados pelo modelo de dinâmica de sistemas apresentado nesse trabalho. O modelo pode então, ser expandido para auxiliar na verificação das

consequências do balanceamento entre agilidade e disciplina durante a fase de requisitos, complementando o trabalho de Boehm e Turner.

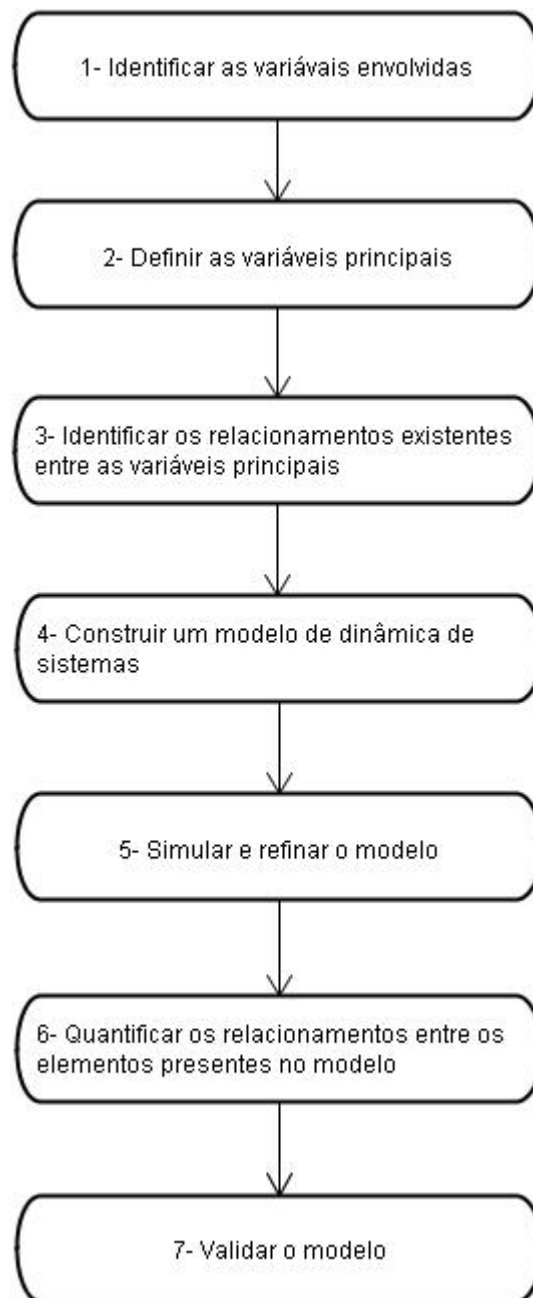
3 CONSTRUÇÃO DE UM MODELO DINÂMICO PARA A FASE DE REQUISITOS

Após o estudo e a compreensão do comportamento dinâmico dos requisitos e das principais variáveis envolvidas na fase de requisitos, foi definido e construído um modelo de dinâmica de sistemas abrangendo as principais variáveis e os respectivos relacionamentos existentes entre elas. O modelo é baseado em um processo de desenvolvimento iterativo, como o ciclo em espiral (BOEHM, 1988) e o RUP (*Rational Unified Process*) (KRUCHTEN, 2000), em que os requisitos do sistema são implementados e entregues em diversas liberações (*releases*). Cada liberação constitui uma versão operacional parcial que implementa um subconjunto das funcionalidades do sistema. A simulação do modelo permite observar o comportamento das variáveis durante a especificação dos requisitos de uma determinada liberação.

3.1 O processo de construção do modelo

O processo utilizado para construir o modelo de dinâmica de sistemas apresentado nesse trabalho é constituído pelas atividades contidas no diagrama de atividades apresentado na Figura 3.1.

A primeira atividade consistiu em identificar as variáveis envolvidas na fase de requisitos em processos de desenvolvimento de software. Entre essas variáveis, estão as utilizadas para determinar as decisões gerenciais tomadas e as que representam os riscos (alteração de requisitos, *turnover* entre os profissionais e outros) que podem ocorrer. As variáveis foram identificadas a partir da leitura de livros que abordam a fase de requisitos e de artigos científicos que descrevem estudos e pesquisas relacionadas com essa fase dos processos de desenvolvimento de software. A experiência dos autores no desenvolvimento de sistemas de software e as trocas de experiências não documentadas com profissionais de empresas de desenvolvimento de software também auxiliaram na identificação das variáveis.



Fonte: Elaborado pelo Autor

Figura 3.1. Diagrama de atividades para a construção do modelo

A segunda atividade teve por objetivo definir as variáveis principais, a partir das variáveis obtidas na atividade anterior. Foram definidas as variáveis que são investigadas nesse trabalho e que estão incluídas no modelo de dinâmica de sistemas construído. Para diminuir o tamanho e a complexidade do modelo, foram analisadas apenas as variáveis que afetam consideravelmente a fase de requisitos e cujas influências sobre as demais variáveis correlacionadas podem ser confirmadas por dados publicados na literatura.

A partir de informações disponibilizadas na literatura, na terceira atividade foram identificados os relacionamentos existentes entre as variáveis principais. Após definir as variáveis e os relacionamentos que seriam investigados, a quarta atividade visou a construção de um modelo de dinâmica de sistemas (modelo estoque-fluxo), abrangendo as variáveis principais e os relacionamentos identificados anteriormente. Para construir o modelo e realizar as simulações, foi utilizada uma versão gratuita para uso acadêmico da ferramenta Vensim (VENSIM, 2008).

A quinta atividade consistiu em simular e refinar o modelo. Ao simular o modelo e analisar os resultados gerados, foi obtido um maior entendimento acerca das interações entre as variáveis e verificou-se a necessidade de realizar ajustes nos relacionamentos entre os elementos (estoques, fluxos e variáveis) do modelo. Esses ajustes foram realizados em diversas etapas com o objetivo de estabilizar o modelo para que não fossem gerados resultados inconsistentes.

A sexta atividade teve por objetivo quantificar os relacionamentos entre os elementos do modelo, a partir de informações e dados encontrados na literatura. Conforme será apresentado mais adiante ao longo do texto, alguns desses relacionamentos podem ser modificados pelo usuário, possibilitando verificar o comportamento do modelo diante de novas suposições.

A sétima atividade teve por objetivo validar o modelo, após obter uma versão final do mesmo. Para isso, foram realizadas simulações de vários cenários e os resultados obtidos mostraram que o comportamento do modelo e o padrão de variação das variáveis estão coerentes com o conhecimento comum da comunidade acadêmica e dos profissionais que atuam na área de Engenharia de Software.

Para especificar os relacionamentos entre as variáveis, não foram utilizados dados de bases contendo dados históricos coletados em projetos de software ao longo do tempo. Essas bases não estão disponíveis para uso público. O conhecimento utilizado para estabelecer (terceira atividade) e quantificar (sexta atividade) as relações entre as diversas variáveis foi obtido a partir de informações e resultados relatados por outros grupos de pesquisa, disponíveis em artigos científicos citados ao longo do texto.

Durante a construção do modelo, as atividades descritas não foram executadas com a estrita seqüência apresentada. Frequentemente, durante a realização de atividades posteriores foram necessárias revisões e alterações nos resultados obtidos em atividades anteriores. O modelo foi construído por meio de um processo iterativo,

retornando às atividades anteriores quando necessário, mas sempre revisando as atividades seguintes para manter a consistência nos resultados obtidos.

Por exemplo, algumas variáveis contidas no modelo foram descobertas durante a validação do mesmo. Essas variáveis foram acrescentadas no modelo, conforme descreve a segunda atividade, e as atividades seguintes foram realizadas novamente para verificar se outros ajustes seriam necessários. Outro exemplo de retomada às atividades anteriores: para tratar algumas inconsistências percebidas durante a validação do modelo na sétima atividade, foram realizados novos ajustes e refinamentos, alterando os resultados obtidos na quinta atividade.

3.2 Relação entre as medidas pontos de função e homens-dia

Para evitar erros de entendimento durante o estudo do modelo, é importante compreender a relação entre as medidas pontos de função e homens-dia utilizadas. A medida “pontos de função” é utilizada para estimar o tamanho e a complexidade do trabalho que deve ser realizado durante a especificação dos requisitos. Análise de pontos de função é um método que permite quantificar o tamanho e a complexidade de um software em termos das funcionalidades que o sistema disponibiliza para o usuário (MATSON; BARRET; MELLICHAMP, 1994).

Para estimar o esforço necessário para o término da especificação, o modelo utiliza a medida homens-dia. Na literatura podem ser encontradas formalizações que descrevem a relação entre o tamanho do produto e o esforço necessário. Algumas dessas formalizações são discutidas em (MATSON; BARRET; MELLICHAMP, 1994). Apesar da formulação específica de cada formalização variar, elas podem ser consideradas como derivadas da equação: $\text{esforço} = (a \cdot \text{tamanho}^b)$, em que a representa o coeficiente de produtividade e b representa o coeficiente de economia (ou deseconomia) de escala (SHEPPERD; SCHOFIELD; KITCHENHAM, 1996).

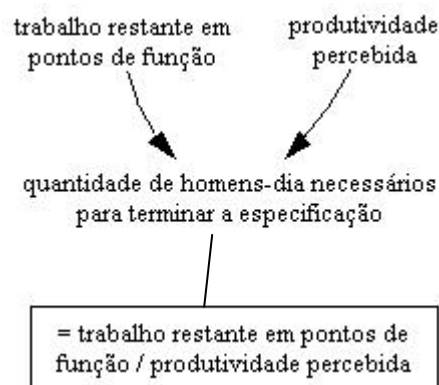
De acordo com (FILHO, 2003), o esforço para o desenvolvimento, medido em homens-mês (HM), pode ser derivado do tamanho do produto medido em pontos de função (PF) através da fórmula: $\text{HM} = (\text{PF} / \text{Produtividade})$. Essa fórmula assume que: $a = 1 / \text{Produtividade}$ e $b = 1$. Para obter a produtividade é necessário manter uma base de dados histórica que armazena o tamanho em pontos de função e a quantidade de esforço gasto em um grupo de projetos semelhantes ao projeto sendo planejado. Pelo fato de essa fórmula não utilizar um coeficiente de economia de

escala, o valor da produtividade deve variar de acordo com as características do projeto. A equação proposta por (FILHO, 2003) pode ser facilmente adaptada para considerar o esforço medido em homens-dia.

O modelo definido nesse trabalho adota uma adaptação da fórmula apresentada em (FILHO, 2003) para calcular o esforço necessário para terminar a especificação. No modelo, o esforço necessário para realizar a especificação dos requisitos é medido em homens-dia e é dado pela equação:

$$\text{Esforço (em homens-dia)} = \frac{\text{Tamanho (em pontos de função)}}{\text{Produtividade (em pontos de função por homens-dia)}}$$

A partir do tamanho do trabalho medido em pontos de função e da produtividade da equipe, o esforço necessário, medido em homens-dia, é determinado. A representação dessa relação no modelo pode ser visualizada na Figura 3.2.



Fonte: Elaborado pelo Autor

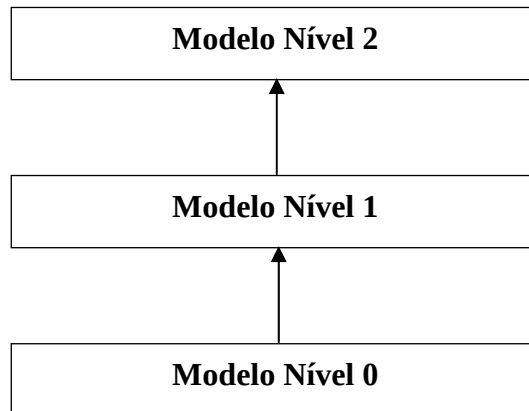
Figura 3.2. Relação entre tamanho do trabalho e esforço necessário

3.3 Visões para análise do modelo

O modelo de dinâmica de sistemas construído pode ser estudado e analisado a partir de três visões diferentes, ilustradas na Figura 3.3.

O Modelo Nível 0 é uma visão que contém todas as variáveis e relacionamentos do modelo e pode ser utilizada para realizar simulações. Algumas variáveis contidas nessa visão são utilizadas apenas para armazenar valores temporários e realizar operações matemáticas durante a simulação. Devido à complexidade e à quantidade de variáveis, essa visão do modelo não é apresentada

nesse texto de forma detalhada. No Capítulo 4 são apresentadas algumas variáveis contidas nessa visão, que estão incluídas no painel de controle do modelo e são utilizadas para ajustar o modelo ao realizar as simulações.



Fonte: Elaborado pelo Autor

Figura 3.3. Visões para estudo e análise do modelo

O Modelo Nível 1 é uma visão que contém as principais variáveis e os respectivos relacionamentos existentes no modelo. Essa é a principal visão do modelo e é utilizada para descrever toda a sua estrutura e as interações dinâmicas entre as variáveis. Por ser mais simples que o Modelo Nível 0, essa visão facilita a explicação e o entendimento acerca dos relacionamentos e da estrutura do modelo. O Modelo Nível 1 é apresentado na seção 3.5.

O Modelo Nível 2 é uma visão mais simplificada do modelo, constituída por apenas um pequeno subconjunto das variáveis contidas no Modelo Nível 1. No Modelo Nível 2, muitos relacionamentos e fatores não estão incluídos, pois o seu objetivo é apresentar a estrutura básica do modelo, facilitando o entendimento posterior do Modelo Nível 1. O Modelo Nível 2 é descrito na seção 3.4.

Para facilitar a compreensão das ilustrações e das descrições do modelo, foi adotado um padrão para a nomeação dos elementos apresentados. Os nomes dos elementos são escritos em *itálico*, sendo que os nomes dos estoques e dos fluxos iniciam com caractere maiúsculo, e os nomes das variáveis (conversores) iniciam com caractere minúsculo.

3.4 Elementos do Modelo Nível 2

A Figura 3.4 permite visualizar a estrutura do Modelo Nível 2. Esse modelo contém apenas cinco estoques:

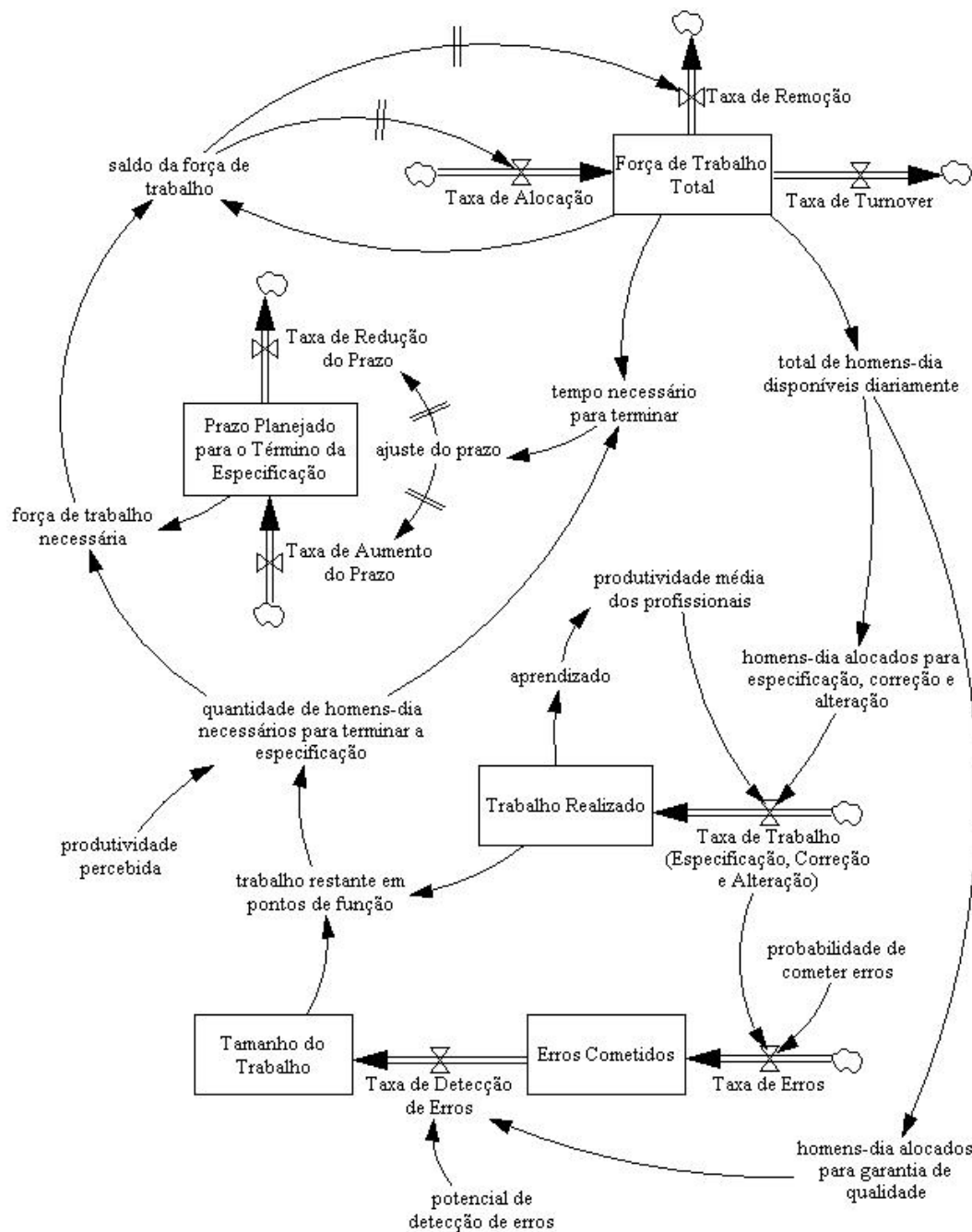
- *Força de Trabalho Total*: representa a quantidade de pessoas alocadas na equipe responsável por realizar a especificação dos requisitos;
- *Trabalho Realizado*: representa a quantidade de trabalho realizado medido em pontos de função;
- *Erros Cometidos*: representa a quantidade de erros cometidos, ainda não detectados pelas atividades de garantia de qualidade;
- *Tamanho do Trabalho*: representa a quantidade de trabalho, medido em pontos de função, que deve ser realizado pela equipe;
- *Prazo Planejado para o Término da Especificação*: representa o prazo planejado para o término da especificação dos requisitos da liberação.

O valor do estoque *Força de Trabalho Total* e a quantidade de esforço, medido em homens-dia, disponibilizado pelos profissionais da equipe determinam o valor da variável *total de homens-dia disponíveis diariamente*. Essa variável representa a quantidade de esforço disponibilizado diariamente pela equipe para a realização das atividades. A variável *homens-dia alocados para garantia de qualidade* representa a porção desse esforço que é alocada para as atividades de garantia de qualidade, como revisões e inspeções. E a variável *homens-dia alocados para especificação, correção e alteração* representa a porção do esforço que é alocada para especificar os requisitos, corrigir as falhas descobertas e realizar as alterações solicitadas.

Os valores das variáveis *homens-dia alocados para especificação, correção e alteração* e *produtividade média dos profissionais* determinam o valor do fluxo *Taxa de Trabalho*. Esse fluxo representa a taxa com que a equipe realiza as atividades no decorrer da especificação dos requisitos e é responsável por aumentar o valor acumulado no estoque *Trabalho Realizado*.

À medida que as atividades são realizadas pela equipe, ocorre um aprendizado sobre o projeto e o domínio do problema, o que aumenta a produtividade da equipe (ABDEL-HAMID, 1996). No modelo, isso é representado pelo relacionamento entre as variáveis *aprendizado* e *produtividade média dos profissionais*.

Durante o desenvolvimento de software, erros também são cometidos pela equipe (LIN; ABDEL-HAMID; SHERIF, 1997; ABDEL-HAMID; MADNICK, 1991). O fluxo *Taxa de Erros* representa a taxa com que os erros são cometidos pela equipe durante a realização das atividades.



Fonte: Elaborado pelo Autor

Figura 3.4. Estrutura do Modelo Nível 2

Para evitar a propagação de defeitos para as fases seguintes do projeto, são realizadas inspeções e revisões dos artefatos produzidos durante a fase de requisitos (KIRNER; ABIB, 1997). O valor da variável *homens-dia alocados para garantia de qualidade*, junto com o valor da variável *potencial de detecção de erros*, determina o valor do fluxo *Taxa de Detecção de Erros*. Esse fluxo representa a taxa com que os erros são detectados pela equipe. Considerando que as falhas detectadas devem ser

corrigidas pela equipe, ocorre um aumento no valor do estoque *Tamanho do Trabalho*.

A diferença entre os valores dos estoques *Tamanho do Trabalho* e *Trabalho Realizado* determina o valor da variável *trabalho restante em pontos de função*, que representa a quantidade de trabalho necessário para especificar os requisitos restantes.

A variável *quantidade de homens-dia necessários para terminar a especificação* corresponde ao esforço necessário para terminar a especificação dos requisitos. Alguns trabalhos como (MATSON; BARRET; MELLICHAMP, 1994), (SHEPPERD; SCHOFIELD; KITCHENHAM, 1996) e (FILHO, 2003) relatam que o esforço necessário para o término de um projeto é determinado por duas variáveis: a quantidade de trabalho que ainda deve ser realizado e a produtividade percebida das pessoas que compõem a equipe. Conforme foi relatado na seção 3.2, o modelo adota uma adaptação da fórmula apresentada em (FILHO, 2003) para calcular o esforço necessário para terminar a especificação. O valor da variável *quantidade de homens-dia necessários para terminar a especificação* é determinado pelos valores das variáveis *trabalho restante em pontos de função* e *produtividade percebida*.

Segundo (ABDEL-HAMID, 1993), o tamanho adequado da equipe pode ser determinado a partir do prazo restante e da quantidade de esforço necessário para terminar o projeto. No modelo, os valores da variável *quantidade de homens-dia necessários para terminar a especificação* e do estoque *Prazo Planejado para o Término da Especificação* determinam o valor da variável *força de trabalho necessária*. Essa variável representa a quantidade de pessoas que devem compor a equipe para que a especificação termine no prazo planejado.

A diferença entre os valores do estoque *Força de Trabalho Total* e da variável *força de trabalho necessária* define o valor da variável *saldo da força de trabalho*. O valor dessa variável pode ser:

- **negativo:** significa que é necessário alocar mais pessoas na equipe. O fluxo *Taxa de Alocação* representa a alocação de novos profissionais na equipe;
- **positivo:** significa que há pessoas em excesso na equipe e é necessário diminuir o seu tamanho. O fluxo *Taxa de Remoção* representa a remoção de profissionais da equipe;
- **zero:** significa que a equipe possui um tamanho adequado.

Se for necessário alocar mais profissionais na equipe, pode demorar algum tempo para a gerência remanejar profissionais de outras equipes ou contratar profissionais disponíveis no mercado (SENGUPTA; ABDEL-HAMID; BOSLEY, 1999; ABDEL-HAMID; MADNICK, 1991). Por isso, o valor do estoque *Força de Trabalho Total* pode ser diferente do valor da variável *força de trabalho necessária*, o que pode levar a realização de ajustes no prazo.

Os valores da variável *quantidade de homens-dia necessários para terminar a especificação* e do estoque *Força de Trabalho Total* determinam o valor da variável *tempo necessário para terminar*. Essa variável representa a quantidade de dias necessários para o término da especificação e determina se devem ser realizados ajustes no prazo. A reavaliação do prazo é representada pela influência que a variável *ajuste do prazo* exerce sobre os fluxos *Taxa de Aumento do Prazo* e *Taxa de Redução do Prazo*.

3.5 Elementos do Modelo Nível 1

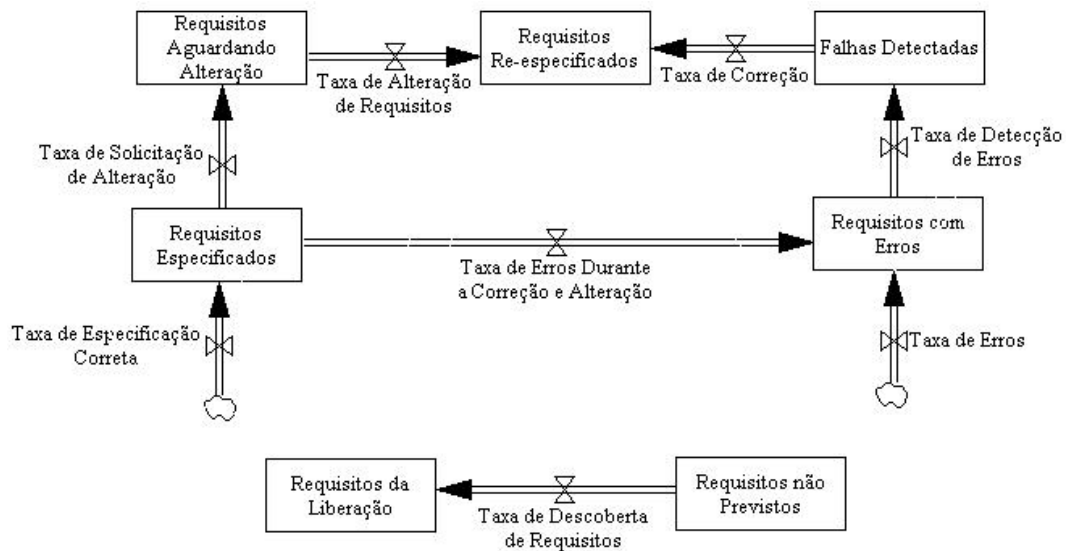
Visando a facilitar o entendimento, o Modelo Nível 1 foi dividido em sub-modelos complementares, que focam em determinadas variáveis e relacionamentos. Esses sub-modelos são apresentados por meio de um diagrama de estoque-fluxo da dinâmica de sistemas e alguns relacionamentos são também representados por um diagrama de influência.

3.5.1 Estado dos requisitos a serem entregues na próxima liberação

A Figura 3.5 apresenta a porção do modelo que contém os estoques utilizados para representar o estado ou condição de cada requisito. O estoque *Requisitos da Liberação* representa todos os requisitos que devem ser especificados e entregues na próxima liberação. O valor desse estoque deve ser definido antes de executar a simulação do modelo. Conhecer a quantidade exata de requisitos antes de terminar a especificação pode não ser possível, mas as reuniões com o cliente e a concepção do sistema permitem que o valor inicial do estoque *Requisitos da Liberação* seja ao menos estimado.

Para obter essa estimativa podem ser utilizadas diversas técnicas de extração de requisitos. Nuseibeh; Easterbrook (2000), Hickey; Davis (2003), Saiedian; Dale (2000) e Demirors; Gencel; Tarhan (2003) apresentam algumas técnicas, tais como: abordagem tradicional, prototipação, oficinas de grupo, análises dos processos de

negócio e outras. A estimativa do valor do estoque *Requisitos da Liberação* é essencial para que se tenha uma noção do esforço e do prazo necessários para terminar a especificação dos requisitos. Isso permite à gerência controlar o progresso da especificação e tomar decisões gerenciais acerca do prazo e do tamanho da equipe.



Fonte: Elaborado pelo Autor

Figura 3.5. Estados dos requisitos

Estimar com boa precisão o tamanho e o custo de projetos na indústria de software continua sendo um grande desafio. Segundo Boehm (BOEHM, 1981), isso ocorre porque há uma tendência em focar nos componentes principais do sistema e subestimar as funcionalidades secundárias (tratamento de erros, mensagens de ajuda e outras). De acordo com (DEMARCO; LISTER, 2003), a maioria dos gerentes é capaz de prever as tarefas que devem ser feitas, mas não conseguem antever as tarefas que eventualmente podem tornar-se necessárias.

O estoque *Requisitos não Previstos* representa os requisitos, não identificados inicialmente, que poderão ser descobertos após o início da especificação. O fluxo *Taxa de Descoberta de Requisitos* representa a taxa com que esses requisitos são descobertos e incluídos entre os requisitos que serão entregues na próxima liberação.

Os demais estoques presentes no modelo representam o estado de cada requisito solicitado pelo cliente. Esses estoques são:

- *Requisitos Especificados*: representa todos os requisitos que foram corretamente especificados;

- *Requisitos com Erros*: representa todos os requisitos especificados que apresentam erros ainda não detectados pelas atividades de garantia de qualidade;
- *Falhas Detectadas*: representa todos os requisitos especificados que apresentam erros já detectados pelas atividades de garantia de qualidade;
- *Requisitos Aguardando Alteração*: representa todos os requisitos especificados que necessitam ser alterados para atender às novas necessidades do cliente;
- *Requisitos Re-especificados*: representa todos os requisitos que foram re-especificados. A re-especificação é realizada quando é necessário corrigir ou alterar algum requisito especificado anteriormente.

As atividades realizadas no decorrer da especificação dos requisitos são responsáveis por acrescentar requisitos nesses estoques e transferir requisitos de um estoque para o outro, alterando o estado dos requisitos. Ao término da simulação do modelo, cada requisito contido no estoque *Requisitos da Liberação* deve estar armazenado também em um dos cinco estoques apresentados acima, ou seja, cada requisito deve estar em um determinado estado. As simulações do modelo terminam quando a soma dos valores acumulados nos estoques *Requisitos Especificados*, *Requisitos com Erros*, *Falhas Detectadas*, *Requisitos Aguardando Alteração* e *Requisitos Re-especificados* for igual ao valor do estoque *Requisitos da Liberação*.

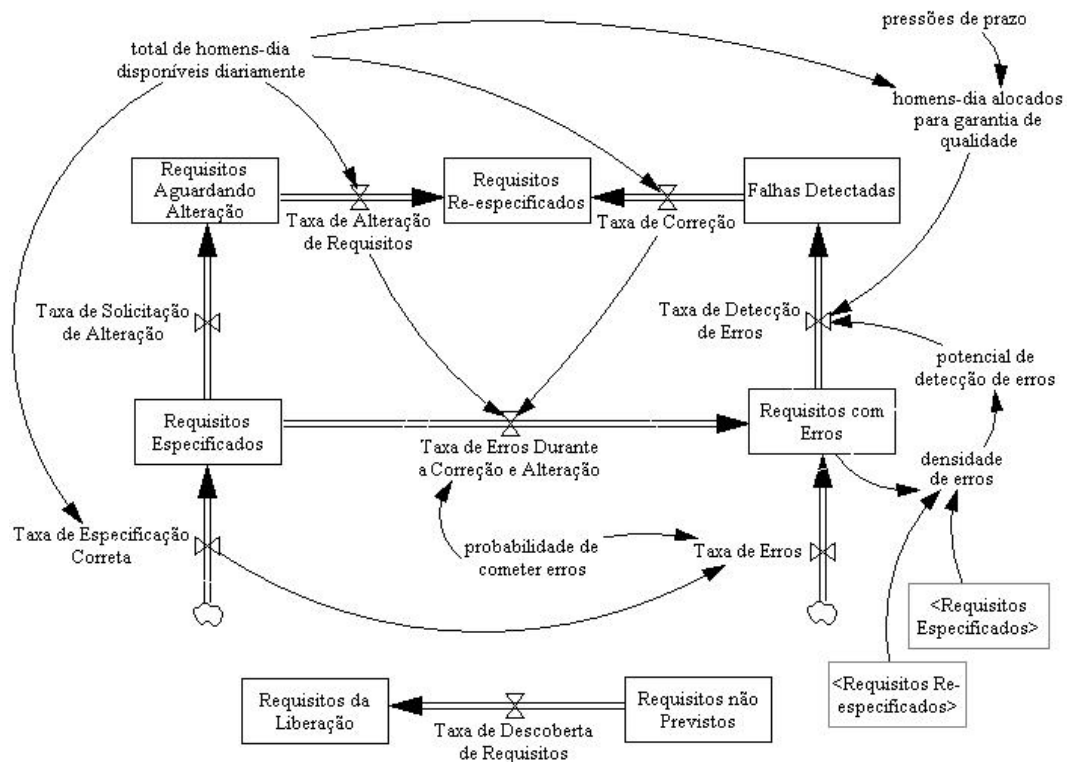
Ao utilizar as versões parciais do sistema, os usuários podem encontrar defeitos e solicitar reparos e/ou alterações nos requisitos implementados e entregues em outras liberações. A gerência deverá alocar recursos para atender a essas solicitações, o que poderá causar impactos no orçamento e no prazo do projeto. A porção do modelo que descreve os estados dos requisitos entregues em liberações anteriores é apresentada na seção 3.5.3.

3.5.2 Alocação de esforço para as atividades

No decorrer da especificação dos requisitos, os recursos disponíveis devem ser alocados para a realização de atividades como detectar erros, especificar novos requisitos e outras que serão apresentadas adiante. A Figura 3.6 apresenta uma porção do modelo que descreve como a alocação de esforço para a realização dessas atividades influencia o estado ou condição de cada requisito.

A quantidade de esforço, medido em homens-dia, disponibilizado diariamente pela equipe para a realização das atividades durante a especificação dos requisitos é

representada pela variável *total de homens-dia disponíveis diariamente*. O valor dessa variável é determinado pela quantidade de profissionais na equipe e pela quantidade de homens-dia disponibilizados por cada profissional. Essas variáveis serão apresentadas na seção 3.5.7.



Fonte: Elaborado pelo Autor

Figura 3.6. Alocação de esforço para as atividades

No modelo apresentado na Figura 3.6, a quantidade de esforço disponibilizado será alocada para a realização das atividades realizadas durante a especificação dos requisitos. Essas atividades são representadas pelos seguintes fluxos:

- *Taxa de Detecção de Erros*: representa as atividades de garantia de qualidade como revisões e inspeções que têm o objetivo de detectar erros cometidos durante a especificação;
- *Taxa de Correção*: representa a correção das falhas detectadas pelas atividades de garantia de qualidade;
- *Taxa de Alteração de Requisitos*: representa a alteração dos requisitos conforme as solicitações do cliente;
- *Taxa de Especificação Correta*: representa a especificação dos requisitos solicitados pelo cliente.

As inspeções e revisões dos artefatos produzidos durante a especificação dos requisitos visam a evitar a propagação de defeitos ao longo do projeto, minimizando os custos para os reparos (KIRNER; ABIB, 1997). A quantidade de esforço alocado para as atividades de garantia de qualidade é representada pela variável *homens-dia alocados para garantia de qualidade*. O valor dessa variável, junto com o valor da variável *potencial de detecção de erros*, irá determinar o valor do fluxo *Taxa de Detecção de Erros*. Esse fluxo representa a taxa com que os erros são detectados pela equipe e é responsável por transferir requisitos do estoque *Requisitos com Erros* para o estoque *Falhas Detectadas*. Esses relacionamentos podem ser visualizados na Figura 3.6.

A densidade de erros é definida pela relação entre a quantidade de erros inseridos no sistema e o tamanho do sistema (HENDIS, 1981; BASILI; BRIAND; MELO, 1996). Esse trabalho adapta essa definição para a fase de requisitos, considerando que a densidade de erros corresponde à relação entre a quantidade de erros não descobertos e a quantidade de requisitos especificados. O valor da variável *densidade de erros* é definido a partir dos valores dos estoques *Requisitos com Erros*, *Requisitos Especificados* e *Requisitos Re-especificados*. Como pode ser visto na Figura 3.7, considerando uma determinada quantidade de requisitos especificados, quanto maior for o valor do estoque *Requisitos com Erros*, maior será o valor da variável *densidade de erros*.

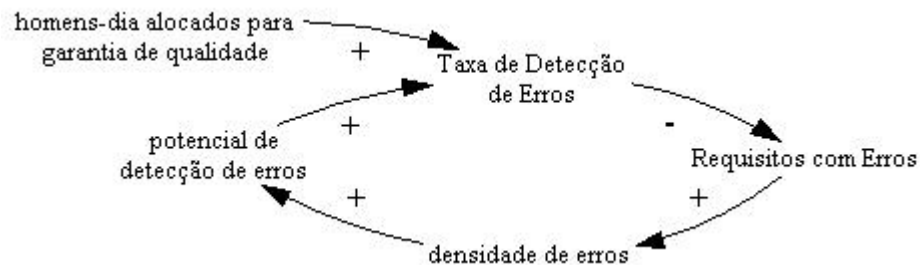
Definição conceitual		Representação no modelo
↓ Quantidade de erros não descobertos		↓ <i>Requisitos com Erros</i>
$densidade\ de\ erros = \frac{\text{Quantidade de requisitos especificados}}{\text{Quantidade de requisitos especificados}} = \frac{\text{Requisitos com Erros}}{\text{Requisitos Especificados} + \text{Requisitos Re-especificados}}$		

Fonte: Elaborado pelo Autor

Figura 3.7. Cálculo do valor da variável *densidade de erros*

A densidade de erros no projeto afeta o potencial de detecção de erros da equipe (LIN; ABDEL-HAMID; SHERIF, 1997). Quando a densidade de erros aumenta/diminui, o potencial de detecção de erros também aumenta/diminui. Por exemplo, à medida que a densidade de erros diminui, torna-se mais trabalhoso detectar os erros cometidos, o que diminui o potencial de detecção de erros. Um

estudo sobre esse relacionamento é apresentado em (ABDEL-HAMID; MADNICK, 1991) e (LIN; ABDEL-HAMID; SHERIF, 1997). Nos modelos apresentados nas Figuras 3.6 e 3.8, essa relação é representada pela influência que a variável *densidade de erros* exerce sobre a variável *potencial de detecção de erros*.



Fonte: Elaborado pelo Autor

Figura 3.8. Relação entre a densidade de erros e o potencial de detecção de erros da equipe

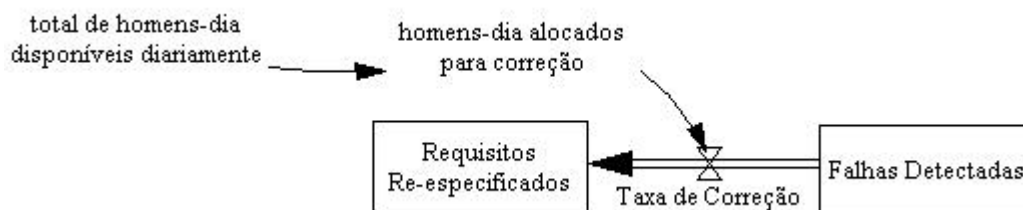
Como pode ser visto na Figura 3.8, a variável *potencial de detecção de erros* também afeta, indiretamente, a variável *densidade de erros*, formando um ciclo. Nesse caso, o relacionamento entre as variáveis é inverso, ou seja, quando o valor de uma aumenta/diminui, o valor da outra diminui/aumenta. Ao aumentar o valor da variável *potencial de detecção de erros*, o valor do fluxo *Taxa de Detecção de Erros* também aumenta, o que diminui o valor do estoque *Requisitos com Erros*, e por fim, diminui também o valor da variável *densidade de erros*.

Abdel-Hamid; Madnick (1991) e Abdel-Hamid; Sengupta; Swett (1999) ressaltam que quando o projeto está atrasado, ocorrem pressões de prazo e há uma tendência em diminuir a quantidade de esforço alocado para as atividades de garantia de qualidade. Os recursos são então concentrados no progresso do projeto, ou seja, na especificação de novos requisitos. O modelo representa esse fato por meio da influência da variável *pressões de prazo* sobre a variável *homens-dia alocados para garantia de qualidade*. As pressões de prazo também podem surgir devido ao *time to market*, ou seja, devido à necessidade de inserir um produto novo no mercado antes da concorrência, quando a empresa estiver inserida nesse tipo de contexto.

Devido às dependências entre requisitos, modificações em um determinado requisito podem afetar os demais (DAHLSTEDT; PERSSON, 2003). Após alocar uma quantidade de esforço para as atividades de garantia de qualidade, a alocação de esforço para a correção de falhas detectadas deve receber prioridade. Isso irá evitar que, caso exista uma dependência entre requisitos (DAHLSTEDT; PERSSON,

2003), um novo requisito seja especificado tomando como base um requisito com erro.

A quantidade de esforço alocado para corrigir as falhas detectadas irá influenciar o valor do fluxo *Taxa de Correção*. Esse fluxo representa a taxa com que as falhas descobertas são corrigidas pela equipe e é responsável por transferir requisitos do estoque *Falhas Detectadas* para o estoque *Requisitos Re-especificados*. A Figura 3.9 destaca a porção do modelo que contém esse relacionamento.



Fonte: Elaborado pelo Autor

Figura 3.9. Correção de erros

Durante a especificação dos requisitos e até mesmo em outras fases do projeto, os clientes podem solicitar alterações nos requisitos. Cabe à equipe, junto com o cliente, verificar a viabilidade dessas alterações. O valor do fluxo *Taxa de Solicitação de Alteração* corresponde à taxa com que as alterações de requisitos são solicitadas pelo cliente. Esse fluxo transfere requisitos acumulados no estoque *Requisitos Especificados* para o estoque *Requisitos Aguardando Alteração*. Os requisitos contidos no estoque *Requisitos Aguardando Alteração* deverão em algum momento ser modificados pela equipe.

Após alocar esforço para corrigir erros, a alocação de esforço para realizar alterações deve receber prioridade. Isso irá evitar que, caso exista uma dependência entre requisitos (DAHLSTEDT; PERSSON, 2003), um novo requisito seja especificado tomando como base um requisito que deverá ser alterado posteriormente.

A quantidade de esforço alocado para realizar alterações de requisitos irá afetar o valor do fluxo *Taxa de Alteração de Requisitos*. Esse fluxo representa a taxa com que os requisitos são alterados pela equipe e é responsável por transferir requisitos do estoque *Requisitos Aguardando Alteração* para o estoque *Requisitos Re-especificados*.

A Figura 3.10 ressalta a diferença entre os dois fluxos descritos anteriormente: enquanto o fluxo *Taxa de Solicitação de Alteração* representa a taxa com que as

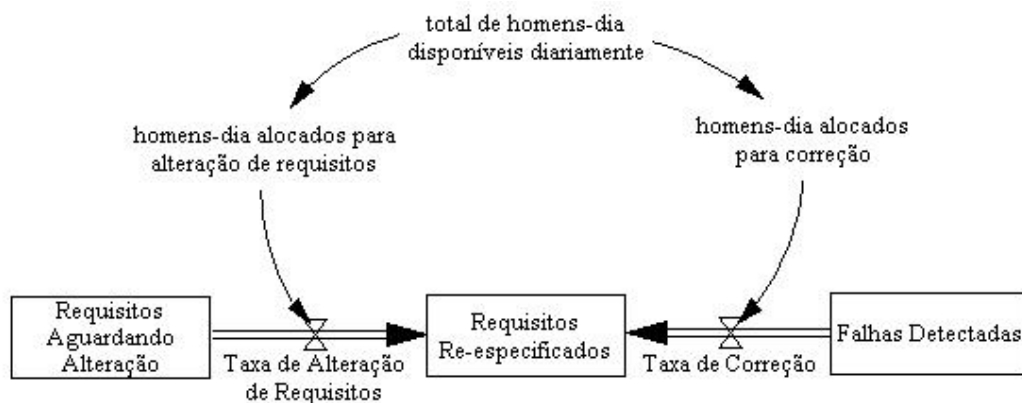
alterações de requisitos são solicitadas pelo cliente, o fluxo *Taxa de Alteração de Requisitos* representa a taxa com que os requisitos são efetivamente modificados.



Fonte: Elaborado pelo Autor

Figura 3.10. Alteração de requisitos

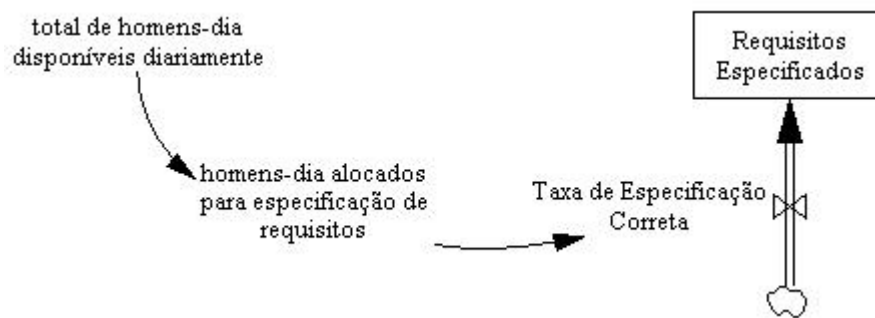
O estoque *Requisitos Re-especificados* armazena todos os requisitos que foram corrigidos ou alterados no decorrer da especificação dos requisitos. As correções e as alterações de requisitos constituem um retrabalho que aumenta tanto o custo final do projeto quanto os riscos de novas falhas serem introduzidas durante as próprias correções e alterações. O valor armazenado no estoque *Requisitos Re-especificados* permite à gerência verificar a quantidade de retrabalho gerado ao longo da especificação. Os fluxos que acumulam requisitos nesse estoque podem ser visualizados na Figura 3.6 e são apresentados também na Figura 3.11.



Fonte: Elaborado pelo Autor

Figura 3.11. Estoque *Requisitos Re-especificados*

A quantidade de esforço alocado para especificar novos requisitos irá determinar o valor do fluxo *Taxa de Especificação Correta*. Esse fluxo representa a taxa com que novos requisitos são especificados corretamente pela equipe e é responsável por acumular requisitos no estoque *Requisitos Especificados*. A Figura 3.12 ressalta a porção do modelo que contém esse relacionamento.

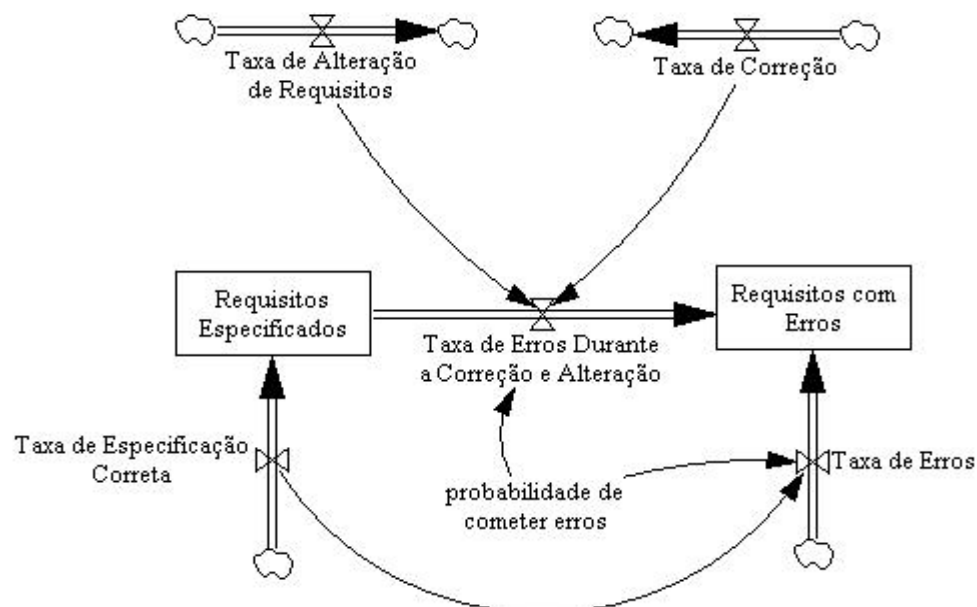


Fonte: Elaborado pelo Autor

Figura 3.12. Especificação de novos requisitos

Erros também são cometidos durante o ciclo de vida do desenvolvimento de software (LIN; ABDEL-HAMID; SHERIF, 1997; ABDEL-HAMID; MADNICK, 1991). O fluxo *Taxa de Erros* representa a taxa com que os erros são cometidos durante a especificação de novos requisitos. E o fluxo *Taxa de Erros Durante a Correção e Alteração* representa a taxa com que os erros são cometidos durante a correção e/ou alteração de requisitos.

Como pode ser visto no modelo apresentado na Figura 3.6 e na porção mostrada na Figura 3.13, os valores dos fluxos *Taxa de Erros* e *Taxa de Erros Durante a Correção e Alteração* são afetados pelo valor da variável *probabilidade de cometer erros*. Essa variável representa a probabilidade com que os erros são cometidos durante a realização das atividades na fase de requisitos e a determinação do seu valor será explicada na seção 3.5.8.



Fonte: Elaborado pelo Autor

Figura 3.13. Fluxos que representam os erros cometidos

Nessa seção foram apresentados os fluxos que representam as atividades realizadas durante a especificação dos requisitos: *Taxa de Detecção de Erros*, *Taxa de Correção*, *Taxa de Alteração de Requisitos* e *Taxa de Especificação Correta*. Os valores desses fluxos dependem também da produtividade média de cada membro da equipe ao realizar essas atividades.

A produtividade dos membros da equipe pode ser obtida por meio da manutenção de uma base de dados histórica contendo informações coletadas durante a realização dos projetos (MAXWELL; FORSELIUS, 2000; FILHO, 2003). A coleta de métricas acerca do desempenho da equipe nas atividades realizadas durante a especificação dos requisitos permite à gerência estimar a produtividade em projetos futuros.

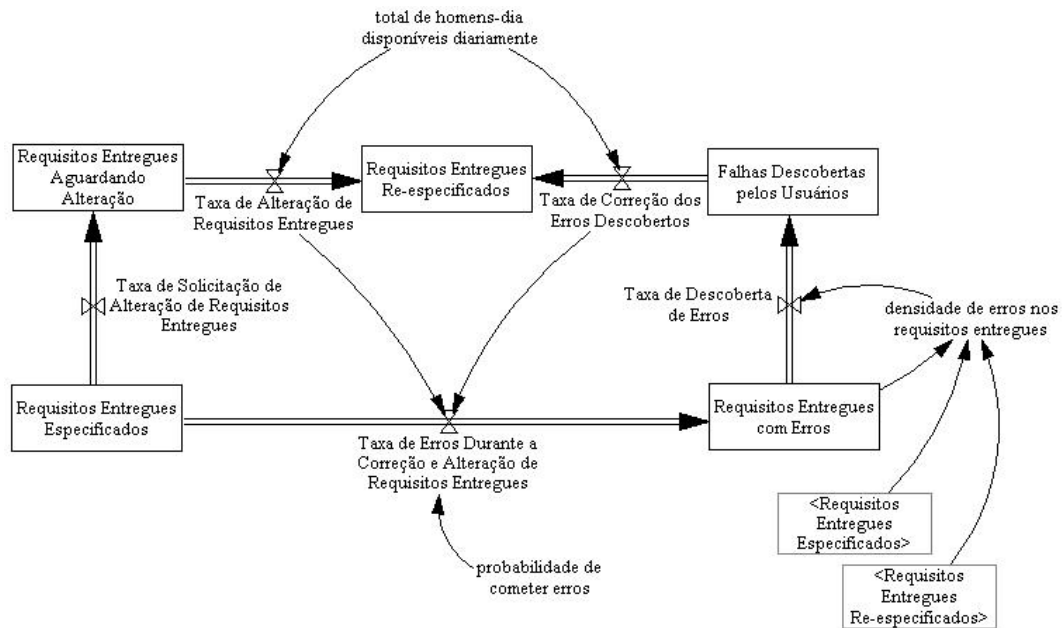
Uma alternativa para obter os dados sobre a produtividade é utilizar o PSP (*Personal Software Process*) (HUMPHREY, 2000a) que, embora seja utilizado para medir o desempenho de programadores durante a codificação do sistema, pode ser adaptado para medir a produtividade nas atividades realizadas no decorrer da especificação dos requisitos. O PSP abrange algumas medidas como taxa de erros cometidos, taxa de detecção de erros e taxa de remoção de erros que permitem à gerência conhecer a capacidade dos membros da equipe. O TSP (*Team Software Process*) (HUMPHREY, 2000b) guia a coleta de métricas no nível de equipes e permite à gerência conhecer a produtividade das equipes estabelecidas em uma organização.

3.5.3 Requisitos entregues em liberações anteriores

Em um processo de desenvolvimento iterativo, como o ciclo em espiral (BOEHM, 1988) e o RUP (*Rational Unified Process*) (KRUCHTEN, 2000), em que os requisitos do sistema são implementados e entregues em diversas liberações, os usuários começam a utilizar mais cedo as funcionalidades do sistema. Isso contribui para que eles forneçam, o quanto antes, um *feedback* acerca dos requisitos implementados (GREER; RUHE, 2004).

Ao utilizar as versões parciais do sistema, os usuários podem descobrir erros de requisitos ao perceberem que uma determinada funcionalidade não foi implementada da forma como foi solicitada. Esses erros serão reportados à equipe para que sejam reparados. Os usuários também podem solicitar alterações nos requisitos implementados quando alguma funcionalidade não satisfaz às suas expectativas.

As correções e as alterações solicitadas deverão ser entregues junto com os requisitos que serão implementados nas próximas liberações. A alocação de recursos para realizar essas atividades pode acarretar impactos no custo e no prazo estimados para as futuras liberações. A Figura 3.14 apresenta um modelo em que os estoques representam os estados dos requisitos entregues em liberações anteriores. Os fluxos presentes no modelo representam as atividades que podem alterar o estado desses requisitos.



Fonte: Elaborado pelo Autor

Figura 3.14. Estados dos requisitos entregues em liberações anteriores

Os estoques utilizados para representar os estados dos requisitos são:

- *Requisitos Entregues Especificados*: representa todos os requisitos que foram corretamente especificados e entregues em liberações anteriores;
- *Requisitos Entregues com Erros*: representa todos os requisitos que foram especificados incorretamente e entregues em liberações anteriores;
- *Falhas Descobertas pelos Usuários*: representa todos os requisitos, entregues em liberações anteriores, que apresentam erros descobertos pelos usuários;
- *Requisitos Entregues Aguardando Alteração*: representa todos os requisitos, entregues em liberações anteriores, que necessitam ser alterados para atender às solicitações do cliente;

- *Requisitos Entregues Re-especificados*: representa todos os requisitos entregues que foram re-especificados. A re-especificação é realizada quando é necessário corrigir ou alterar algum requisito entregue.

Os fluxos presentes no modelo são responsáveis por transferir requisitos de um estoque para o outro. Esses fluxos são:

- *Taxa de Descoberta de Erros*: representa a taxa com que os erros de requisitos são descobertos pelos usuários ao utilizarem o software;
- *Taxa de Correção dos Erros Descobertos*: representa a taxa com que os erros, descobertos e reportados pelos usuários, são corrigidos pela equipe;
- *Taxa de Erros Durante a Correção e Alteração de Requisitos Entregues*: representa a taxa com que os erros são cometidos durante a correção e/ou alteração de requisitos entregues em liberações anteriores;
- *Taxa de Solicitação de Alteração de Requisitos Entregues*: representa a taxa com que são solicitadas as alterações nos requisitos entregues em liberações anteriores;
- *Taxa de Alteração de Requisitos Entregues*: representa a taxa com que os requisitos entregues em liberações anteriores são alterados pela equipe.

No modelo, o valor do fluxo *Taxa de Erros Durante a Correção e Alteração de Requisitos Entregues* é determinado pelos valores da variável *probabilidade de cometer erros* e dos fluxos *Taxa de Correção dos Erros Descobertos* e *Taxa de Alteração de Requisitos Entregues*. Como foi informado na seção 3.5.2, a variável *probabilidade de cometer erros* representa a probabilidade com que os erros são cometidos durante a realização das atividades na fase de requisitos e a determinação do seu valor será explicada na seção 3.5.8

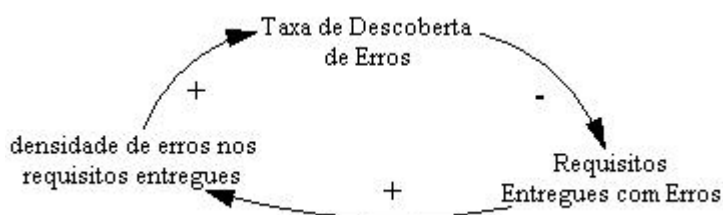
A densidade de erros (HENDIS, 1981; BASILI; BRIAND; MELO, 1996) nos requisitos entregues corresponde à relação entre a quantidade de erros não descobertos e a quantidade de requisitos entregues. O valor da variável *densidade de erros nos requisitos entregues* é definido a partir dos valores dos estoques *Requisitos Entregues com Erros*, *Requisitos Entregues Especificados* e *Requisitos Entregues Re-especificados*. Como pode ser visto na Figura 3.15, considerando uma determinada quantidade de requisitos entregues, quanto maior for o valor do estoque *Requisitos Entregues com Erros*, maior será o valor da variável *densidade de erros nos requisitos entregues*.

	Definição conceitual	Representação no modelo
	↓	↓
<i>densidade de erros nos requisitos entregues</i>	Quantidade de erros não descobertos Quantidade de requisitos especificados entregues	<i>Requisitos Entregues com Erros</i> <i>Requisitos Entregues Especificados + Requisitos Entregues Re-especificados</i>
=	=	

Fonte: Elaborado pelo Autor

Figura 3.15. Cálculo do valor da variável *densidade de erros nos requisitos entregues*

O valor da variável *densidade de erros nos requisitos entregues* influencia diretamente o valor do fluxo *Taxa de Descoberta de Erros*, ou seja, quando a densidade de erros aumenta/diminui, a taxa de descoberta de erros também aumenta/diminui. No entanto, existe um outro relacionamento entre essas variáveis. O valor do fluxo *Taxa de Descoberta de Erros* afeta, de modo inverso, o valor da variável *densidade de erros nos requisitos entregues*, ou seja, quando a taxa de descoberta de erros aumenta/diminui, a densidade de erros diminui/aumenta. Por exemplo, quando o valor do fluxo *Taxa de Descoberta de Erros* aumenta, o valor do estoque *Requisitos Entregues com Erros* diminui, o que diminui também o valor da variável *densidade de erros nos requisitos entregues*. Esses relacionamentos formam um ciclo e são apresentados na Figura 3.16.



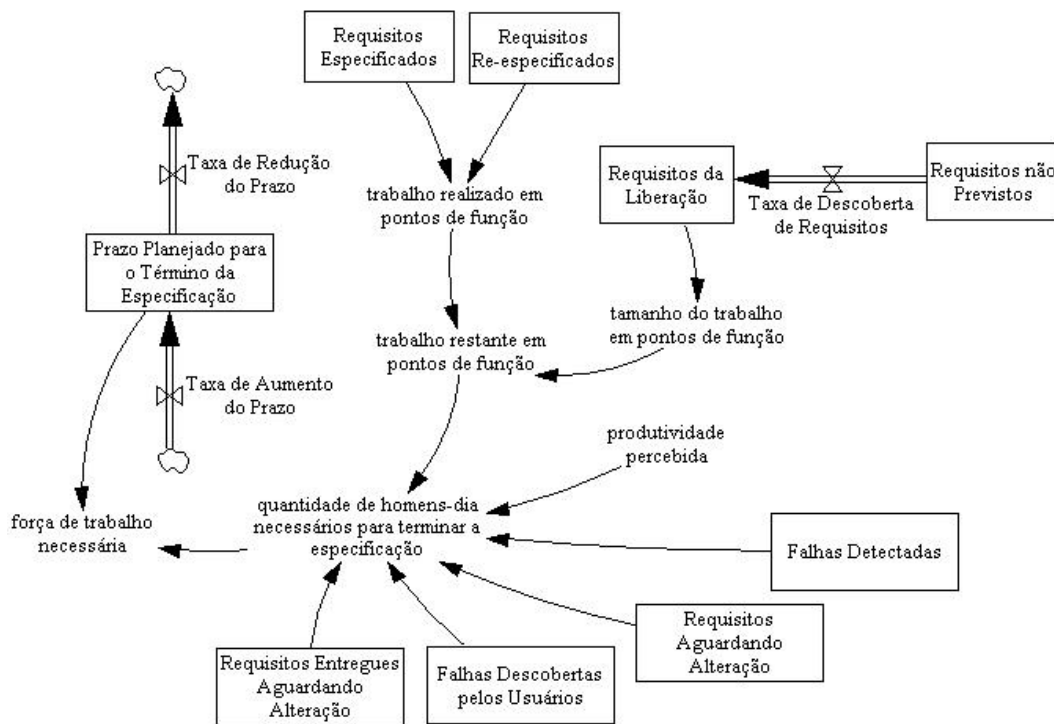
Fonte: Elaborado pelo Autor

Figura 3.16. Relação entre a densidade de erros nos requisitos entregues e a taxa com que os erros são descobertos pelos usuários

Os valores dos fluxos *Taxa de Correção dos Erros Descobertos* e *Taxa de Alteração de Requisitos Entregues* são influenciados pela quantidade de esforço medido em homens-dia alocados para realizar, respectivamente, a correção dos erros e as alterações nos requisitos entregues. A produtividade média dos membros da equipe também afeta os valores desses fluxos.

3.5.4 Quantidade de esforço necessário e definição do tamanho da equipe

A Figura 3.17 apresenta a porção do modelo que mostra as variáveis e as relações utilizadas na determinação da quantidade de esforço, medido em homens-dia, necessário para terminar a especificação. A partir da quantidade de esforço necessário e do prazo planejado para o término da especificação, é possível definir o tamanho da equipe para que a especificação termine no prazo planejado (ABDELHAMID, 1993).



Fonte: Elaborado pelo Autor

Figura 3.17. Quantidade de esforço necessário e definição do tamanho da equipe

O valor acumulado no estoque *Requisitos da Liberação* permite estimar o valor da variável *tamanho do trabalho em pontos de função*. Os valores dos estoques *Requisitos Especificados* e *Requisitos Re-especificados* permitem estimar o valor da variável *trabalho realizado em pontos de função*. O valor da variável *trabalho restante em pontos de função*, que representa a quantidade de trabalho necessário para especificar os requisitos restantes, é obtido a partir da diferença entre os valores das variáveis *tamanho do trabalho em pontos de função* e *trabalho realizado em pontos de função*.

A variável *quantidade de homens-dia necessários para terminar a especificação* corresponde ao esforço necessário para terminar a especificação dos

requisitos. Alguns trabalhos como (MATSON; BARRET; MELLICHAMP, 1994), (SHEPPERD; SCHOFIELD; KITCHENHAM, 1996) e (FILHO, 2003) relatam que o esforço necessário para o término de um projeto é determinado por duas variáveis: a quantidade de trabalho que ainda deve ser realizado e a produtividade percebida dos profissionais que compõem a equipe.

Conforme foi informado na seção 3.2, o modelo adota uma adaptação da fórmula apresentada em (FILHO, 2003) para calcular o esforço necessário para terminar a especificação. A partir do tamanho do trabalho medido em pontos de função e da produtividade da equipe, o esforço necessário medido em homens-dia é determinado.

Baseado na fórmula proposta por (FILHO, 2003), o modelo considera que o valor da variável *quantidade de homens-dia necessários para terminar a especificação* pode ser determinado pelo esforço necessário para: corrigir todas as falhas descobertas, realizar todas as alterações solicitadas pelo cliente e especificar os requisitos que ainda não foram especificados. No modelo apresentado na Figura 3.17, o valor dessa variável é determinado pelos valores das variáveis *trabalho restante em pontos de função* e *produtividade percebida*, e dos estoques *Falhas Detectadas*, *Requisitos Aguardando Alteração*, *Falhas Descobertas pelos Usuários* e *Requisitos Entregues Aguardando Alteração*.

O valor acumulado no estoque *Prazo Planejado para o Término da Especificação* corresponde ao prazo em dias estimado para o término da especificação dos requisitos. O valor inicial desse estoque corresponde ao prazo planejado antes do início da especificação dos requisitos. Ao longo da simulação do modelo, o valor armazenado nesse estoque pode ser modificado pelos fluxos *Taxa de Aumento do Prazo* e *Taxa de Redução do Prazo*, o que representa os ajustes feitos no prazo. Esses fluxos serão explicados na seção 3.5.6.

Conforme é afirmado em (ABDEL-HAMID, 1993), o tamanho adequado da equipe pode ser determinado a partir do prazo restante e da quantidade de esforço necessário para terminar o projeto. No modelo, os valores da variável *quantidade de homens-dia necessários para terminar a especificação* e do estoque *Prazo Planejado para o Término da Especificação* permitem determinar o valor da variável *força de trabalho necessária*. Essa variável representa a quantidade de pessoas que devem compor a equipe para que a especificação termine no prazo planejado.

Considerações sobre o nível de competência dos profissionais

Na seção 3.5.5 estabelece-se que os profissionais que compõem a equipe são divididos em dois grupos: membros novatos e membros experientes. Os membros novatos da equipe são aqueles que ingressaram no projeto há pouco tempo e por isso, possuem uma produtividade menor. Após trabalhar durante algum tempo no projeto, os membros novatos se tornam experientes.

Em uma empresa, o nível de competência dos profissionais pode variar. No entanto, por buscar a simplicidade e a facilidade de entendimento, o modelo assume que todos os profissionais possuem o mesmo nível de competência que é igual ao nível de competência médio de todos os profissionais da organização. Quando não ocorrem influências de alguns fatores (apresentados na seção 3.5.7) que podem afetar o desempenho da equipe, esse nível de competência médio permite que um membro experiente disponibilize um esforço igual a 1 homem-dia em cada dia de trabalho, ou seja, empregue seu esforço totalmente para as atividades que lhe são designadas. E, baseado nos modelos de (ABDEL-HAMID; MADNICK, 1991), o esforço disponibilizado por um membro novato é igual à metade do esforço disponibilizado por um membro experiente, ou seja, igual a 0,5 homens-dia.

Considerando a generalização para o nível de competência, a quantidade de profissionais necessários, representada pela variável *força de trabalho necessária*, se refere aos profissionais com nível de competência igual ao nível médio de todos os profissionais da organização. Em uma situação real, se somente profissionais menos capacitados (ou mais capacitados) estiverem disponíveis, poderá ser necessário alocar uma quantidade maior (ou menor) de pessoas.

Para classificar os profissionais quanto ao nível de competência, Alistair Cockburn (COCKBURN, 2001) definiu uma tabela com três níveis de competência que avalia a capacidade dos desenvolvedores em entender e seguir um método para o desenvolvimento de software. Posteriormente, Boehm e Turner (BOEHM; TURNER, 2003a, 2003b) particionaram o nível 1 nos níveis: -1, 1A e 1B. A tabela estendida de Boehm e Turner, apresentada na Tabela 3.1, utiliza cinco níveis para classificar a competência dos profissionais: -1, 1B, 1A, 2 e 3. O nível -1 corresponde aos profissionais menos capacitados e o nível 3 agrupa os profissionais mais capacitados. Não foi possível encontrar na literatura dados mais rigorosos utilizados para a obtenção da tabela. Entretanto, ela é aceita na comunidade dos adeptos dos Métodos Ágeis, onde é utilizada com sucesso.

Tabela 3.1. Nível de competência dos profissionais

3. Capaz de revisar um método (quebrar suas regras) para ajustá-lo a uma nova situação sem precedentes.
2. Capaz de construir um método que se ajuste a novas situações com precedentes.
1A. Com treinamento, é capaz de seguir os passos de um método discreto (dimensionar histórias para ajustar incrementos, definir padrões ou realizar refatorações complexas). Com experiência, pode alcançar o nível 2.
1B. Com treinamento, é capaz de seguir os passos de um método procedural (codificar um método simples, realizar refatorações simples, seguir padrões de codificação ou executar testes). Com experiência, pode atingir habilidades do nível 1A.
–1. Possui habilidades técnicas, mas não é capaz ou não está disposto a colaborar ou a seguir métodos compartilhados.

Fonte: Traduzido de (BOEHM; TURNER, 2003b)

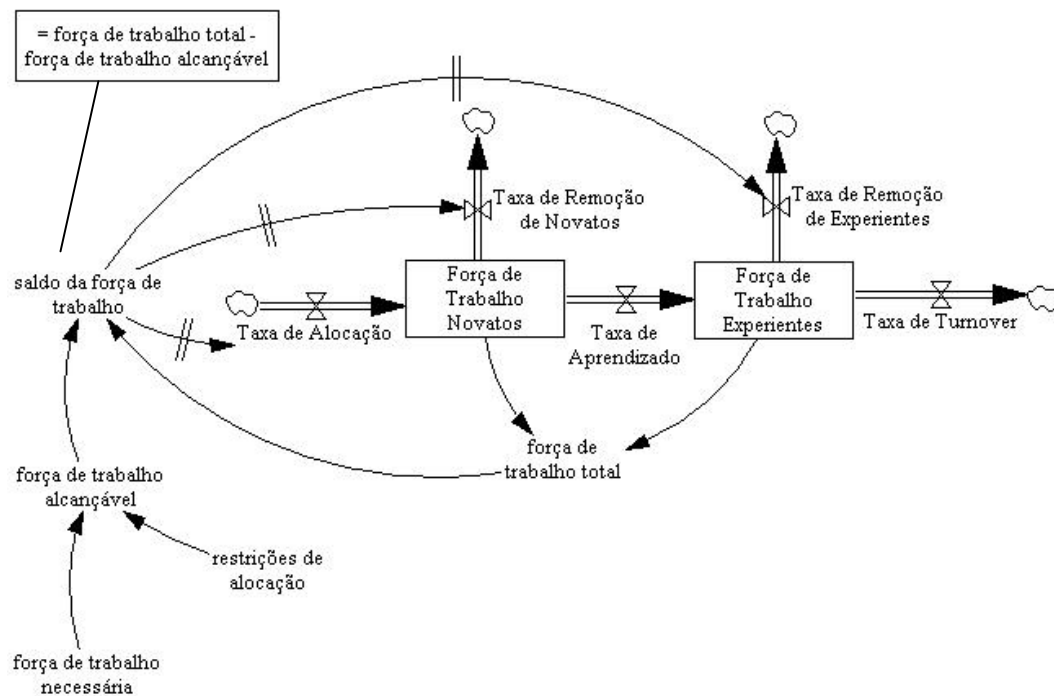
3.5.5 Controle do tamanho da equipe

O modelo apresentado na Figura 3.18 descreve como o tamanho da equipe pode ser modificado de acordo com as necessidades do projeto. Os membros que compõem a equipe são classificados em dois grupos: membros novatos e membros experientes, que são representados respectivamente pelos estoques *Força de Trabalho Novatos* e *Força de Trabalho Experientes*. A divisão da equipe em dois grupos é necessária porque os membros novatos, em geral, possuem uma produtividade menor e cometem mais erros que os membros experientes (LIN; ABDEL-HAMID; SHERIF, 1997).

Na seção 3.5.4 foi explicado que a quantidade necessária de pessoas na equipe para terminar a especificação no prazo planejado é dada pela variável *força de trabalho necessária*. Entretanto, se for preciso adicionar pessoas na equipe, a contratação não ocorre instantaneamente porque é necessário algum tempo para que novas pessoas sejam alocadas (SENGUPTA; ABDEL-HAMID; BOSLEY, 1999; ABDEL-HAMID; MADNICK, 1991).

Pode demorar algum tempo para a gerência remanejar profissionais de outras equipes ou contratar profissionais disponíveis no mercado. Além disso, de acordo com (ABDEL-HAMID; MADNICK, 1991), a quantidade máxima de novatos que podem existir na equipe pode ser limitada pela quantidade de membros experientes e

pela quantidade de novatos que cada membro experiente pode auxiliar por meio de *mentoring* ou acompanhamento. Esse limite existe para que os membros experientes não gastem um tempo excessivo treinando os membros novatos da equipe, o que prejudicaria o progresso da especificação. Essas restrições para a alocação de pessoas são representadas pela variável *restrições de alocação*.



Fonte: Elaborado pelo Autor

Figura 3.18. Controle do tamanho da equipe

Os valores das variáveis *força de trabalho necessária* e *restrições de alocação* determinam o valor da variável *força de trabalho alcançável* (SENGUPTA; ABDEL-HAMID; BOSLEY, 1999; ABDEL-HAMID; MADNICK, 1991). O valor dessa variável corresponde à quantidade de pessoas que podem constituir a equipe no momento, obedecendo às restrições impostas.

A partir dos valores dos estoques *Força de Trabalho Novatos* e *Força de Trabalho Experientes*, o valor da variável *força de trabalho total* é determinado. A diferença entre os valores das variáveis *força de trabalho total* e *força de trabalho alcançável* define o valor da variável *saldo da força de trabalho*. O valor dessa variável pode ser:

- **negativo:** significa que o tamanho da equipe é menor que o necessário. Por isso, é preciso alocar mais pessoas na equipe;

- **positivo:** significa que o tamanho da equipe é maior que o necessário. Por isso, é preciso remover pessoas da equipe;
- **zero:** significa que a equipe possui um tamanho adequado e não é preciso alocar nem remover pessoas da equipe.

Como pode ser visualizado na Figura 3.18, a variável *saldo da força de trabalho* afeta três fluxos presentes no modelo. Esses fluxos têm o objetivo de tornar o valor da variável *força de trabalho total* igual ao valor da variável *força de trabalho alcançável*, e são apresentados a seguir:

- *Taxa de Alocação:* representa a alocação de novas pessoas na equipe e aumenta o valor armazenado no estoque *Força de Trabalho Novatos*;
- *Taxa de Remoção de Novatos:* representa a remoção de membros novatos da equipe e diminui o valor armazenado no estoque *Força de Trabalho Novatos*;
- *Taxa de Remoção de Experientes:* representa a remoção de membros experientes da equipe e diminui o valor armazenado no estoque *Força de Trabalho Experientes*.

Se for necessário adicionar novos membros na equipe, o fluxo *Taxa de Alocação* será positivo. Mas, se for necessário remover membros da equipe, os fluxos *Taxa de Remoção de Novatos* e *Taxa de Remoção de Experientes* serão positivos. Inicialmente, serão removidos os membros novatos (ABDEL-HAMID; MADNICK, 1991). Membros experientes serão removidos somente se ainda for necessário diminuir o tamanho da equipe.

Os membros novatos são aqueles que ingressaram no projeto há pouco tempo e podem ser oriundos de outras organizações ou de outros projetos dentro da própria empresa. Os novos membros da equipe requerem um período de adaptação (SUAREZ *et al.*, 2002). Durante as primeiras semanas de trabalho, os membros novatos adquirem conhecimento sobre o projeto e se familiarizam com o domínio do problema e com as ferramentas utilizadas (SENGUPTA; ABDEL-HAMID; BOSLEY, 1999). Após algum tempo de trabalho, esses membros alcançam o status de experientes.

O fluxo *Taxa de Aprendizado* representa o aprendizado que ocorre durante o trabalho e é responsável por transferir membros do estoque *Força de Trabalho Novatos* para o estoque *Força de Trabalho Experientes*. O valor desse fluxo é

determinado pela quantidade de tempo necessário para que um membro novato se torne um membro experiente.

As inovações da tecnologia da informação e a expansão da economia digital impõem desafios para as organizações manterem seus profissionais mais capacitados (STANDBRIDGE; AUTREY, 2001). As políticas de recursos humanos das empresas e o desequilíbrio entre a baixa disponibilidade e a crescente demanda de profissionais são fatores que influenciam os profissionais a migrarem para outras organizações (ANG; SLAUGHTER, 2004).

No modelo, o fluxo *Taxa de Turnover* representa o *turnover* entre os profissionais da equipe. Por buscar uma simplificação, o modelo assume que o *turnover* ocorre somente entre os membros experientes, que é o que causa maior impacto no processo. Uma justificativa para essa decisão é o fato de que os profissionais são classificados como novatos somente durante um pequeno intervalo de tempo. Após o tempo de aprendizado, eles se tornam membros experientes da equipe e permanecem nesse grupo até o término da especificação, ou até saírem da equipe, o que caracteriza a ocorrência do *turnover*.

3.5.6 Uso de esforço extra da equipe e ajustes no prazo

O modelo na Figura 3.19 apresenta os fatores que podem acarretar em uma nova estimativa do prazo planejado para o término da especificação e exigir que os membros da equipe trabalhem com mais empenho para disponibilizar esforço extra.

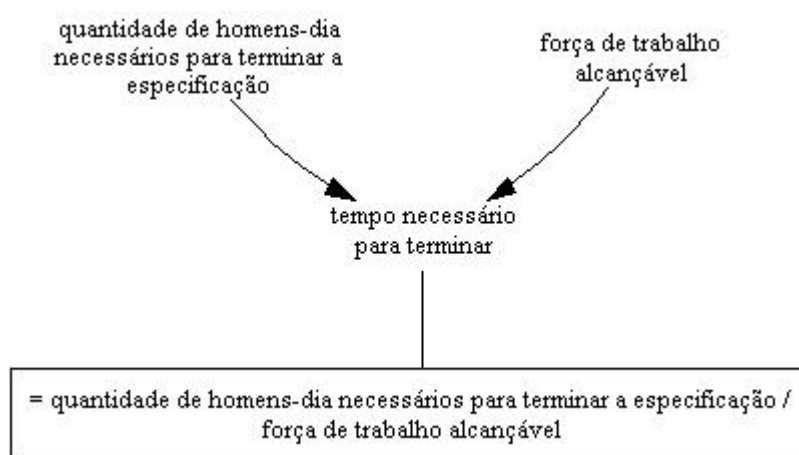
Conforme foi apresentado na seção 3.5.5, devido a algumas restrições (tempo para alocação de novos profissionais (SENGUPTA; ABDEL-HAMID; BOSLEY, 1999) e limite na quantidade de membros novatos na equipe (ABDEL-HAMID; MADNICK, 1991)), a quantidade de pessoas que podem compor a equipe é representada pela variável *força de trabalho alcançável*. Após a alocação ou a remoção de pessoas da equipe, o valor da variável *força de trabalho total* tende a ser igual ao valor da variável *força de trabalho alcançável*.

Uma análise dos modelos construídos em (ABDEL-HAMID; MADNICK, 1991) permite afirmar que a diferença entre os valores das variáveis *força de trabalho necessária* e *força de trabalho total* determina o valor da variável *falta de homens-dia percebida*. O valor dessa variável corresponde à quantidade de esforço, medido em homens-dia, que irá faltar para a realização das atividades, impedindo

alguns dias, aumentando a jornada de trabalho. Esse aumento na jornada pode ser recompensado com o pagamento de horas-extras ou com a criação de um banco de horas para os profissionais.

O incentivo ao esforço extra é representado pela influência que a variável *pressões de prazo* exerce sobre a variável *homens-dia extras disponibilizados por um membro experiente*. O valor dessa variável corresponde à quantidade de homens-dia obtidos com o esforço extra de um membro experiente da equipe. O valor análogo para os membros novatos será explicado na seção 3.5.7.

O planejamento do prazo é realizado continuamente e as estimativas feitas são revisadas ao longo da duração do projeto (ABDEL-HAMID, 1990). No modelo apresentado na Figura 3.19, a variável *tempo necessário para terminar* representa a quantidade de dias necessários para o término da especificação. A Figura 3.20 mostra como o valor dessa variável é determinado a partir dos valores das variáveis *quantidade de homens-dia necessários para terminar a especificação* e *força de trabalho alcançável*.



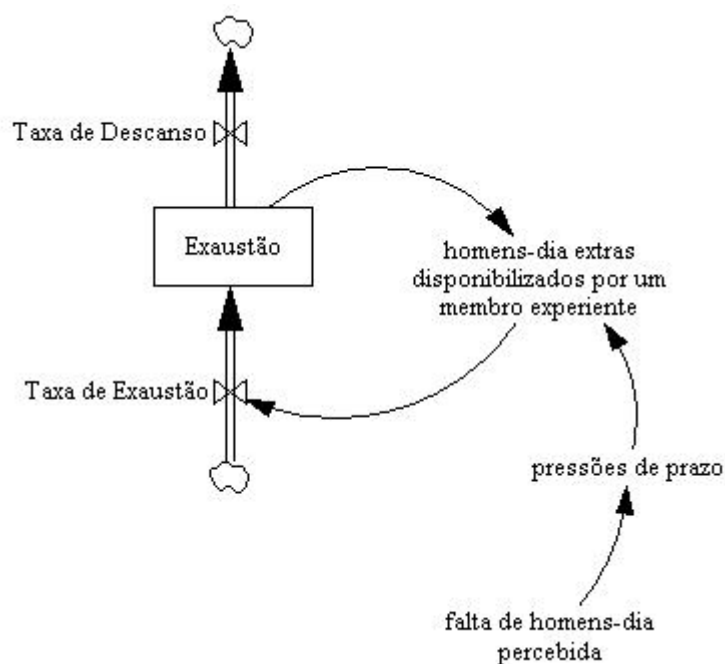
Fonte: Elaborado pelo Autor

Figura 3.20. Cálculo do valor da variável *tempo necessário para terminar*

A diferença entre os valores da variável *tempo necessário para terminar* e do estoque *Prazo Planejado para o Término da Especificação* determina o valor da variável *ajuste do prazo*. A reavaliação do prazo é representada pela influência que a variável *ajuste do prazo* exerce sobre os fluxos *Taxa de Aumento do Prazo* e *Taxa de Redução do Prazo*.

Ao utilizar esforço extra da equipe, deve ser considerado que há um limite para a sobrecarga de trabalho. Quando as pessoas trabalham sobrecarregadas durante um

longo tempo, ocorre exaustão da equipe e esse limite pode ser alcançado (COLLOFELLO *et al.*, 1998). O relacionamento entre esforço extra e exaustão da equipe é subjetivo e intangível. A modelagem e a quantificação desse relacionamento dependem do conhecimento acerca da disponibilidade da equipe em empregar esforço extra durante o trabalho. Além disso, o limite para a sobrecarga de trabalho pode variar entre organizações e até mesmo entre as equipes de uma mesma empresa. A Figura 3.21 exibe o relacionamento entre esforço extra e exaustão da equipe, modelado a partir das suposições apresentadas em (ABDEL-HAMID; MADNICK, 1991).



Fonte: Elaborado pelo Autor

Figura 3.21. Exaustão da equipe devido à sobrecarga de trabalho

A exaustão da equipe é representada pelo estoque *Exaustão*. Esse estoque possui o fluxo de entrada *Taxa de Exaustão* que é positivo quando os membros da equipe dedicam algum esforço extra para a especificação. O modelo assume que o valor desse fluxo é igual ao valor da variável *homens-dia extras disponibilizados por um membro experiente*. Se for utilizado 0,4 homens-dia extras dos membros experientes da equipe durante 5 dias, o valor acumulado no estoque *Exaustão* será 2.

À medida que o valor armazenado no estoque *Exaustão* aumenta e se aproxima do limite máximo suportado pela equipe, as pessoas começam a rejeitar a sobrecarga de trabalho (ABDEL-HAMID; MADNICK, 1991). Quando o limite é alcançado, ocorre uma queda na produtividade e ela permanece abaixo do normal até que os

profissionais estejam recuperados e possam trabalhar com alta produtividade novamente (HANNE; NEU, 2004; COLLOFELLO *et al.*, 1998). Isso é representado no modelo pelo relacionamento entre o estoque *Exaustão* e a variável *homens-dia extras disponibilizados por um membro experiente*, que recebe um valor negativo quando o nível de exaustão alcança o limite máximo.

O fluxo *Taxa de Descanso* corresponde ao descanso da equipe e é responsável por diminuir o valor acumulado no estoque *Exaustão*. O valor desse fluxo, que é positivo quando a equipe não dedica nenhum esforço extra, depende da taxa com que a exaustão da equipe diminui. Ao simular o modelo utilizando o valor default do fluxo *Taxa de Descanso*, é considerado que após o valor do estoque *Exaustão* atingir o limite máximo, demora em média 6 ou 7 dias para que seu valor diminua até 0 (zero). No entanto, o valor desse fluxo pode ser modificado, se desejado.

Quando o prazo para o término da especificação é superestimado e/ou o tamanho da equipe torna-se maior que o necessário, a equipe percebe que o projeto está adiantado em relação ao planejamento. No modelo apresentado na Figura 3.19, isso é representado pela atribuição de um valor negativo à variável *falta de homens-dia percebida* e/ou por uma diminuição no valor da variável *tempo necessário para terminar*.

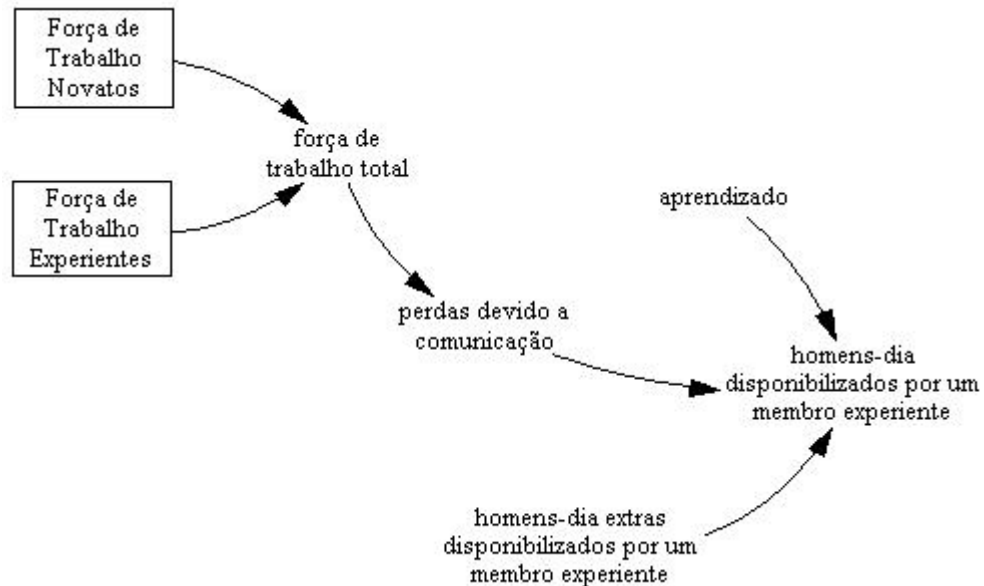
Ao perceber a folga no prazo, a equipe irá trabalhar com menor produtividade, absorvendo uma porção do excesso de tempo disponível. A quantificação da folga absorvida pela equipe é explicada em (ABDEL-HAMID; MADNICK, 1991). O restante do excesso de prazo será reportado à gerência que irá reduzir o prazo estimado para o término da especificação.

No modelo, a absorção do excesso de prazo por meio de uma queda na produtividade é representada pela atribuição de um valor negativo à variável *homens-dia extras disponibilizados por um membro experiente*. A redução do prazo para o término da especificação é representada pelo relacionamento entre a variável *ajuste do prazo* e o fluxo *Taxa de Redução do Prazo*.

3.5.7 Quantidade de esforço disponibilizado pela equipe

A Figura 3.22 mostra uma porção do modelo que apresenta alguns fatores que podem afetar a quantidade de esforço medido em homens-dia disponibilizado pelos membros da equipe. A variável *homens-dia disponibilizados por um membro experiente* representa a quantidade de esforço que um membro experiente da equipe

dedica à especificação em um dia de trabalho. Quando não ocorre nenhuma influência externa, supõe-se que cada membro experiente dedica 1 homem-dia em cada dia de trabalho, ou seja, emprega seu esforço totalmente para as atividades que lhe são designadas.



Fonte: Elaborado pelo Autor

Figura 3.22. Quantidade de esforço disponibilizado por um membro experiente

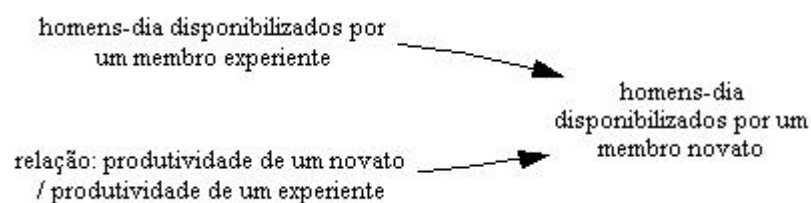
No entanto, alguns fatores podem influenciar o desempenho da equipe (ABDEL-HAMID, 1996), aumentando ou diminuindo a quantidade efetiva de esforço disponibilizado por cada membro. Esses fatores correspondem às seguintes variáveis:

- *Aprendizado*: à medida que as pessoas trabalham em um projeto, ocorre um aprendizado sobre o sistema e o domínio do problema, o que aumenta a produtividade da equipe (ABDEL-HAMID, 1996). Esse aumento da produtividade pode ser modelado como um aumento da quantidade de esforço disponibilizado. No contexto do desenvolvimento de software, o aumento do desempenho da equipe pode ser explicado pela curva de aprendizado (RACCOON, 1996);
- *Perdas devido à comunicação*: quanto maior for o tamanho da equipe, maior será o tempo gasto com comunicação e reuniões envolvendo os membros da equipe (SIMMONS, 1991), o que diminui a quantidade de esforço utilizado nas atividades de especificação;
- *Homens-dia extras disponibilizados por um membro experiente*: conforme foi apresentado na seção 3.5.6, quando o projeto está atrasado, os membros da equipe

são incentivados a trabalhar com mais empenho (ABDEL-HAMID, 1996; LIN; ABDEL-HAMID; SHERIF, 1997), aumentando a quantidade de esforço disponibilizado (ABDEL-HAMID, 1996).

A produtividade de um membro novato, que é menor que a de um membro experiente, deve ser avaliada porque é um fator que pode contribuir para o declínio na taxa com que as atividades são realizadas (ZHANG *et al.*, 2006). A produtividade dos membros novatos pode ser definida em função da produtividade dos membros experientes. Os modelos de (ABDEL-HAMID; MADNICK, 1991) consideram que a produtividade de um membro novato é igual à metade da produtividade de um membro experiente, enquanto que (MADACHY; TARBET, 2000) assume que os novatos têm produtividade igual a oitenta por cento da produtividade dos membros experientes. Para representar a menor produtividade dos membros novatos, o modelo descrito nesse trabalho considera que os membros novatos disponibilizam uma quantidade de esforço menor que a disponibilizada por um membro experiente.

No modelo, a quantidade de esforço disponibilizado por um membro novato é representada pela variável *homens-dia disponibilizados por um membro novato*. O valor dessa variável é determinado pelo valor da variável *homens-dia disponibilizados por um membro experiente* e pela relação entre a produtividade dos novatos e a dos experientes. Esse relacionamento é apresentado na Figura 3.23.

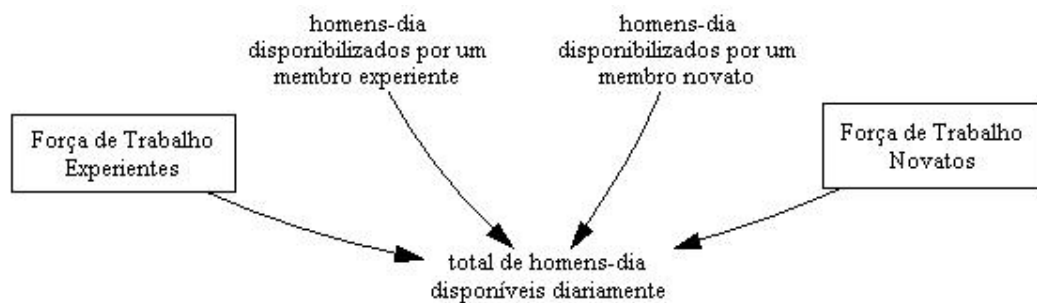


Fonte: Elaborado pelo Autor

Figura 3.23. Relação entre a produtividade dos novatos e a dos experientes

As variáveis do modelo estão ajustadas para que a quantidade de esforço disponibilizado por um membro novato seja sempre igual à metade do esforço disponibilizado por um membro experiente. Esse critério também foi adotado em (ABDEL-HAMID; MADNICK, 1991), que é um dos trabalhos mais referenciados, dentre aqueles que relatam a aplicação da dinâmica de sistemas na modelagem do gerenciamento de projetos de software. Quando não ocorre influência externa, o modelo assume que um membro novato dedica 0,5 homens-dia em cada dia de trabalho. No entanto, essa suposição pode ser modificada pelo usuário do modelo.

A Figura 3.24 ilustra como é calculada a quantidade de esforço disponibilizado diariamente para a realização das atividades durante a especificação dos requisitos. A partir dos valores dos estoques *Força de Trabalho Experientes* e *Força de Trabalho Novatos*, e das variáveis *homens-dia disponibilizados por um membro experiente* e *homens-dia disponibilizados por um membro novato*, o valor da variável *total de homens-dia disponíveis diariamente* é determinado.



Fonte: Elaborado pelo Autor

Figura 3.24. Cálculo do valor da variável *total de homens-dia disponíveis diariamente*

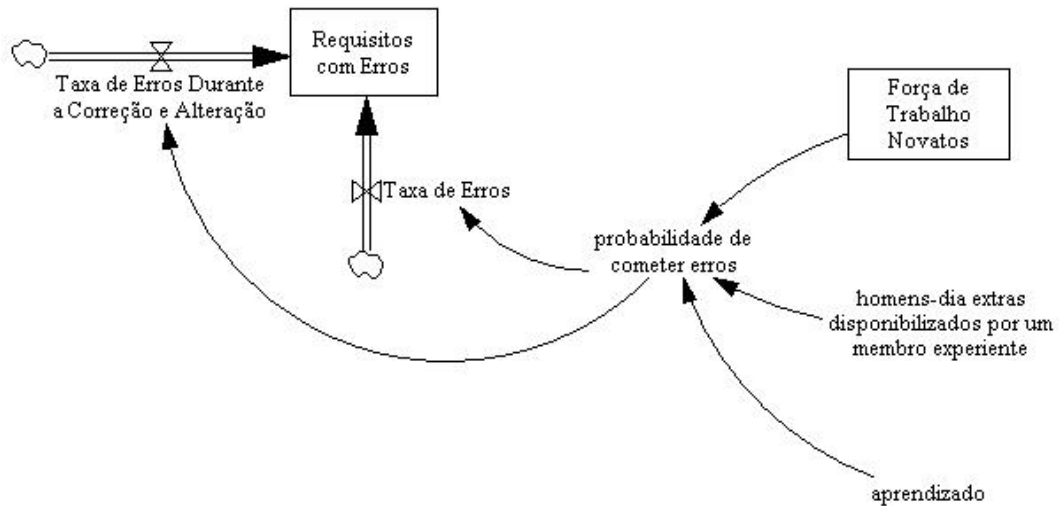
3.5.8 Geração de erros

Na porção do modelo apresentada na Figura 3.25, a variável *probabilidade de cometer erros* representa a probabilidade com que a equipe comete erros. O seu valor é determinado por alguns fatores que geralmente se mantêm constantes durante o projeto: dificuldade do trabalho, experiência e capacidade dos profissionais e ferramentas utilizadas. No entanto, outros fatores, cujas intensidades variam ao longo do tempo, também podem afetar a quantidade de erros cometidos. No modelo, o valor da variável *probabilidade de cometer erros* é afetado pelos valores do estoque *Força de Trabalho Novatos* e das variáveis *homens-dia extras disponibilizados por um membro experiente* e *aprendizado*.

Quanto maior for o valor da variável *homens-dia extras disponibilizados por um membro experiente*, maior será a quantidade de erros cometidos. A sobrecarga de trabalho acarreta estresse e exaustão da equipe, contribuindo para um aumento na quantidade de erros (COLLOFELLO *et al.*, 1998; LIN; ABDEL-HAMID; SHERIF, 1997).

O estoque *Força de Trabalho Novatos* também contribui para o aumento da quantidade de erros. Os membros novatos, em geral, cometem mais erros que os membros experientes (ABDEL-HAMID; MADNICK, 1991; LIN; ABDEL-HAMID; SHERIF, 1997). Baseando-se nos modelos dinâmicos apresentados em (ABDEL-

HAMID; MADNICK, 1991), o modelo descrito nesse trabalho considera que os membros novatos da equipe cometem duas vezes mais erros que os membros experientes. Por outro lado, o aprendizado que ocorre durante o projeto contribui para diminuir a quantidade de erros cometidos (HOUSTON *et al.*, 2001).

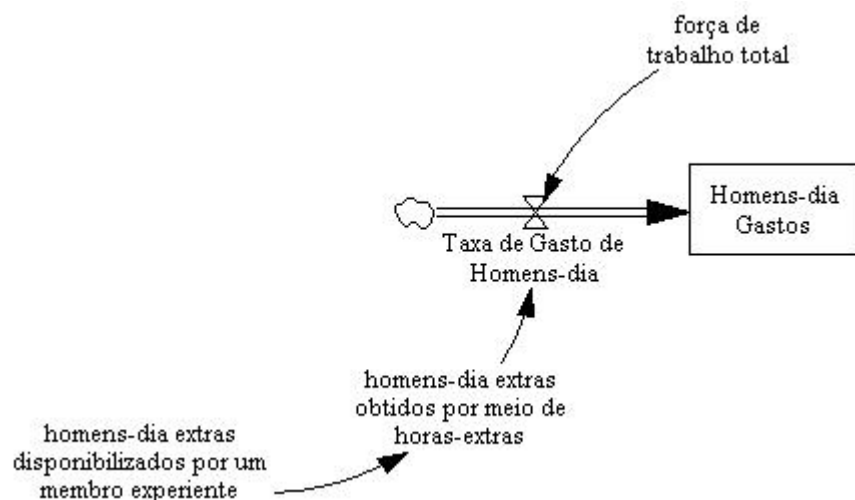


Fonte: Elaborado pelo Autor

Figura 3.25. Geração de erros

3.5.9 Custo para a realização da especificação dos requisitos

A Figura 3.26 contém a porção do modelo que permite obter o custo para a realização da especificação dos requisitos. O estoque *Homens-dia Gastos* armazena a quantidade de esforço, medido em homens-dia, utilizado até o momento. Esse estoque possui o fluxo de entrada *Taxa de Gasto de Homens-dia* que corresponde à quantidade de esforço utilizado por dia.



Fonte: Elaborado pelo Autor

Figura 3.26. Custo para a realização da especificação dos requisitos

O valor do fluxo *Taxa de Gasto de Homens-dia* é determinado pela quantidade de membros na equipe (variável *força de trabalho total*) e pela quantidade de esforço extra obtido por meio de um aumento na jornada de trabalho (variável *homens-dia extras obtidos por meio de horas-extras*).

A quantidade de homens-dia extras obtidos por meio de um maior esforço durante o horário normal de trabalho não contribui para o aumento do fluxo *Taxa de Gasto de Homens-dia*. Apenas os homens-dia extras obtidos por meio do aumento na jornada de trabalho constituem um custo adicional e irão, por tanto, provocar o aumento desse fluxo.

Como foi informado anteriormente, o Modelo Nível 0 contém todas as variáveis e relacionamentos do modelo. Devido à complexidade e à quantidade de variáveis em sua estrutura, o Modelo Nível 0 não é apresentado de forma detalhada nesse texto. No Capítulo 4 são apresentadas algumas de suas variáveis, que estão incluídas no painel de controle do modelo e são utilizadas para ajustar o modelo ao realizar as simulações.

4 USO DO MODELO PARA APOIO À TOMADA DE DECISÃO

O modelo construído pode ser utilizado como uma ferramenta para apoiar os gerentes na tomada de decisão durante a especificação dos requisitos. Ao simular o modelo, é possível investigar as conseqüências da materialização de riscos e os efeitos das decisões gerenciais planejadas. O modelo permite realizar experimentos controlados acerca da fase de requisitos em projetos de software, que são inviáveis de serem feitos em ambientes reais devido ao custo e ao tempo necessários.

No Capítulo 3 foi apresentada uma visão macro do modelo, contendo apenas as suas variáveis principais. Essa abordagem foi utilizada visando a facilitar a explicação e o entendimento acerca dos relacionamentos e da estrutura do modelo. O modelo completo, que pode ser utilizado para realizar simulações, apresenta outras variáveis que são utilizadas para ajustá-lo. Essas variáveis formam o painel de controle do modelo.

4.1 Painel de controle

O painel de controle do modelo é constituído por um conjunto de variáveis que permite aos gerentes ajustar o modelo de acordo com: o cenário que será simulado e as características da organização, da equipe e do projeto que definem o contexto da simulação. As variáveis contidas no painel de controle são classificadas em dois grupos: variáveis de configuração e variáveis de análise.

4.1.1 Variáveis de configuração

As variáveis de configuração são responsáveis pela portabilidade do modelo e permitem ajustá-lo de acordo com as características da organização, da equipe e do projeto que definem o contexto da simulação. Essas variáveis possuem um valor default que pode ser redefinido pelos usuários do modelo. O Quadro 4.1 lista as variáveis de configuração.

diminuição da produtividade após a exaustão: essa variável representa a quantidade de esforço que deixa de ser disponibilizado por um membro experiente, a partir do instante em que o nível de exaustão ultrapassa o máximo suportado e até o momento em que o nível de exaustão atinge o valor 0 (zero). Para determinar o valor análogo para um membro novato, é utilizado o valor da variável *relação: produtividade de um novato / produtividade de um experiente*.

máximo homens-dia não disponibilizados por um membro experiente: essa variável representa a quantidade máxima de esforço que um membro experiente deixa de disponibilizar quando é percebida uma folga no prazo planejado. Para determinar o valor análogo para um membro novato, é utilizado o valor da variável *relação: produtividade de um novato / produtividade de um experiente*.

máximo homens-dia extras disponibilizados sem custo adicional: essa variável representa a quantidade de esforço extra que um membro experiente disponibiliza por meio de um maior empenho durante o horário normal de trabalho, sem exigir o pagamento de horas-extras. Para determinar o valor análogo para um membro novato, é utilizado o valor da variável *relação: produtividade de um novato / produtividade de um experiente*.

multiplicador para a exaustão causada pelo aumento da jornada de trabalho: essa variável representa quantas vezes a exaustão causada por uma quantidade de esforço extra obtida por meio de horas-extras é maior que a exaustão causada por uma quantidade de esforço extra obtida por meio de um maior empenho durante o horário normal de trabalho. Essa variável permite considerar que o aumento na jornada de trabalho causa um maior desgaste que o incentivo a um maior empenho durante o horário normal de trabalho.

nível máximo de exaustão suportada: essa variável representa a quantidade máxima de exaustão que a equipe suporta sem sofrer fadiga e estresse.

porcentagem do excesso de prazo absorvido: essa variável representa a porcentagem do excesso no prazo planejado que será absorvido pela equipe. Essa absorção ocorre por meio de uma queda na produtividade. O restante do excesso de prazo é reportado à gerência para que o prazo seja reduzido.

quantidade de alterações realizadas por homem-dia: essa variável representa a quantidade de alterações que um membro experiente da equipe realiza em um dia de trabalho.

quantidade de erros descobertos por dia: essa variável representa a quantidade de erros de requisitos que os usuários descobrem por dia, ao utilizar as versões do software entregues nas liberações anteriores. O valor definido para essa variável não considera os efeitos da densidade de erros nos requisitos entregues. Veja a variável-gráfico *relação entre densidade de erros nos requisitos entregues e quantidade de erros descobertos por dia* apresentada no Quadro 4.5, na seção 4.2.

quantidade de erros detectados por homem-dia: essa variável representa a quantidade de erros que um membro experiente da equipe detecta em um dia de trabalho. O valor definido para essa variável não considera os efeitos da densidade de erros. Veja a variável-gráfico *relação entre densidade de erros e quantidade de erros detectados por homem-dia* apresentada no Quadro 4.5, na seção 4.2.

Fonte: Elaborado pelo Autor

Quadro 4.1. Variáveis de configuração

quantidade de falhas corrigidas por homem-dia: essa variável representa a quantidade de falhas que um membro experiente da equipe corrige em um dia de trabalho.

quantidade de requisitos especificados por homem-dia: essa variável representa a quantidade de requisitos que um membro experiente da equipe especifica em um dia de trabalho.

relação: produtividade de um novato / produtividade de um experiente: essa variável representa a relação entre a produtividade dos membros novatos da equipe e a produtividade dos membros experientes. Por exemplo, se a produtividade dos novatos for igual à metade da dos experientes, o valor dessa variável deve ser igual a 0,5.

Requisitos da Liberação: o valor desse estoque corresponde à quantidade de requisitos que devem ser especificados e entregues na próxima liberação.

Requisitos Entregues com Erros: o valor desse estoque corresponde à quantidade de requisitos que foram especificados incorretamente e entregues em liberações anteriores. Ao utilizarem o software, os usuários podem descobrir alguns desses erros e solicitar reparos à equipe.

Requisitos Entregues Especificados: o valor desse estoque corresponde à quantidade de requisitos que foram corretamente especificados e entregues em liberações anteriores. Podem ser solicitadas alterações em alguns desses requisitos.

tamanho médio de cada requisito em pontos de função: essa variável permite calcular o tamanho ou quantidade de trabalho que deve ser realizado utilizando a medida pontos de função.

Fonte: Elaborado pelo Autor

Quadro 4.1. Variáveis de configuração (continuação)

4.1.2 Variáveis de análise

As variáveis de análise são utilizadas para determinar as regras que regem como as decisões gerenciais são tomadas e ajustar o modelo para simular a materialização de riscos. Após definir o valor dessas variáveis e executar o modelo, os resultados das simulações podem ser utilizados para verificar os impactos dos riscos e os efeitos das decisões gerenciais modeladas.

As variáveis de análise são classificadas em três grupos conforme o propósito de cada uma. O primeiro grupo corresponde às variáveis utilizadas para simular os riscos. O segundo grupo reúne as variáveis utilizadas para determinar as decisões e políticas gerenciais aplicadas no decorrer da especificação dos requisitos. E o terceiro grupo corresponde às variáveis que são analisadas após a simulação para verificar os impactos dos riscos e das decisões gerenciais sobre o custo, o tempo gasto e a qualidade do trabalho realizado durante a fase de requisitos. Os Quadros 4.2, 4.3 e 4.4 apresentam as variáveis de análise e o propósito de cada uma.

porcentagem de requisitos não previstos: essa variável representa a porcentagem dos requisitos da liberação que não são identificados inicialmente, permitindo simular o risco de subestimar a quantidade de requisitos da liberação. Quando isso ocorre, alguns requisitos serão descobertos somente após o início da especificação.

probabilidade de alteração de requisitos: essa variável permite simular uma alta taxa de alteração de requisitos.

probabilidade de alteração de requisitos entregues: essa variável permite simular uma alta taxa de alteração nos requisitos entregues em outras liberações.

probabilidade de cometer erros: essa variável representa a probabilidade com que os erros são cometidos durante a especificação de novos requisitos. O seu valor pode ser ajustado para simular um projeto em que há uma alta probabilidade de cometer erros durante a fase de requisitos. O aumento na quantidade de erros pode ocorrer devido a vários fatores: alocação de pessoas menos capacitadas na equipe, dificuldades inerentes do domínio do problema e baixa confiabilidade das ferramentas utilizadas. O valor definido para essa variável não considera os efeitos do aprendizado e da quantidade de esforço extra utilizado. Veja as variáveis-gráfico *relação entre porcentagem da especificação completada e quantidade de erros cometidos* e *relação entre esforço extra utilizado e quantidade de erros cometidos* apresentadas no Quadro 4.5, na seção 4.2.

atenuação para erros durante a correção e alteração: em geral, a probabilidade com que os erros são cometidos durante a re-especificação (correção e alteração) de requisitos é menor que a probabilidade com que os erros são cometidos durante a especificação de novos requisitos. Essa variável permite considerar essa suposição, de forma que seu valor define quantas vezes a probabilidade com que os erros são cometidos durante a re-especificação é menor que a probabilidade com que os erros são cometidos durante a especificação de novos requisitos. Por exemplo, se a probabilidade de cometer erros durante a especificação é 0,12 (12%), e o valor dessa variável é 2, então a probabilidade de cometer erros durante a re-especificação é 0,06 (6%).

Definindo o turnover: para definir o *turnover*, o painel de controle não disponibiliza uma variável que permite ao usuário definir a probabilidade ou a taxa de ocorrência de *turnover*. A taxa de *turnover* geralmente é medida ao ano e considera o *turnover* entre todos os profissionais da organização. A fase de requisitos costuma demorar apenas algumas semanas ou meses e o tamanho da equipe é pequeno em relação à quantidade de profissionais na organização. Para permitir a definição do *turnover*, o modelo disponibiliza as duas variáveis a seguir:

quantidade de profissionais que abandonam a equipe: essa variável permite definir a quantidade de profissionais que irão abandonar a equipe, caracterizando a ocorrência de *turnover*.

tempo em que ocorre turnover: essa variável permite definir o tempo, medido em dias (a partir do início da especificação), em que irá ocorrer a saída de profissionais da equipe.

Fonte: Elaborado pelo Autor

Quadro 4.2. Variáveis de análise para simular a ocorrência de riscos

força de trabalho reserva: essa variável representa a quantidade de pessoas que serão mantidas de reserva na equipe, até a ocorrência do *turnover*. Ela permite simular a decisão gerencial de manter a equipe com um tamanho maior que o necessário. Essa decisão pode ser utilizada para amenizar os efeitos do *turnover* durante o projeto.

máximo homens-dia extras disponibilizados por um membro experiente: essa variável representa a quantidade máxima de esforço extra que um membro experiente dedica ao trabalho quando há pressões de prazo. Ela permite definir a quantidade máxima de sobrecarga de trabalho que será imposta à equipe para tentar suprir a falta de homens-dia percebida e evitar o atraso do projeto. Para determinar o valor análogo para um membro novato, é utilizado o valor da variável *relação: produtividade de um novato / produtividade de um experiente*, apresentada no Quadro 4.1.

porcentagem da falta de homens-dia tratada com esforço extra: essa variável representa a porcentagem da falta de esforço, medido em homens-dia, que será tratada com o uso do esforço extra da equipe. Se o esforço extra utilizado não for suficiente, o prazo para o término da especificação pode ser alterado.

porcentagem desejada de homens-dia alocados para garantia de qualidade: essa variável define a porcentagem do esforço disponível que é alocado para as atividades de garantia de qualidade. O valor definido para essa variável não considera os efeitos das pressões de prazo. Veja a variável-gráfico *relação entre esforço extra utilizado e esforço alocado para garantia de qualidade* apresentada no Quadro 4.5, na seção 4.2.

Prazo Planejado para o Término da Especificação: o valor inicial desse estoque corresponde ao prazo planejado, medido em dias. Durante a simulação, o valor do estoque pode ser modificado pelos ajustes no prazo.

quantidade inicial de pessoas na equipe: o valor dessa variável corresponde à quantidade de pessoas alocadas na equipe para iniciar a especificação.

quantidade máxima de novatos que um membro experiente pode treinar: essa variável representa a quantidade máxima de novatos que um membro experiente pode treinar por meio de *mentoring* ou acompanhamento. Esse limite existe para que os membros experientes não gastem um tempo excessivo treinando os membros novatos da equipe, o que prejudicaria o progresso da especificação.

quantidade mínima de pessoas na equipe: essa variável representa a quantidade mínima de pessoas que devem ser mantidas na equipe.

tempo desejado para realizar as alterações: essa variável representa a quantidade de dias que serão utilizados para realizar as alterações de requisitos. O aumento no valor dessa variável corresponde a um gerenciamento que não prioriza a realização das alterações.

tempo desejado para realizar as correções: essa variável representa a quantidade de dias que serão utilizados para realizar as correções das falhas detectadas. O aumento no valor dessa variável corresponde a um gerenciamento que não prioriza a realização das correções.

Fonte: Elaborado pelo Autor

Quadro 4.3. Variáveis de análise para determinar as decisões e as políticas gerenciais

tempo para aumento do uso de esforço extra: essa variável representa a quantidade de dias necessários para que seja aumentada a sobrecarga de trabalho imposta à equipe. O valor dessa variável pode ser usado para modelar a decisão gerencial quanto a aumentar ou não o uso de esforço extra. O aumento no valor dessa variável corresponde a um gerenciamento mais inclinado a não aumentar a quantidade de esforço extra utilizado. Por exemplo, considere que está sendo utilizado 0,1 homens-dia extra da equipe, e a gerência percebe a necessidade de aumentar o esforço extra para 0,3 homens-dia. O aumento desejado é de $0,3 - 0,1 = 0,2$ homens-dia. Se o valor dessa variável for 4, a quantidade de esforço extra utilizado irá aumentar em $0,2 / 4 = 0,05$ homens-dia por dia, até que no quarto dia, a quantidade de esforço extra utilizado seja 0,3 homens-dia. As cinco variáveis a seguir se comportam de modo semelhante.

tempo para redução do uso de esforço extra: essa variável representa a quantidade de dias necessários para que seja reduzida a sobrecarga de trabalho imposta à equipe. O valor dessa variável pode ser usado para modelar a decisão gerencial quanto a reduzir ou não o uso de esforço extra. O aumento no valor dessa variável corresponde a um gerenciamento mais inclinado a não reduzir a quantidade de esforço extra utilizado.

tempo para aumento do prazo: essa variável representa a quantidade de dias necessários para que o prazo planejado seja aumentado. O valor dessa variável pode ser usado para modelar a decisão gerencial quanto a aumentar ou não o prazo. O aumento no valor dessa variável corresponde a um gerenciamento mais inclinado a não aumentar o prazo planejado.

tempo para redução do prazo: essa variável representa a quantidade de dias necessários para que o prazo planejado seja reduzido. O valor dessa variável pode ser usado para modelar a decisão gerencial quanto a reduzir ou não o prazo. O aumento no valor dessa variável corresponde a um gerenciamento mais inclinado a não reduzir o prazo planejado.

tempo para aumento do tamanho da equipe: essa variável representa a quantidade de dias necessários para que as pessoas sejam alocadas na equipe. O valor dessa variável pode ser usado para modelar a decisão gerencial quanto a aumentar ou não o tamanho da equipe. O aumento no valor dessa variável corresponde a um gerenciamento mais inclinado a não aumentar o tamanho da equipe.

tempo para redução do tamanho da equipe: essa variável representa a quantidade de dias necessários para que as pessoas sejam removidas da equipe. O valor dessa variável pode ser usado para modelar a decisão gerencial quanto a reduzir ou não o tamanho da equipe. O aumento no valor dessa variável corresponde a um gerenciamento mais inclinado a não reduzir o tamanho da equipe.

Fonte: Elaborado pelo Autor

Quadro 4.3. Variáveis de análise para determinar as decisões e as políticas gerenciais (continuação)

Homens-dia Gastos: após o término da simulação, o valor desse estoque corresponde à quantidade de esforço, medido em homens-dia, utilizado na fase de requisitos.

Prazo Planejado para o Término da Especificação: após o término da simulação, o valor acumulado nesse estoque corresponde ao tempo gasto para terminar a especificação.

Requisitos com Erros: após o término da simulação, o valor desse estoque corresponde à quantidade de requisitos com erros que serão entregues para as fases seguintes do processo de desenvolvimento. Os erros que não forem detectados e corrigidos serão entregues na próxima liberação. Ao utilizar as versões parciais do sistema, os usuários podem descobrir alguns desses erros e solicitar reparos à equipe.

Fonte: Elaborado pelo Autor

Quadro 4.4. Variáveis de análise para verificação após a simulação

O estoque *Prazo Planejado para o Término da Especificação* está incluído em dois grupos. Ele faz parte do grupo constituído pelas variáveis utilizadas para determinar as decisões e as políticas gerenciais porque o seu valor inicial corresponde ao prazo planejado para o término da especificação dos requisitos. E também faz parte do grupo formado pelas variáveis utilizadas para verificação após a simulação porque o valor final desse estoque, obtido após o término da simulação, corresponde ao tempo gasto para terminar a especificação dos requisitos.

4.2 Variáveis-gráfico

Alguns relacionamentos entre as variáveis do modelo são complexos e difíceis de serem definidos por meio de equações e operações matemáticas. Quando ao menos a forma da curva do gráfico que denota o relacionamento é conhecida, uma variável-gráfico pode ser utilizada para representá-lo. Variável-gráfico é um tipo de variável disponível nas ferramentas de Dinâmica de Sistemas. O seu valor é definido por meio de um esboço do gráfico correspondente à função matemática que quantifica o relacionamento.

O Quadro 4.5 apresenta as variáveis-gráfico presentes no modelo de dinâmica de sistemas construído.

diminuição da produtividade devido ao tempo gasto com comunicação: essa variável representa como o aumento da quantidade de pessoas na equipe (variável *força de trabalho total*) contribui para diminuir a produtividade dos membros experientes da equipe (variável *homens-dia disponibilizados por um membro experiente*). Para determinar o valor análogo para um membro novato, é utilizado o valor da variável *relação: produtividade de um novato / produtividade de um experiente* apresentada no Quadro 4.1, na seção 4.1.

relação entre densidade de erros e quantidade de erros detectados por homem-dia: essa variável representa como a densidade de erros (variável *densidade de erros*) afeta a quantidade de erros detectados pela equipe (variável *quantidade de erros detectados por homem-dia*).

relação entre densidade de erros nos requisitos entregues e quantidade de erros descobertos por dia: essa variável representa como a densidade de erros nos requisitos entregues (variável *densidade de erros nos requisitos entregues*) afeta a quantidade de erros descobertos pelos usuários, ao utilizar as funcionalidades do sistema entregues nas liberações anteriores (variável *quantidade de erros descobertos por dia*).

relação entre esforço extra utilizado e esforço alocado para garantia de qualidade: essa variável representa como a quantidade de esforço extra utilizado (variável *homens-dia extras disponibilizados por um membro experiente*), devido às pressões de prazo, afeta a quantidade de esforço alocado para as atividades de garantia de qualidade (variável *homens-dia alocados para garantia de qualidade*).

relação entre esforço extra utilizado e quantidade de erros cometidos: essa variável representa como a quantidade de esforço extra utilizado (variável *homens-dia extras disponibilizados por um membro experiente*), devido às pressões de prazo, afeta a quantidade de erros cometidos (variável *probabilidade de cometer erros*).

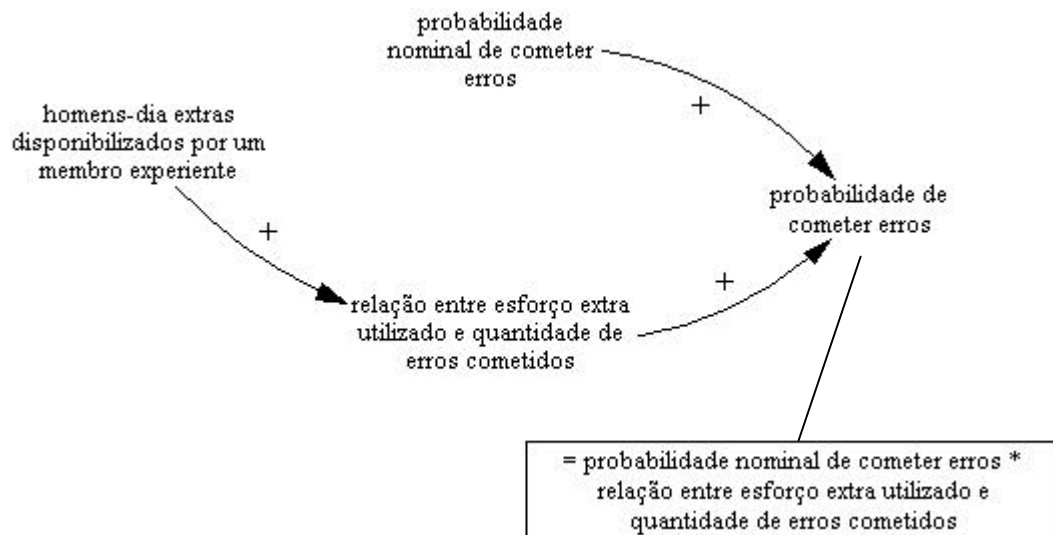
relação entre porcentagem da especificação completada e esforço disponibilizado por um membro experiente: essa variável representa como o aumento na porcentagem da especificação completada (variável *porcentagem da especificação completada*) é responsável por um aprendizado que provoca o aumento na quantidade de esforço disponibilizado por um membro experiente (variável *homens-dia disponibilizados por um membro experiente*). Para determinar o valor análogo para um membro novato, é utilizado o valor da variável *relação: produtividade de um novato / produtividade de um experiente* apresentada no Quadro 4.1, na seção 4.1.

relação entre porcentagem da especificação completada e quantidade de erros cometidos: essa variável representa como o aumento na porcentagem da especificação completada (variável *porcentagem da especificação completada*) é responsável por um aprendizado que provoca uma redução na quantidade de erros cometidos (variável *probabilidade de cometer erros*).

Fonte: Elaborado pelo Autor

Quadro 4.5. Variáveis-gráfico do modelo

A Figura 4.1 apresenta os relacionamentos em torno da variável-gráfico *relação entre esforço extra utilizado e quantidade de erros cometidos*.

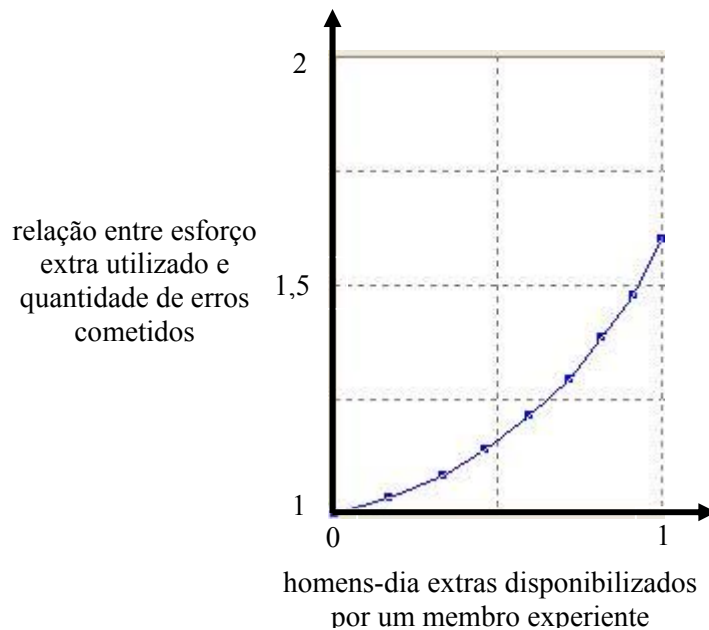


Fonte: Elaborado pelo Autor

Figura 4.1. Relacionamentos envolvendo a variável-gráfico *relação entre esforço extra utilizado e quantidade de erros cometidos*

A Figura 4.2 apresenta o esboço do gráfico, definido no modelo, para a variável *relação entre esforço extra utilizado e quantidade de erros cometidos*. De acordo com a curva do gráfico, quando o valor da variável de entrada *homens-dia extras disponibilizados por um membro experiente* é 0 (zero), o valor da variável-gráfico *relação entre esforço extra utilizado e quantidade de erros cometidos* é 1. À medida que o valor da variável de entrada aumenta, o valor da variável-gráfico aumenta exponencialmente, o que provoca um aumento no valor da variável *probabilidade de cometer erros*.

Os esboços dos gráficos de todas as variáveis-gráfico do modelo foram definidos a partir de informações encontradas na literatura. A forma da curva desses gráficos pode ser modificada, o que permite considerar e simular novas suposições para os relacionamentos representados.



Fonte: Adaptado de (ABDEL-HAMID; MADNICK, 1991)

Figura 4.2. Esboço do gráfico da variável *relação entre esforço extra utilizado e quantidade de erros cometidos*

4.3 Modelagem de cenários

Para direcionar a quantificação de algumas variáveis de análise foram delimitados intervalos de valores que classificam o valor das variáveis como baixo, médio ou alto. Os limites dos intervalos definidos podem ser visualizados na Tabela 4.1.

A partir dos intervalos definidos, é possível criar cenários para realizar as simulações. Por exemplo, os gerentes podem verificar o comportamento do modelo em um cenário de alto risco em que a taxa de alteração de requisitos (representada pela variável *probabilidade de alteração de requisitos*) e a probabilidade com que os erros são cometidos (representada pela variável *probabilidade de cometer erros*) são altas. Outra possibilidade é analisar um cenário otimista em que os valores das variáveis *probabilidade de alteração de requisitos* e *probabilidade de cometer erros* são baixos.

Entre as variáveis apresentadas na Tabela 4.1, apenas a variável *probabilidade de cometer erros* não teve seus intervalos de valores definidos a partir de informações encontradas na literatura. Durante as pesquisas realizadas, não foram encontrados trabalhos que relatam a quantidade ou a probabilidade com que os erros são cometidos durante a especificação dos requisitos de sistemas de software. Para

definir os intervalos de valores para a variável *probabilidade de cometer erros* foram utilizados dados coletados em uma empresa de desenvolvimento de software, com sede em Viçosa-MG. Ao responderem o questionário elaborado para realizar a coleta dos dados, os profissionais informaram que: o valor mais provável para essa variável é 15%, um valor baixo é 5% e um valor alto é 20%. Na seção 5.2 são apresentados detalhadamente o questionário utilizado para coletar os dados e os demais valores informados.

Tabela 4.1. Intervalos de valores para algumas variáveis de análise

Variáveis	Baixo	Médio	Alto
<i>probabilidade de cometer erros</i>	menor que 10%	entre 10 e 17%	maior que 17%
<i>probabilidade de alteração de requisitos (ao mês)</i>	menor que 3%	entre 3 e 8%	maior que 8%
<i>probabilidade de alteração de requisitos entregues (ao mês)</i>	menor que 3%	entre 3 e 8%	maior que 8%
<i>probabilidade de turnover (ao ano)</i>	menor que 12%	entre 12 e 20%	maior que 20%
<i>porcentagem de requisitos não previstos</i>	menor que 30%	entre 30 e 40%	maior que 40%
<i>porcentagem desejada de homens-dia alocados para garantia de qualidade</i>	menor que 11%	entre 11 e 20%	maior que 20%
<i>máximo homens-dia extras disponibilizados por um membro experiente</i>	menos que 0,2 homens-dia	entre 0,2 e 0,4 homens-dia	mais que 0,4 homens-dia

Fonte: Elaborado pelo Autor

Os intervalos de valores para a variável *probabilidade de cometer erros* foram definidos de modo que os valores intermediários aos três valores informados (5, 15 e 20%) fossem distribuídos igualmente entre os intervalos. Por exemplo, 5% é um valor baixo para a variável e 15% é um valor médio. Os valores intermediários (6 a 14%) foram distribuídos nos intervalos de valores baixo e médio, de modo que os valores a partir de 6 até 9% são considerados valores baixos e os valores a partir de 10 até 14% são considerados valores médios.

Com exceção da variável *probabilidade de cometer erros*, os intervalos de valores para as variáveis apresentadas na Tabela 4.1 foram estabelecidos a partir de informações encontradas na literatura. Em (JONES, 1996c) são apresentados dados sobre a taxa mensal de alteração de requisitos em vários tipos de projetos de

software. Kulk; Verhoef (2008) sintetizam os dados mais recentes obtidos nas pesquisas realizadas por Caper Jones (JONES, 1996c, 1998a, 1998b, 2000, 2001) com o objetivo de investigar a taxa mensal de alteração de requisitos na indústria de software. Os valores informados nesses trabalhos determinaram a escolha do intervalo de 3 a 8% para representar valores médios para as variáveis *probabilidade de alteração de requisitos* e *probabilidade de alteração de requisitos entregues*. Javed; Manzil-E-Maqsood; Durrani (2004) relatam que, segundo Caper Jones, o ideal é manter uma taxa de volatilidade de requisitos baixa, de até 3%. De acordo com (SUDHAKAR, 2005), uma taxa de alteração acima de 8% é considerada alta.

Ang; Slaughter (2004) investigam a taxa anual de *turnover* em vários cargos no setor de tecnologia da informação e informam que o valor médio dessa taxa para os cargos gerenciais é 6,27%; e para os cargos técnicos é 8,3%. Segundo (SIMMONS, 2003), uma taxa de *turnover* ideal está entre 12 e 20% e o seu valor não deveria exceder 30%. Em (DAYASINDHU, 2002), é informado que a indústria de software indiana enfrenta uma alta taxa de *turnover*, que está em torno de 20% ou mais. Os valores informados nos trabalhos analisados determinaram a escolha do intervalo de 12 a 20% para representar valores médios para a variável *probabilidade de turnover*.

Em (MOLOKKEN; JORGENSEN, 2003) é apresentada uma revisão acerca dos resultados de pesquisas sobre erros na estimativa do prazo e do esforço necessários em projetos de software. Entre as pesquisas consideradas estão as relatadas em (PHAN, 1990) e (BERGERON; ST-ARNAUD, 1992). O trabalho informa que, em média, a quantidade de esforço gasto é 30 a 40% maior que o planejado. Uma investigação realizada por (MOLOEKKEN-OESTVOLD *et al.*, 2004) sugere que esse percentual é 41%. Esses valores determinaram a escolha do intervalo de 30 a 40% para representar valores médios para a variável *porcentagem de requisitos não previstos*.

Abdel-Hamid; Madnick (1991) descrevem que a porcentagem dos recursos destinados às atividades de garantia de qualidade varia desde 6% até valores mais altos como 25%, e considera 15% um valor médio para essa variável. Os intervalos de valores para a variável *porcentagem desejada de homens-dia alocados para garantia de qualidade* foram definidos de modo que os valores intermediários aos valores informados em (ABDEL-HAMID; MADNICK, 1991) fossem distribuídos igualmente entre os intervalos. Por exemplo, 15% é um valor médio para a variável e 25% é um valor alto. Os valores intermediários (16 a 24%) foram distribuídos nos

intervalos de valores médio e alto, de modo que os valores a partir de 16 até 20% são considerados valores médios e os valores a partir de 21 até 24% são considerados valores altos.

A partir de um estudo de caso, Perlow (2001) afirma que diante de pressões de prazo, a duração da jornada de trabalho pode aumentar em até 50%, chegando a 12 horas diárias (assumindo uma jornada normal de 8 horas diárias). Em (TAPIA, 2004), o autor relata que durante sua permanência em uma empresa, também foi observada uma jornada que chegava a 12 horas quando havia risco de atraso no projeto.

O modelo dinâmico de (ABDEL-HAMID; MADNICK, 1991) assume que 0,4 homens-dia são gastos em atividades pessoais durante o trabalho e quando há pressões de prazo, o desperdício pode ser reduzido para 0,15 homens-dia. Isso significa que 0,25 homens-dia podem ser obtidos por meio de um maior empenho durante o horário normal de trabalho. Considerando ainda a possibilidade de um aumento de até 50% na jornada de trabalho, a quantidade total de esforço extra que pode ser disponibilizado é aproximadamente 0,6 homens-dia. A segmentação desse valor (0,6 homens-dia) em três intervalos de valores com o mesmo comprimento determinou os intervalos de valores para a variável *máximo homens-dia extras disponibilizados por um membro experiente*.

4.4 Análises gerenciais suportadas pelo modelo

O modelo de dinâmica de sistemas construído possibilita aos gerentes estudar e compreender as interações dinâmicas entre as variáveis envolvidas na fase de requisitos. As simulações com o modelo permitem aos gerentes testar e verificar as conseqüências da aplicação de determinadas políticas gerenciais em diversos cenários, sujeitos à ocorrência de vários tipos de risco. Diante da quantidade de variáveis e relacionamentos envolvidos, sem o uso de uma ferramenta desse tipo, torna-se muito difícil, senão impossível, para os gerentes entenderem e reproduzirem mentalmente o comportamento do sistema.

O modelo constitui uma ferramenta que suporta a realização de diversas análises gerenciais, que consistem em verificar ou examinar:

- os impactos de uma alta taxa de alteração de requisitos;

- os impactos de uma alta taxa de *turnover* ou da saída de um membro da equipe durante a especificação;
- os impactos de cometer erros ao estimar a quantidade de requisitos da liberação;
- os impactos do aumento da probabilidade com que os erros são cometidos;
- os efeitos da alocação de uma determinada porcentagem dos recursos para as atividades de garantia de qualidade;
- os efeitos do planejamento de um determinado prazo para o término da especificação;
- os efeitos da alocação de uma determinada quantidade de pessoas na equipe para iniciar a especificação;
- os efeitos do uso de uma determinada quantidade de esforço extra para suprir a falta de homens-dia percebida e evitar alterações no prazo planejado;
- os efeitos de uma maior ou menor tendência em: alterar o prazo planejado, modificar o tamanho da equipe e utilizar esforço extra da equipe.

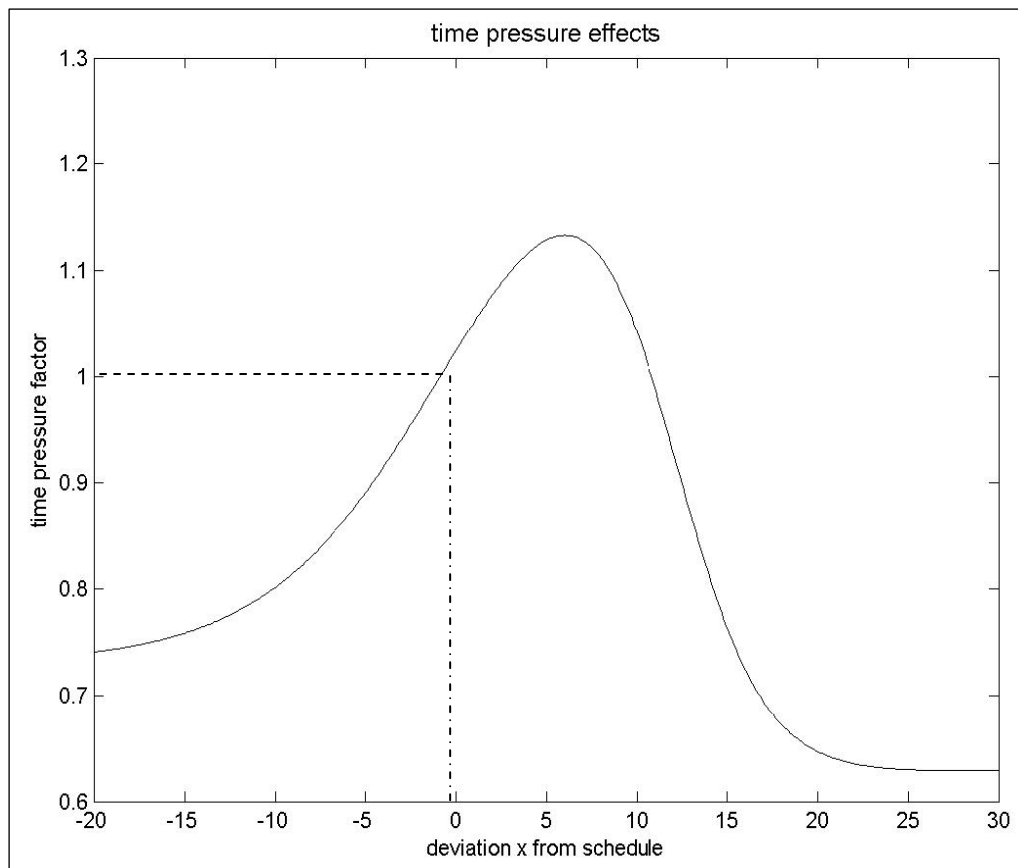
Após simular o modelo, os valores dos estoques *Requisitos com Erros*, *Prazo Planejado para o Término da Especificação* e *Homens-dia Gastos* informam respectivamente a qualidade da especificação produzida, o tempo gasto e o custo do trabalho. Os valores desses estoques podem ser utilizados para avaliar os impactos dos riscos e as conseqüências das decisões gerenciais planejadas.

O modelo também permite avaliar a eficácia de algumas ações planejadas com o objetivo de mitigar ou diminuir os impactos da materialização de riscos. Por exemplo, uma alta taxa de *turnover* ou uma alta taxa de alteração de requisitos podem demandar o uso de esforço extra da equipe, o que pode aumentar o custo. Para evitar ou minimizar o uso do esforço extra, pode-se iniciar a especificação com uma equipe maior e/ou aumentar o prazo planejado. Mas essas alternativas também aumentam o custo. As simulações do modelo permitem comparar as conseqüências das diferentes ações e guiar o gerente na escolha da ação mais eficaz.

Em um cenário em que há risco de atraso no projeto e o adiamento do prazo é inaceitável, pode ser preciso aumentar o tamanho da equipe. No entanto, é necessário um intervalo de tempo para efetuar a alocação de novas pessoas. Há também o tempo de aprendizagem, que é o período em que os membros novatos possuem uma baixa

produtividade. Para evitar o atraso do projeto, pode-se utilizar, durante algum tempo, esforço extra da equipe.

Entretanto, se a sobrecarga de trabalho for excessiva e/ou prolongada, ocorrerá fadiga e estresse da equipe, o que pode provocar uma queda brusca na produtividade (HANNE; NEU, 2004; COLLOFELLO *et al.*, 1998). O gráfico na Figura 4.3 mostra como isso pode ocorrer. O efeito final pode ser um atraso maior que o temido inicialmente. As simulações do modelo podem ser utilizadas para verificar até que ponto a sobrecarga de trabalho é benéfica para o projeto.



Fonte: (HANNE; NEU, 2004)

Figura 4.3. Aumento na produtividade devido ao esforço extra e queda na produtividade devido à fadiga e estresse da equipe

No gráfico da Figura 4.3, o eixo horizontal representa o desvio do projeto em relação ao prazo planejado, que é percebido pela equipe. O valor do desvio é determinado pela diferença entre o tempo utilizado (t) e o tempo planejado (t_p) para alcançar um determinado marco do projeto, ou seja, $t - t_p$. O eixo vertical representa o efeito das pressões de prazo sobre a produtividade. Se o valor nesse eixo for maior que 1, há um aumento na produtividade; se for menor que 1, há uma redução na produtividade; e se for igual a 1, a produtividade não é afetada.

O gráfico mostra que quando o desvio ($t - t_p$) é igual a 0, o valor do efeito das pressões de prazo sobre a produtividade é igual a 1, ou seja, a produtividade não é afetada. O aumento nas pressões de prazo, que ocorre quando o desvio é positivo, inicialmente provoca um aumento no valor do efeito das pressões de prazo, aumentando a produtividade. No entanto, a partir de um determinado momento, o aumento das pressões de prazo provoca uma diminuição no valor do efeito, causando uma queda na produtividade.

5 SIMULAÇÕES DO MODELO

Nesse capítulo são apresentados os resultados de simulações realizadas com o objetivo de mostrar a viabilidade e a aplicabilidade do modelo em ser utilizado como uma ferramenta de apoio à tomada de decisão durante a fase de requisitos. As simulações apresentadas mostram como o modelo pode ser utilizado para realizar análises gerenciais.

A seção 5.1 apresenta os resultados de simulações realizadas em que os cenários foram definidos a partir de dados disponibilizados na literatura, relatados por outros grupos de pesquisa. Enquanto que a seção 5.2 apresenta os resultados de simulações em que os cenários foram definidos utilizando-se dados coletados em uma empresa de desenvolvimento de software que desenvolve aplicações para uso no domínio científico e tecnológico.

5.1 Ajustando o modelo com dados disponibilizados na literatura

5.1.1 Definição e simulação de um cenário base

Inicialmente foi definido e simulado um cenário base em que não é considerada a ocorrência de nenhum risco. Em seguida, outros cenários foram estabelecidos e simulados, e os resultados das simulações desses cenários foram comparados com os obtidos com a simulação do cenário base. A Tabela 5.1 apresenta as suposições e os valores atribuídos a algumas variáveis do painel de controle para simular o cenário base. Com exceção das variáveis 1, 2 e 3, os valores definidos na Tabela 5.1 são baseados em dados disponibilizados na literatura, relatados por outros grupos de pesquisa.

O valor utilizado para configurar a variável 1 foi definido a partir de dados fornecidos por uma empresa de desenvolvimento de software. A coleta desses dados será apresentada detalhadamente na seção 5.2. Os valores utilizados para configurar as variáveis 2 e 3 são baseados na experiência dos autores e em trocas de experiências não documentadas com profissionais de empresas de desenvolvimento de software. Em uma situação hipotética em que a equipe não disponibiliza esforço

extra e nenhum esforço é utilizado para a correção de erros, alterações de requisitos e atividades de garantia de qualidade, os valores atribuídos às variáveis 1, 2 e 3 indicam que são necessários 30 dias para terminar a especificação dos requisitos. Ou seja, uma equipe com 2 pessoas, onde cada uma especifica 2 requisitos por dia, precisará de 30 dias para especificar 120 requisitos.

Tabela 5.1. Valores utilizados para configurar o cenário base

1 - <i>Quantidade inicial de pessoas na equipe</i> : 2
2 - <i>Requisitos da Liberação</i> : 120 requisitos estimados em 120 pontos de função
3 - <i>Quantidade de requisitos especificados por homem-dia</i> : 2
4 - <i>Probabilidade de cometer erros</i> : 12%
5 - <i>Probabilidade de alteração de requisitos</i> : 3%
6 - <i>Probabilidade de turnover</i> : 0% (sem ocorrência de turnover)
7 - <i>Porcentagem de requisitos não previstos</i> : 0% (todos os requisitos são previstos)
8 - <i>Porcentagem desejada de homens-dia alocados para garantia de qualidade</i> : 20%
9 - <i>Máximo homens-dia extras disponibilizados por um membro experiente</i> : 0,5 homens-dia

Observação: Ao definir que a quantidade de requisitos especificados por homem-dia é 2, supõe-se que a produtividade da equipe é 2 pontos de função / homem-dia.

Fonte: Elaborado pelo Autor

Os valores utilizados para configurar as demais variáveis (4 a 9) da Tabela 5.1 foram definidos a partir dos intervalos de valores delimitados na Tabela 4.1, apresentada na seção 4.3. Por exemplo, as variáveis *probabilidade de cometer erros* e *probabilidade de alteração de requisitos* recebem respectivamente os valores 12% e 3%, que são valores médios de acordo com a Tabela 4.1. Já a variável *máximo homens-dia extras disponibilizados por um membro experiente* recebe o valor 0,5, que é um valor alto de acordo com a Tabela 4.1. Para configurar os demais cenários apresentados nas seções seguintes (5.1.2 a 5.1.9) também foram utilizados os intervalos de valores definidos na Tabela 4.1. Como foi informado na seção 4.3, os intervalos de valores apresentados na Tabela 4.1 foram delimitados a partir de dados disponibilizados na literatura.

Os valores das demais variáveis do painel de controle, que não estão relacionadas na Tabela 5.1, foram definidos a partir da experiência dos autores e de trocas de experiências não documentadas com profissionais de empresas de desenvolvimento de software. Isso não prejudica a realização de comparações entre

os cenários apresentados, pois os valores atribuídos a essas variáveis são os mesmos em todos os cenários simulados. As Figuras 5.1 e 5.2 mostram o painel de controle do modelo. Na Figura 5.1 é possível verificar como as variáveis do modelo foram ajustadas para simular o cenário base. E na Figura 5.2 são apresentados os valores obtidos para as variáveis utilizadas nas análises realizadas após a simulação.

Configuração			
máximo homens-dia extras disponibilizados sem custo adicional	nível máximo de exaustão suportada	tamanho médio de cada requisito em pontos de função	quantidade de requisitos especificados por homem-dia
máximo homens-dia não disponibilizados por um membro experiente	diminuição da produtividade após a exaustão	Requisitos Entregues Especificados	quantidade de alterações realizadas por homem-dia
porcentagem do excesso de prazo absorvido	multiplicador para a exaustão causada pelo aumento da jornada de trabalho	Requisitos Entregues com Erros	quantidade de erros detectados por homem-dia
relação: produtividade de um novato / produtividade de um experiente	Requisitos da Liberação	quantidade de erros descobertos por dia	quantidade de falhas corrigidas por homem-dia

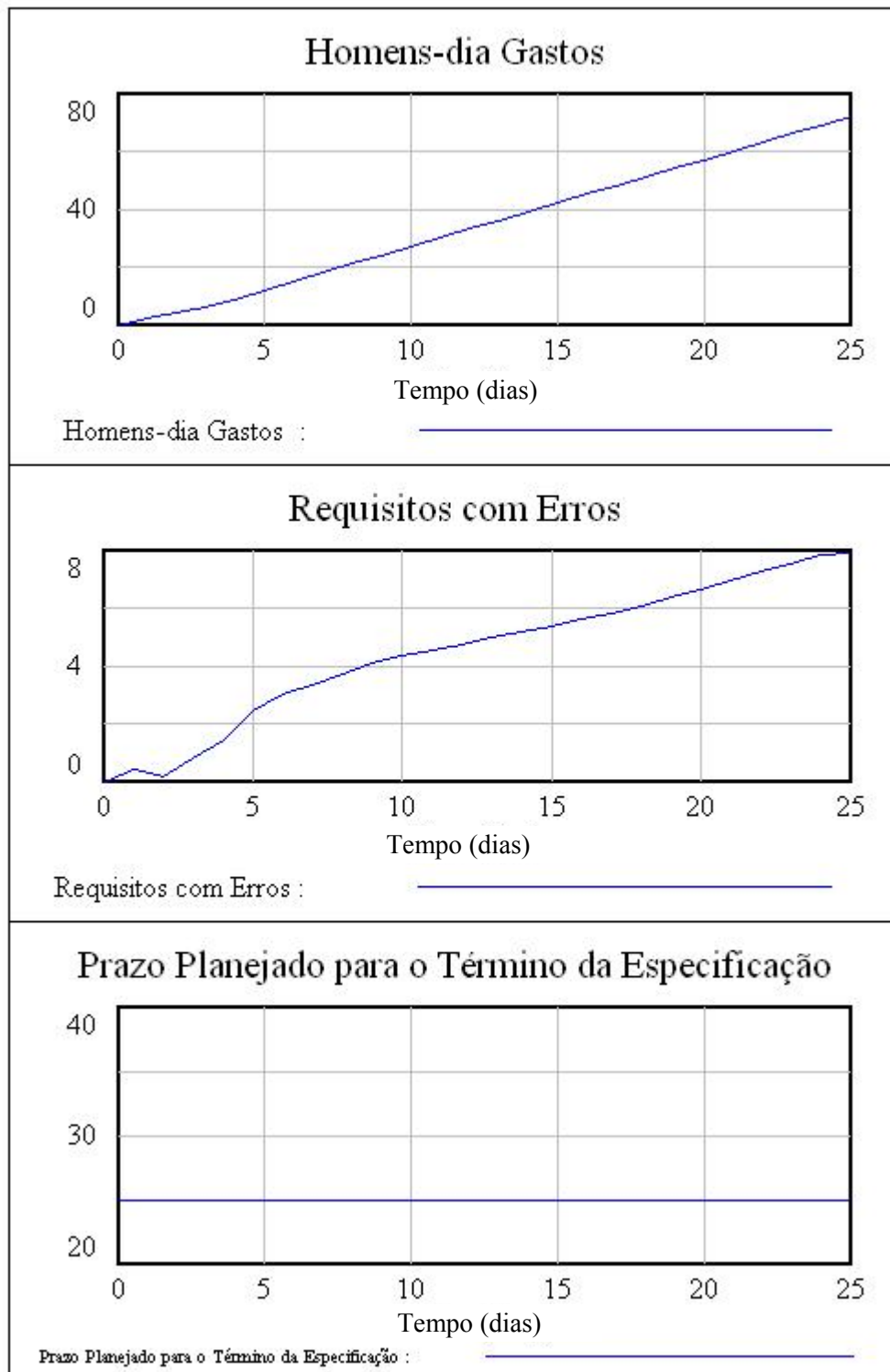
Análise - simular riscos			
porcentagem de requisitos não previstos	probabilidade de alteração de requisitos	probabilidade de alteração de requisitos entregues	probabilidade de cometer erros
atenuação para erros durante a correção e alteração	quantidade de profissionais que abandonam a equipe	tempo em que ocorre turnover	

Análise - determinar decisões e políticas gerenciais			
quantidade inicial de pessoas na equipe	porcentagem desejada de homens-dia alocados para garantia de qualidade	tempo desejado para realizar as alterações	tempo para aumento do prazo
quantidade mínima de pessoas na equipe	porcentagem da falta de homens-dia tratada com esforço extra	tempo desejado para realizar as correções	tempo para redução do prazo
força de trabalho reserva	máximo homens-dia extras disponibilizados por um membro experiente	tempo para aumento do uso de esforço extra	tempo para aumento do tamanho da equipe
Prazo (inicial) Planejado para o Término da Especificação	quantidade máxima de novatos que um membro experiente pode treinar	tempo para redução do uso de esforço extra	tempo para redução do tamanho da equipe

Fonte: Elaborado pelo Autor

Figura 5.1. Painel de controle – entrada de valores

Análise - verificação após a simulação



Observação: Os valores apresentados nos gráficos são os valores acumulados nos respectivos estoques em um determinado tempo, medido em dias, após o início da especificação.

Fonte: Elaborado pelo Autor

Figura 5.2. Painel de controle – saída de valores

Os valores atribuídos às variáveis *tempo para aumento do uso de esforço extra*, *tempo para aumento do prazo* e *tempo para aumento do tamanho da equipe* indicam

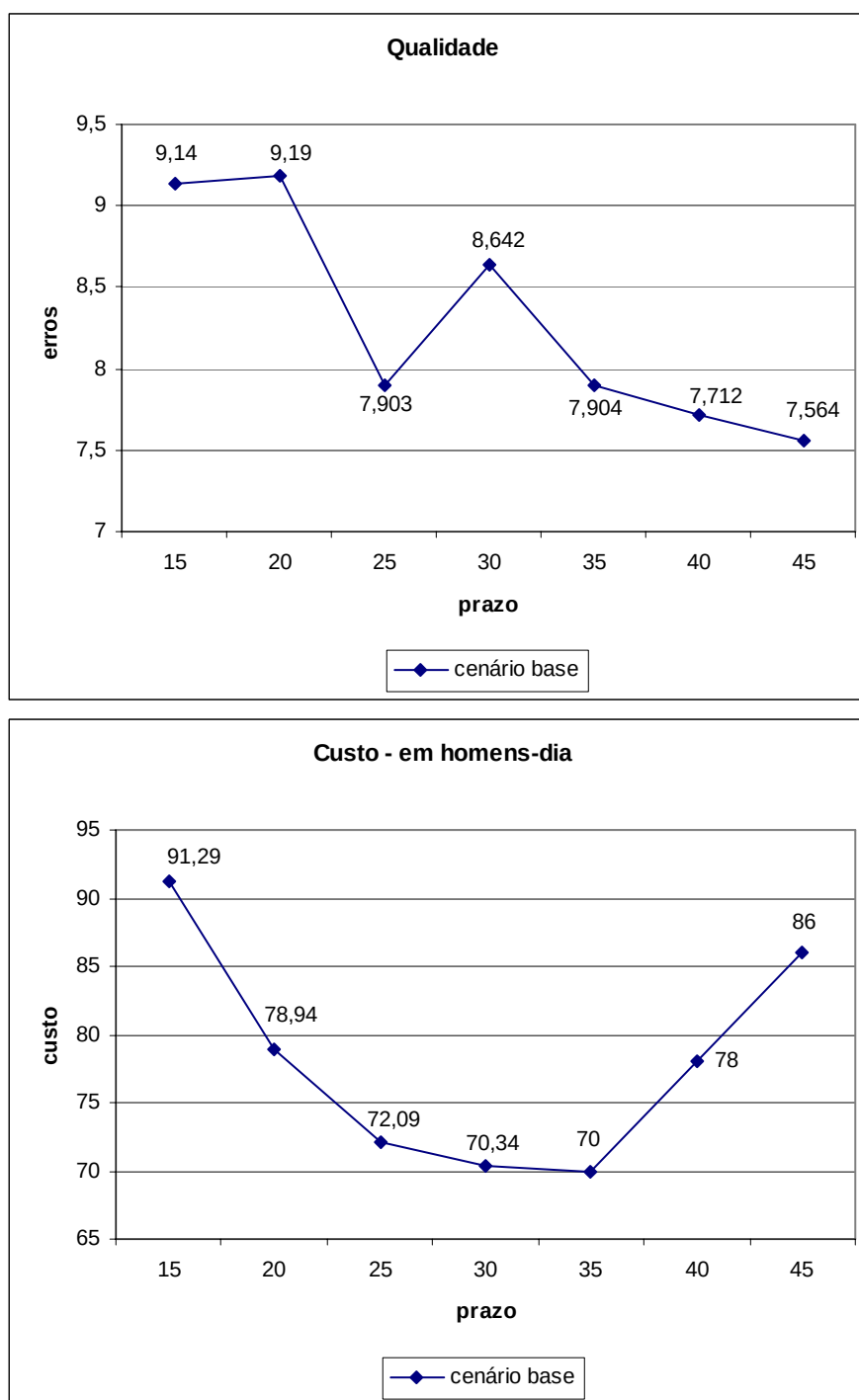
que a simulação irá considerar um gerenciamento que, diante de riscos de atraso no projeto, tende a aumentar o uso do esforço extra e evitar modificações no tamanho da equipe e no prazo. Se o uso de esforço extra não for suficiente para evitar o atraso, os valores atribuídos às variáveis indicam que a segunda opção da gerência é aumentar o tamanho da equipe. Modificações no prazo só serão feitas se forem estritamente necessárias.

Conforme foi informado na seção 3.5, a qualidade da especificação de requisitos pode ser medida pelo valor final do estoque *Requisitos com Erros*, que representa a quantidade de requisitos especificados incorretamente que serão entregues para as fases seguintes do processo de desenvolvimento. O custo para a realização da especificação corresponde ao valor final do estoque *Homens-dia Gastos*, que representa a quantidade de esforço medido em homens-dia utilizado.

Os gráficos apresentados na Figura 5.3 mostram a qualidade (valor do estoque *Requisitos com Erros*) e o custo (valor do estoque *Homens-dia Gastos*) da especificação de requisitos em função do prazo planejado. Os valores nos gráficos foram obtidos ao simular o modelo considerando o cenário base descrito.

O gráfico que compara a qualidade mostra que a redução no prazo planejado tende a aumentar a quantidade de requisitos especificados incorretamente. Segundo (ABDEL-HAMID, 1996) e (LIN; ABDEL-HAMID; SHERIF, 1997), para terminar a especificação em um tempo menor, pode ser necessário aumentar o tamanho da equipe e utilizar esforço extra dos profissionais, o que contribui para aumentar a quantidade de erros cometidos (COLLOFELLO *et al.*, 1998; LIN; ABDEL-HAMID; SHERIF, 1997).

Quando o prazo planejado é 30 dias, a quantidade de erros de especificação é maior se comparada à quantidade de erros quando o prazo é 25 ou 35 dias, o que provoca uma oscilação na curva do gráfico de qualidade. Os valores obtidos ao simular o modelo, variando linearmente o valor de uma variável (como foi feito com a variável que representa o prazo), podem oscilar porque as variáveis que formam a estrutura dos modelos de dinâmica de sistemas se relacionam dinamicamente de forma não linear com vários *loops* de realimentação (*feedback*). Isso explica o comportamento observado e a oscilação na curva do gráfico. É comum ocorrerem oscilações nas curvas dos gráficos ao utilizá-los para organizar e visualizar os resultados das simulações do modelo. Em alguns gráficos apresentados nas seções seguintes também ocorrem oscilações em suas curvas.



Fonte: Elaborado pelo Autor

Figura 5.3. Qualidade e custo obtidos com o cenário base

Ao simular o modelo considerando um prazo de 25 dias, o tamanho da equipe, que inicialmente é 2, aumenta para 3 no terceiro dia e permanece assim até o fim da especificação. Nesse caso, como ocorre aumento no tamanho da equipe, a quantidade de esforço extra utilizado é pequena. Ao planejar um prazo de 30 dias, a equipe permanece com 2 pessoas até o dia 27 (assumindo que a especificação foi iniciada no

dia 1). Para que a especificação termine no prazo planejado e seja evitado um aumento do tamanho da equipe, é necessário utilizar uma quantidade maior de esforço extra, que nesse caso provoca exaustão da equipe. O aumento no uso do esforço extra contribui para aumentar a quantidade de erros cometidos (COLLOFELLO *et al.*, 1998; LIN; ABDEL-HAMID; SHERIF, 1997), o que explica o aumento na quantidade de requisitos com erros quando o prazo planejado é 30 dias. Por fim, quando o prazo aumenta para 35 dias, a equipe permanece com 2 pessoas até o término da especificação e a quantidade de esforço extra utilizado é pequena, contribuindo para que poucos erros sejam cometidos.

O gráfico de custo mostra que um prazo de 35 dias estabelece o menor custo. Ao planejar e incentivar o término da especificação em um prazo menor, pode ser necessário aumentar o tamanho da equipe e/ou utilizar uma quantidade excessiva de esforço extra dos profissionais (ABDEL-HAMID, 1996; LIN; ABDEL-HAMID; SHERIF, 1997), o que implica no aumento do custo. Por outro lado, quando o prazo para o término da especificação é superestimado, a equipe percebe que o projeto está adiantado em relação ao planejamento. Ao perceber a folga no prazo, a equipe irá trabalhar com menor produtividade, absorvendo uma parte do excesso de tempo disponível (ABDEL-HAMID; MADNICK, 1991). Isso explica o aumento que ocorre no custo quando o prazo planejado é maior que 35 dias.

O custo mínimo e a inversão na tendência da curva no gráfico de custo ocorrem próximo ao prazo de 30 dias devido aos valores atribuídos às variáveis: *quantidade inicial de pessoas na equipe, requisitos da liberação e quantidade de requisitos especificados por homem-dia*. Como foi informado na Tabela 5.1, essas variáveis recebem, respectivamente, os valores: 2, 120 e 2. Considerando que a equipe não disponibiliza esforço extra e nenhum esforço é utilizado para a correção de erros, alterações de requisitos e atividades de garantia de qualidade, uma equipe com 2 pessoas, onde cada uma especifica 2 requisitos por dia, precisará de 30 dias para especificar 120 requisitos.

Em uma situação real, é necessário disponibilizar esforço para correção de erros, alterações de requisitos e atividades de garantia de qualidade, o que aumenta o custo. No entanto, o esforço extra disponibilizado pela equipe, por meio de um maior empenho durante o horário normal de trabalho, pode amenizar os impactos sobre o custo. Isso contribui para que o prazo ideal continue próximo a 30 dias. As

simulações mostraram que, para o cenário base, o prazo ideal é em torno de 30 a 35 dias.

Nas seções seguintes (5.1.2 a 5.1.9) são apresentados outros cenários, que consideram a ocorrência de riscos e permitem comparar os efeitos da implementação de diferentes políticas gerenciais. Para os cenários simulados nas seções seguintes, as variáveis que não possuem seus valores explicitamente informados recebem os mesmos valores definidos para a simulação do cenário base. Esses valores podem ser visualizados na Figura 5.1, que mostra o painel de controle configurado para simular o cenário base.

5.1.2 Aumento na quantidade de erros cometidos

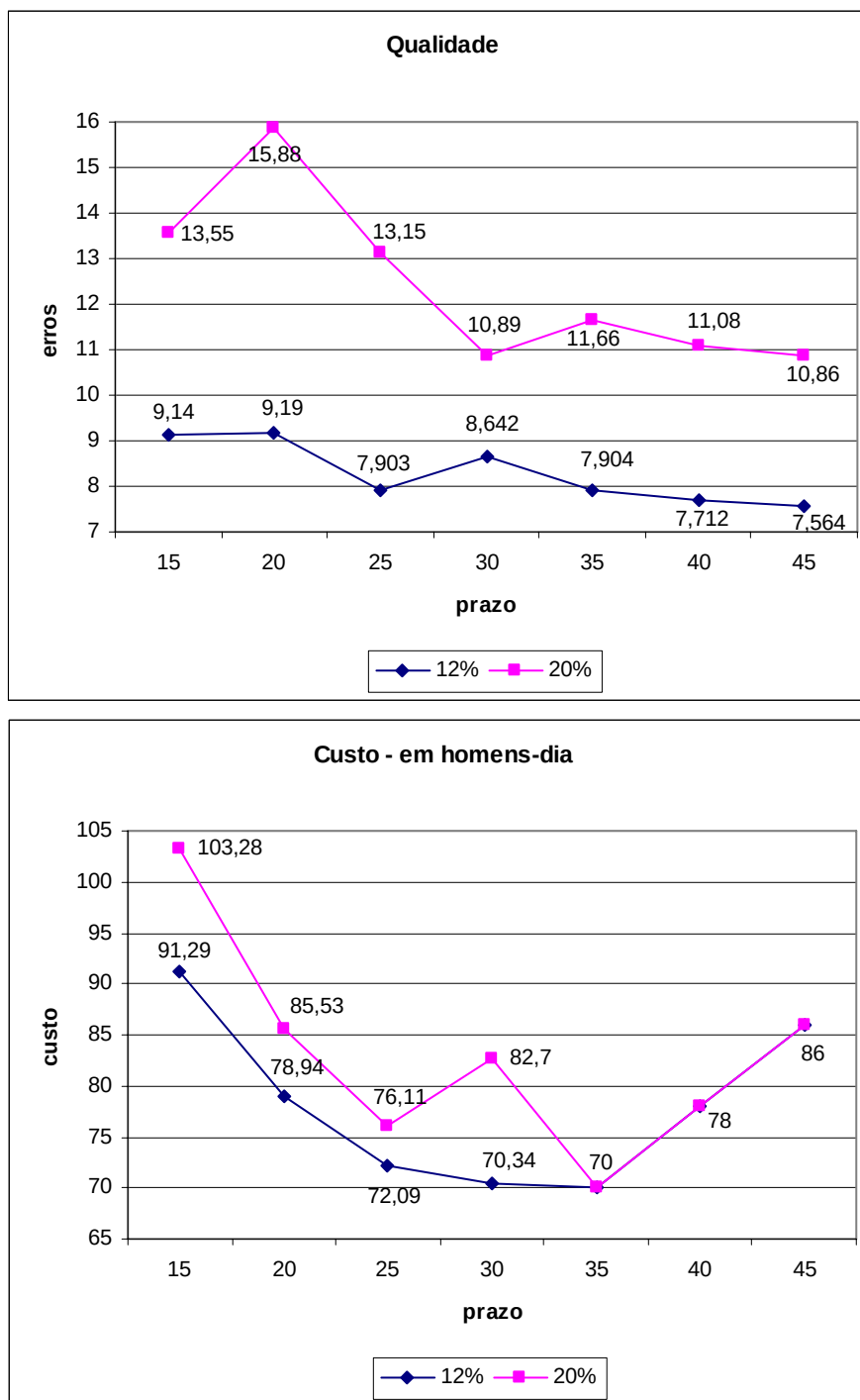
Os gráficos apresentados na Figura 5.4 mostram a qualidade e o custo da especificação de requisitos em função do prazo planejado. Os valores permitem comparar dois cenários: o cenário base em que a probabilidade de cometer erros é 12%, e outro cenário em que a probabilidade de cometer erros é 20%, que é um valor alto de acordo com os intervalos definidos na Tabela 4.1.

Em um projeto real, o aumento na quantidade de erros pode ser provocado por diversos fatores, tais como: dificuldade e complexidade de um determinado projeto, inexperiência e menor capacidade dos profissionais alocados na equipe, baixa confiabilidade das ferramentas utilizadas, etc.

Os formatos das curvas nos gráficos são parecidos, o que é esperado, pois as variáveis que determinam as decisões gerenciais recebem os mesmos valores nos dois cenários simulados. O gráfico que compara a qualidade mostra que o aumento na quantidade de erros cometidos contribui para o aumento da quantidade de requisitos especificados incorretamente. À medida que os erros são detectados pelas atividades de garantia de qualidade, deve ser alocado esforço adicional para realizar as correções. Como a quantidade de esforço necessário é maior, para evitar atrasos no projeto pode ser preciso utilizar esforço extra da equipe, o que aumenta ainda mais a quantidade de erros cometidos (COLLOFELLO *et al.*, 1998; LIN; ABDEL-HAMID; SHERIF, 1997).

A experiência da equipe também deve ser considerada. Se o uso de esforço extra não for suficiente para realizar as correções, pode ser necessário alocar novos profissionais na equipe. Inicialmente, esses profissionais são inexperientes e cometem mais erros que os demais (ABDEL-HAMID; MADNICK, 1991; LIN;

ABDEL-HAMID; SHERIF, 1997). Além disso, as pressões de prazo e o risco de atraso tendem a diminuir a quantidade de esforço alocado para as atividades de garantia de qualidade (ABDEL-HAMID; MADNICK, 1991; ABDEL-HAMID; SENGUPTA; SWETT, 1999), o que contribui para que menos erros sejam detectados.



Fonte: Elaborado pelo Autor

Figura 5.4. Aumento na quantidade de erros cometidos

O gráfico de custo mostra que o aumento na quantidade de erros cometidos provoca o aumento do custo. Isso ocorre porque é necessário alocar esforço adicional para realizar as correções, o que aumenta a quantidade de retrabalho no projeto. Quando o prazo para o término da especificação é superestimado, tornando-se maior que o necessário, o gráfico de custo mostra que a diferença entre os valores nos dois cenários diminui. Nesse caso, as curvas tendem a se aproximar (ou até mesmo sobrepõem-se) porque quando o prazo é maior que o necessário no cenário em que a probabilidade de cometer erros é 20%, grande parte do esforço adicional utilizado para realizar as correções não é obtida por meio do uso de horas-extras dos profissionais da equipe. A maior parte desse esforço é obtida a partir do esforço que deixaria de ser disponibilizado pela equipe, para absorver uma parte do excesso de tempo disponível.

No cenário em que a probabilidade de cometer erros é 20%, o custo obtido quando o prazo é 30 dias é maior que o custo obtido quando o prazo planejado é 25 ou 35 dias, provocando uma oscilação que interfere na tendência da curva no gráfico de custo.

Para os prazos de 25 e 30 dias, o tamanho da equipe, que inicialmente é 2, aumenta para 3 no quarto e no quinto dia respectivamente, e assim permanece até o dia 22 (assumindo que a especificação foi iniciada no dia 1). Durante esse período (22 dias de trabalho), a equipe especifica 86% dos requisitos quando o prazo é 25 dias e 78% dos requisitos quando o prazo é 30 dias. A maior produtividade, obtida quando o prazo é 25 dias, é alcançada com o uso de uma maior quantidade de esforço extra da equipe. No entanto, grande parte desse esforço extra é obtido durante o horário normal de trabalho, sem exigir o uso de horas-extras. Com isso, ao comparar os custos do trabalho realizado nesses 22 dias, verifica-se que o custo quando o prazo é 25 dias é apenas um pouco maior que o custo obtido quando o prazo é 30 dias. No 23º dia, o tamanho da equipe aumenta para 4 quando o prazo é 25 dias e com mais dois dias de trabalho, a especificação termina. Quando o prazo é 30 dias, não ocorre aumento no tamanho da equipe, mas são necessários mais sete dias de trabalho, o que é suficiente para que o custo obtido quando o prazo é 30 dias seja maior que o obtido quando o prazo é 25 dias.

Quando o prazo é 35 dias, a equipe permanece com 2 pessoas até o final da especificação e todo o esforço extra utilizado é obtido durante o horário normal de trabalho dos profissionais que compõem a equipe. Quando o prazo é 30 dias, a

equipe permanece com 3 pessoas durante 22 dias, o que explica porque o custo é maior para esse prazo.

5.1.3 Aumento na volatilidade dos requisitos

Os gráficos apresentados na Figura 5.5 mostram a qualidade e o custo da especificação de requisitos em função do prazo planejado. Os valores permitem comparar dois cenários: o cenário base em que a probabilidade de alteração de requisitos é 3%, e outro cenário em que a probabilidade de alteração de requisitos é 9%, que é um valor alto de acordo com os intervalos definidos na Tabela 4.1.

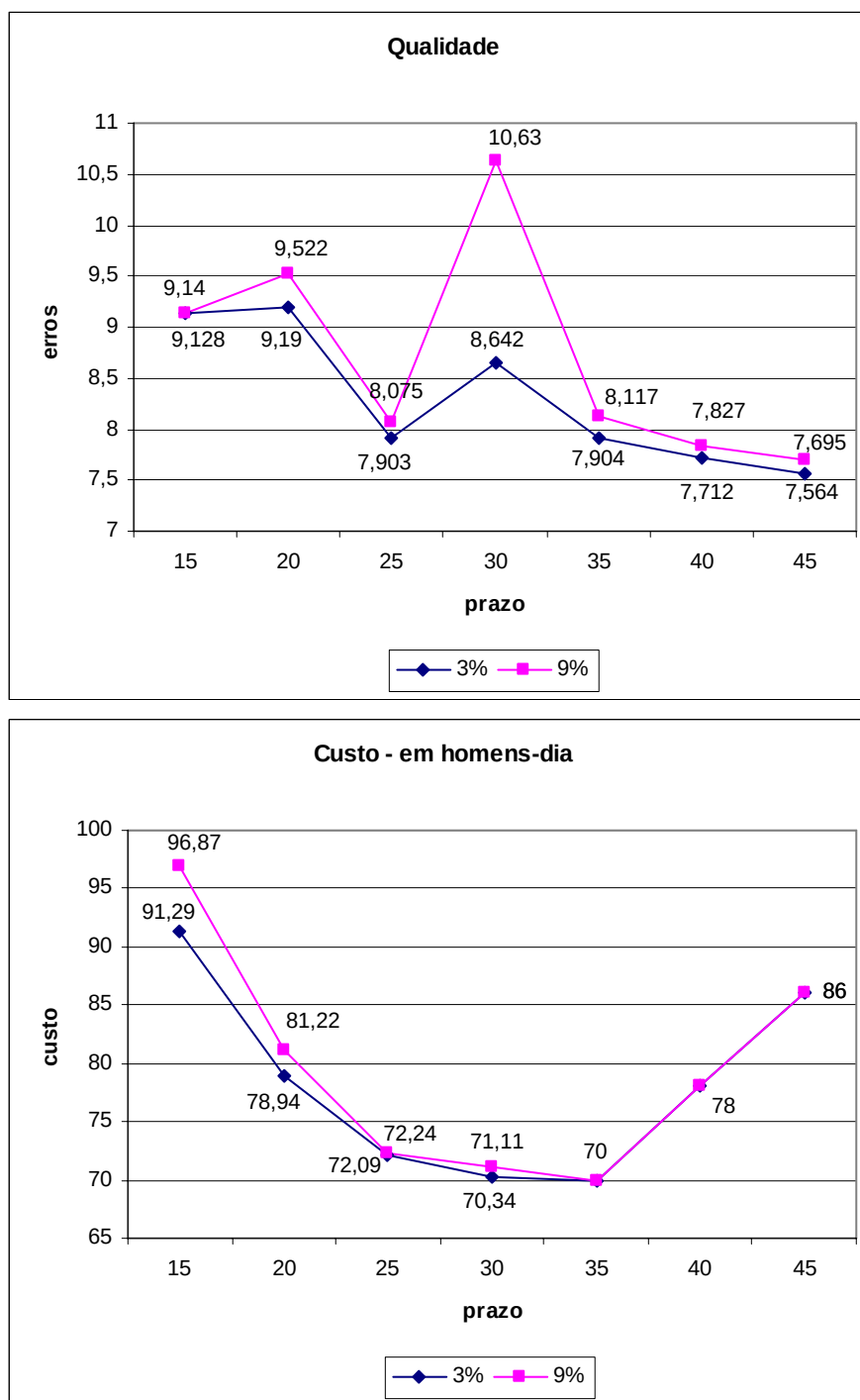
De modo semelhante ao ocorrido na seção 5.1.2, os formatos das curvas nos gráficos são parecidos, exceto pela oscilação que ocorre no gráfico de qualidade quando o prazo planejado é 30 dias. A semelhança entre os formatos das curvas é esperada porque as variáveis que determinam as decisões gerenciais recebem os mesmos valores nos dois cenários simulados.

O gráfico que compara a qualidade mostra que o aumento na volatilidade dos requisitos provoca o aumento da quantidade de requisitos especificados incorretamente. Para realizar uma quantidade maior de alterações, deve ser utilizada uma quantidade maior de esforço. Como a quantidade de esforço necessário é maior, para evitar atrasos no projeto pode ser preciso utilizar esforço extra da equipe, o que aumenta a quantidade de erros cometidos (COLLOFELLO *et al.*, 1998; LIN; ABDEL-HAMID; SHERIF, 1997).

A experiência da equipe também deve ser considerada. Se o uso de esforço extra não for suficiente para realizar as alterações, pode ser necessário alocar novos profissionais na equipe. Inicialmente, esses profissionais são inexperientes e cometem mais erros que os demais (ABDEL-HAMID; MADNICK, 1991; LIN; ABDEL-HAMID; SHERIF, 1997). Além disso, as pressões de prazo e o risco de atraso tendem a diminuir a quantidade de esforço alocado para as atividades de garantia de qualidade (ABDEL-HAMID; MADNICK, 1991; ABDEL-HAMID; SENGUPTA; SWETT, 1999), o que contribui para que menos erros sejam detectados.

De modo semelhante ao ocorrido no cenário base descrito na seção 5.1.1, no cenário em que a probabilidade de alteração de requisitos é 9%, quando o prazo planejado é 30 dias, a quantidade de erros de especificação é maior se comparada à

quantidade de erros quando o prazo é 25 ou 35 dias. Esse comportamento do modelo é responsável pela oscilação na curva do gráfico de qualidade.



Fonte: Elaborado pelo Autor

Figura 5.5. Aumento na volatilidade dos requisitos

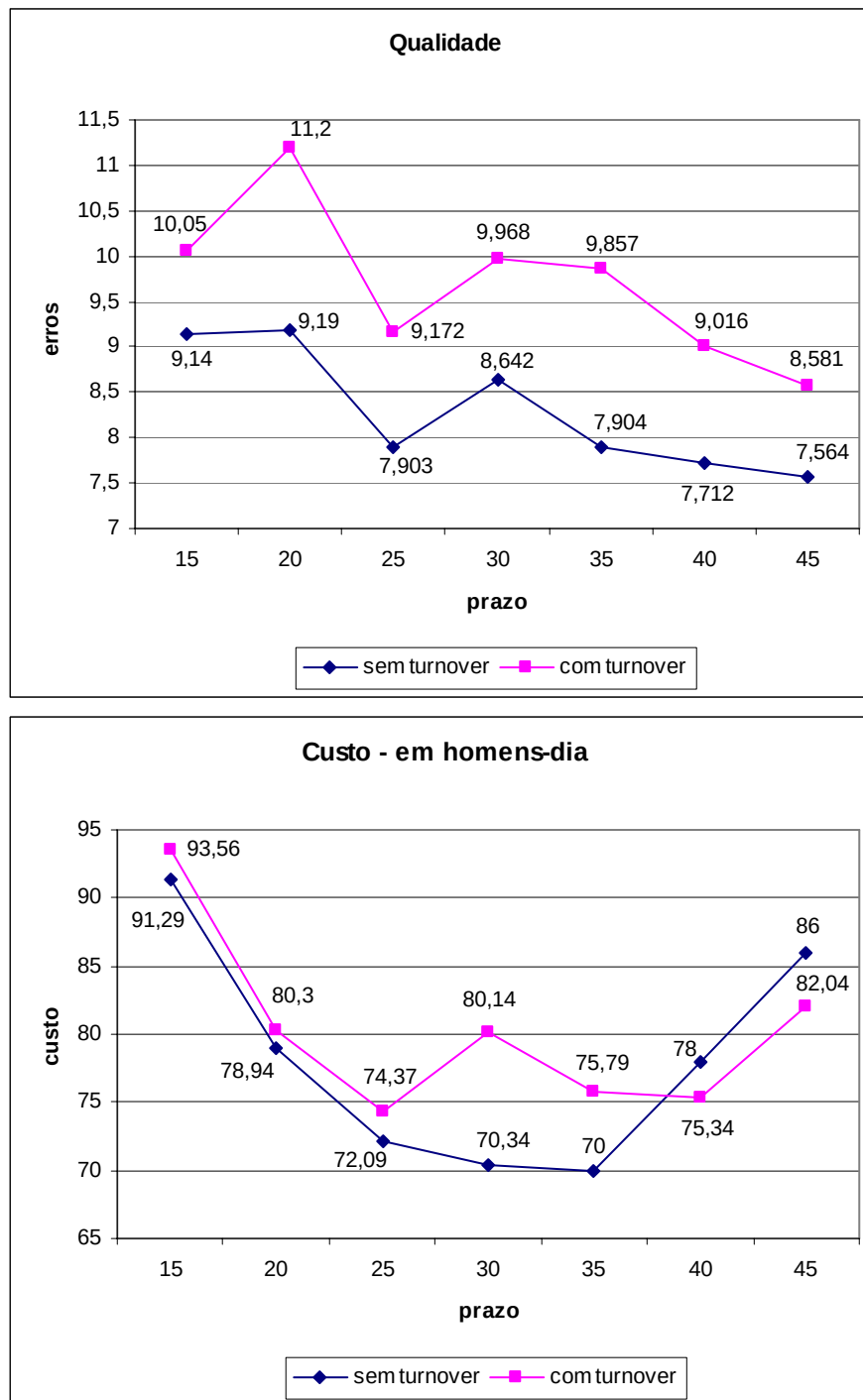
No cenário em que a probabilidade de alteração de requisitos é 9%, ao simular o modelo considerando um prazo de 25 dias, o tamanho da equipe, que inicialmente é 2, aumenta para 3 no terceiro dia e permanece assim até o término da especificação.

Nesse caso, como ocorre aumento no tamanho da equipe, a quantidade de esforço extra utilizado é pequena. Ao planejar um prazo de 30 dias, a equipe permanece com 2 pessoas até o dia 18 (assumindo que a especificação foi iniciada no dia 1). Para que a especificação termine no prazo e seja evitado um aumento do tamanho da equipe, é necessário utilizar uma quantidade maior de esforço extra, que nesse caso provoca exaustão da equipe. Quando a exaustão ocorre no dia 19, é alocado um novo membro na equipe. O aumento no uso do esforço extra e a alocação de membros novatos na equipe contribuem para aumentar a quantidade de erros cometidos, o que explica o aumento na quantidade de requisitos com erros quando o prazo planejado é 30 dias. Por fim, quando o prazo é 35 dias, a equipe permanece com 2 pessoas até o término da especificação e a quantidade de esforço extra utilizado é pequena, contribuindo para que poucos erros sejam cometidos.

O gráfico de custo mostra que o aumento na volatilidade dos requisitos provoca o aumento do custo. Isso ocorre porque é necessário alocar esforço adicional para realizar as alterações, o que aumenta a quantidade de retrabalho no projeto. Quando o prazo para o término da especificação é superestimado, tornando-se maior que o necessário, o gráfico de custo mostra que a diferença entre os valores nos dois cenários diminui. Nesse caso, as curvas tendem a se aproximar (ou até mesmo sobrepõem-se) porque quando o prazo aumenta no cenário em que a probabilidade de alteração de requisitos é 9%, grande parte do esforço adicional utilizado para realizar as alterações não é obtida por meio do uso de horas-extras dos profissionais da equipe. A maior parte desse esforço é obtida a partir do esforço que deixaria de ser disponibilizado pela equipe, para absorver uma parte do excesso de tempo disponível.

5.1.4 Turnover de pessoal

Os valores nos gráficos apresentados na Figura 5.6 permitem comparar dois cenários: o cenário base em que não ocorre *turnover*, e outro cenário em que ocorre *turnover*. Nesse caso, o *turnover* corresponde a uma pessoa abandonar a equipe quando o tempo decorrido do início da especificação for igual à metade do prazo planejado.



Fonte: Elaborado pelo Autor

Figura 5.6. Impactos da ocorrência de *turnover*

O gráfico que compara a qualidade mostra que quando ocorre *turnover*, há um aumento na quantidade de requisitos especificados incorretamente. Para compensar a saída dos membros experientes da equipe, membros novatos são alocados ao projeto. Os membros novatos, devido à falta de experiência, cometem mais erros. Além disso, o aumento na porcentagem de membros novatos da equipe pode exigir o uso

de esforço extra dos demais para compensar a menor produtividade dos novatos e não atrasar o término da especificação. A sobrecarga de trabalho acarreta estresse e exaustão da equipe, contribuindo para um aumento na quantidade de erros (COLLOFELLO *et al.*, 1998; LIN; ABDEL-HAMID; SHERIF, 1997).

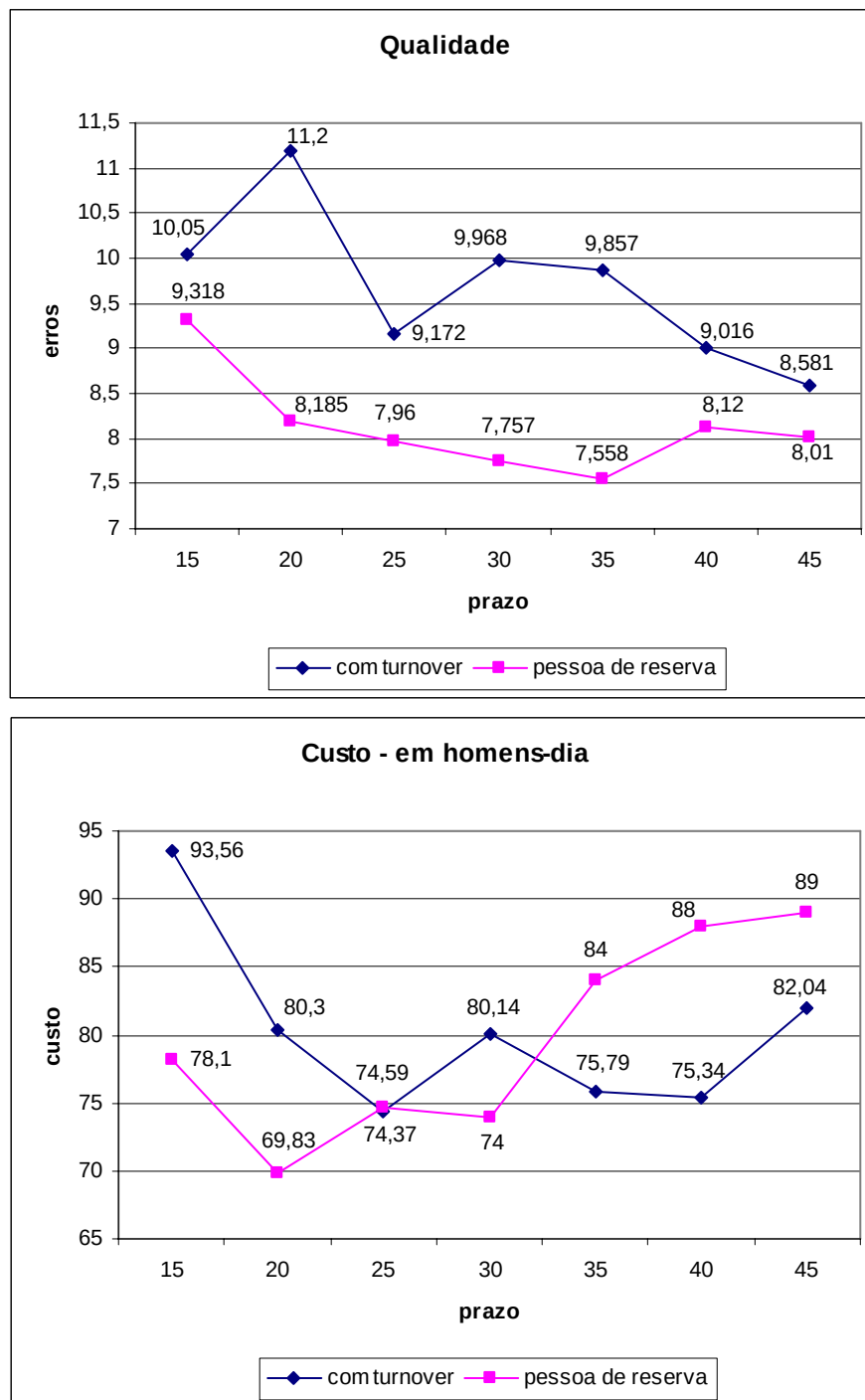
O gráfico de custo mostra que a ocorrência de *turnover* aumenta o custo quando o prazo planejado é menor ou igual a 35 dias. No entanto, quando o prazo aumenta, as duas curvas se cruzam em um determinado ponto no gráfico e então o *turnover* passa a contribuir para uma redução no custo. Em um primeiro momento isso pode parecer um erro de modelagem, contudo esse comportamento do modelo pode ser explicado. A ocorrência de *turnover* pode demandar a alocação de membros novatos, que em geral possuem menor produtividade e cometem mais erros, contribuindo para o aumento do custo. Entretanto, quando o prazo planejado aumenta para 40 dias ou mais, tornando-se maior que o necessário, e um integrante da equipe abandona o projeto, pode não ser preciso alocar imediatamente um novo membro para substituí-lo. Nesse caso, o *turnover* pode contribuir para diminuir o custo, como é mostrado no gráfico.

Na Figura 5.7, os gráficos comparam um cenário em que ocorre *turnover* com outro cenário em que a ocorrência de *turnover* é combinada com uma ação utilizada para mitigar o risco de *turnover* (DEMARCO; LISTER, 2003). Essa ação consiste em manter a equipe com um tamanho maior que o necessário. Nesse caso, se algum integrante abandonar a equipe, há outro profissional, que já conhece o domínio do problema, disponível para assumir suas tarefas. Para simular o cenário em que essa ação gerencial é utilizada, as variáveis do modelo foram ajustadas para considerar que a gerência mantém uma pessoa de reserva na equipe, a partir do início da especificação e até a ocorrência do *turnover*.

O gráfico que compara a qualidade mostra que manter uma pessoa de reserva contribui para diminuir a quantidade de requisitos especificados incorretamente. Isso ocorre porque o aumento no tamanho da equipe diminui a necessidade de utilizar esforço extra, o que diminui a taxa com que os erros são cometidos.

O gráfico de custo mostra que, diante da ocorrência de *turnover*, a decisão de manter uma pessoa de reserva na equipe diminui o custo quando o prazo é menor ou igual a 30 dias. Mas, quando o prazo aumenta para 35 dias ou mais, as duas curvas se cruzam no gráfico, e a decisão de manter uma pessoa de reserva na equipe passa a ter efeito contrário, aumentando o custo. Isso ocorre porque com uma equipe maior, se o

prazo aumentar além do necessário, a folga no prazo torna-se mais evidente. A equipe irá então, trabalhar com menor produtividade, absorvendo uma parte do excesso de tempo disponível, o que aumenta o custo.



Fonte: Elaborado pelo Autor

Figura 5.7. Efeitos de manter uma pessoa de reserva na equipe

Nas simulações que comparam cenários que modelam diferentes decisões gerenciais é importante analisar o ponto em que as curvas se cruzam no gráfico.

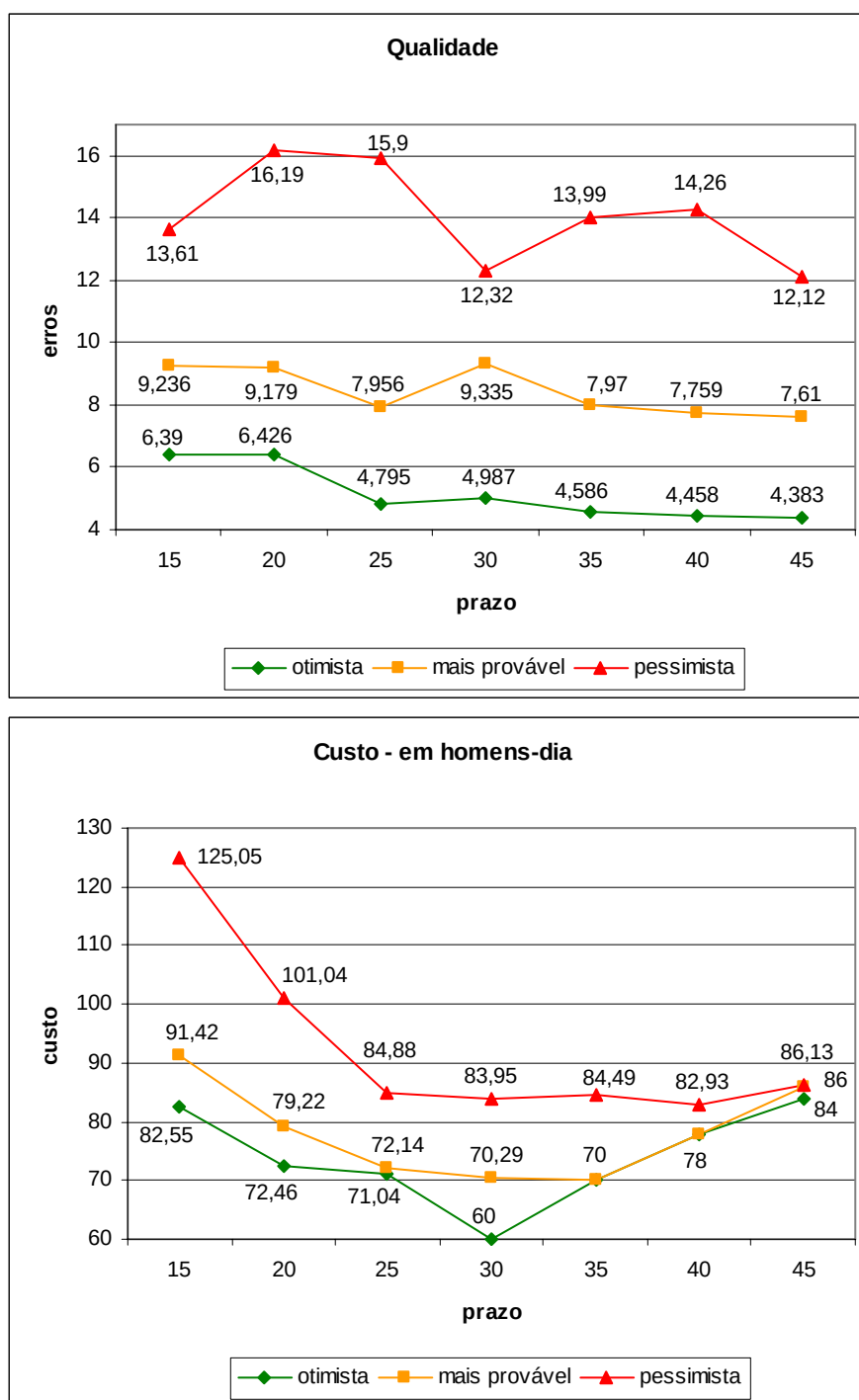
Nesse ponto, as decisões apresentam o mesmo efeito sobre a variável ou fator sendo analisado no gráfico. E a partir dele, os efeitos das decisões podem começar a divergir mais fortemente. Ao caminhar pelo gráfico em um sentido a partir do ponto de cruzamento, as decisões têm efeito contrário ao obtido quando se caminha pelo outro sentido. Tomando como exemplo o ponto de cruzamento entre os prazos 30 e 35 dias no gráfico de custo da Figura 5.7: ao partir desse ponto prosseguindo para a esquerda, a decisão de manter uma pessoa de reserva na equipe contribui para reduzir o custo. Mas, ao prosseguir para a direita, a decisão de manter uma pessoa de reserva na equipe contribui para aumentar o custo.

5.1.5 Comparação entre cenários: otimista, mais provável e pessimista

O cenário otimista corresponde a um cenário ideal em que os fatores são favoráveis ao sucesso do projeto. No cenário mais provável, as variáveis que representam os riscos recebem valores médios e mais prováveis. No cenário pessimista é considerada a materialização de riscos, tais como: aumento na quantidade de erros cometidos, alta volatilidade dos requisitos e ocorrência de *turnover*. As configurações para os três cenários, que são baseadas nos intervalos de valores delimitados na Tabela 4.1, são:

- **otimista:** probabilidade de alteração de requisitos igual a 2%, probabilidade de cometer erros igual a 6% e sem ocorrência de *turnover*;
- **mais provável:** probabilidade de alteração de requisitos igual a 5%, probabilidade de cometer erros igual a 12% e também sem ocorrência de *turover*;
- **pessimista:** probabilidade de alteração de requisitos igual a 10%, probabilidade de cometer erros igual a 20% e ocorrência de *turnover*. Nesse caso, o *turnover* corresponde a uma pessoa abandonar a equipe quando o tempo decorrido do início da especificação for igual à metade do prazo planejado.

Os gráficos apresentados na Figura 5.8 mostram a qualidade e o custo da especificação de requisitos em função do prazo planejado. Mais uma vez, os formatos das curvas nos gráficos são parecidos, o que é esperado, pois as variáveis que determinam as decisões gerenciais recebem os mesmos valores nos três cenários simulados. O gráfico que compara a qualidade mostra que os cenários otimista e pessimista apresentam, respectivamente, a menor e a maior quantidade de requisitos especificados incorretamente.



Fonte: Elaborado pelo Autor

Figura 5.8. Comparação entre cenários

No cenário pessimista, há uma maior quantidade de retrabalho no projeto, o que exige o uso de uma maior quantidade de esforço. Além disso, a saída de um profissional experiente da equipe exige um maior empenho dos demais, até que outro profissional seja alocado na equipe. De modo semelhante às análises feitas anteriormente nas seções 5.1.2, 5.1.3 e 5.1.4, para evitar o atraso do projeto, pode ser

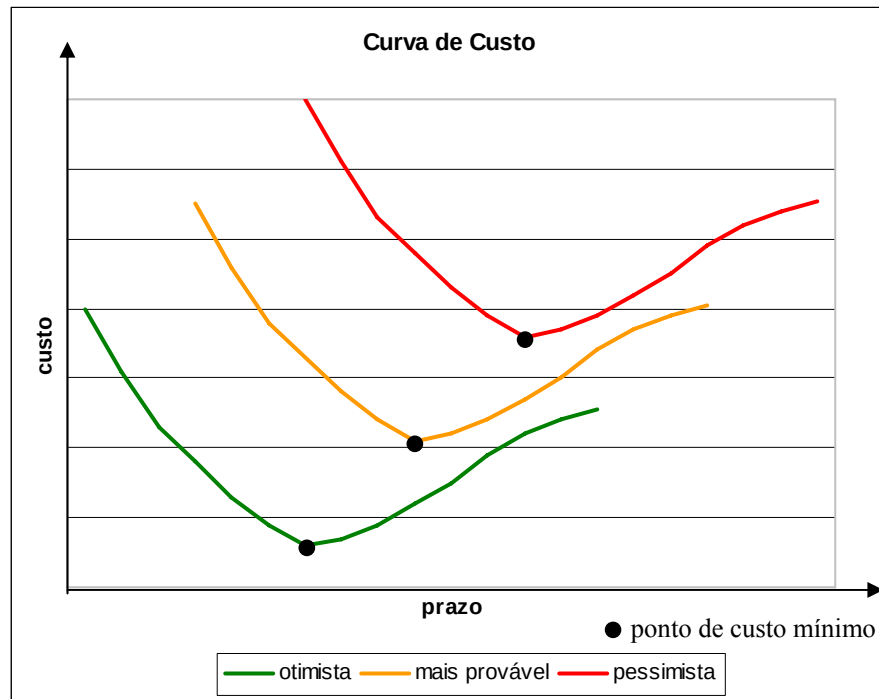
preciso utilizar esforço extra da equipe, o que aumenta a quantidade de erros cometidos (COLLOFELLO *et al.*, 1998; LIN; ABDEL-HAMID; SHERIF, 1997). Além disso, as pressões de prazo e o risco de atraso tendem a diminuir a quantidade de esforço alocado para as atividades de garantia de qualidade (ABDEL-HAMID; MADNICK, 1991; ABDEL-HAMID; SENGUPTA; SWETT, 1999), o que contribui para que menos erros sejam detectados. Um comportamento inverso ocorre para o cenário otimista.

O gráfico de custo mostra que o cenário pessimista apresenta um custo maior que o cenário mais provável, que por sua vez apresenta um custo maior que o cenário otimista. Quando o prazo para o término da especificação é superestimado, tornando-se maior que o necessário, as curvas no gráfico de custo tendem a se aproximar, ou seja, os custos nos três cenários tendem a se igualar. De modo semelhante ao explicado nas seções 5.1.2 e 5.1.3, quando o prazo aumenta, grande parte do esforço utilizado para tratar o aumento da quantidade de retrabalho no cenário pessimista, é obtida a partir do esforço que deixaria de ser disponibilizado pela equipe para absorver uma parte do excesso de tempo disponível. Como o tempo em que a equipe ficaria ociosa será aproveitado para tratar o aumento da quantidade de retrabalho, o custo para o cenário pessimista se aproxima do custo para o cenário mais provável, quando o prazo aumenta.

No cenário otimista, em que há uma quantidade pequena de retrabalho, a quantidade de esforço necessário para o trabalho é menor e a equipe percebe uma folga maior no prazo planejado. Como a folga no prazo é maior, ocorrerá uma queda mais acentuada na produtividade da equipe para que seja absorvida uma parte do excesso de tempo disponível. Isso faz com que o custo para o cenário otimista se aproxime do custo para o cenário mais provável, quando o prazo aumenta.

Os resultados das simulações apresentadas permitem derivar a tendência das curvas de custo para os cenários. Como é mostrado na Figura 5.9, a tendência é que as curvas de custo para os três cenários apresentem formatos semelhantes, se for considerado que as decisões e as políticas gerenciais são as mesmas nos três cenários. A curva para o cenário pessimista deve estar localizada mais acima no gráfico, correspondendo a um custo maior, e a curva para o cenário otimista deve estar localizada mais abaixo no gráfico, correspondendo a um custo menor. Além disso, quando o prazo planejado aumenta demasiadamente, tornando-se maior que o

necessário, as curvas de custo tendem a se aproximar, o que foi explicado anteriormente nessa seção.



Fonte: Elaborado pelo Autor

Figura 5.9. Tendência das curvas de custo

Para cada cenário, há um prazo (ou intervalo de prazo) em que se obtém o menor custo para a realização da especificação. Na Figura 5.9, esse prazo é chamado “ponto de custo mínimo”. Ao reduzir ou aumentar o prazo, a tendência é que ocorra um aumento no custo. Como foi explicado na seção 5.1.1, ao planejar e incentivar o término da especificação em um prazo menor, pode ser necessário aumentar o tamanho da equipe e/ou utilizar horas-extras dos profissionais (ABDEL-HAMID, 1996; LIN; ABDEL-HAMID; SHERIF, 1997), o que implica no aumento do custo. Por outro lado, quando o prazo para o término da especificação é superestimado, a equipe percebe que o projeto está adiantado em relação ao planejamento. Ao perceber a folga no prazo, a equipe irá trabalhar com menor produtividade, absorvendo uma porção do excesso de tempo disponível (ABDEL-HAMID; MADNICK, 1991), o que provoca um aumento no custo.

Esse efeito do prazo sobre o custo pode ser explicado utilizando-se a mesma análise realizada no estudo dos efeitos dos avanços tecnológicos (PINDYCK; RUBINFELD, 2006), na Microeconomia. Considerando-se uma determinada quantidade de mão-de-obra, inicialmente, o alongamento do prazo reduz a pressão

sobre a equipe, o que diminui a quantidade de erros cometidos e de retrabalho no projeto. Consegue-se com isso, produzir mais com a mesma quantidade do insumo, no caso, a mão-de-obra. Isso é exatamente o que se convencionou chamar de avanço ou inovação tecnológica em Microeconomia. Contudo, ao aumentar demasiadamente o prazo, as mesmas pessoas se acomodam, o que faz com que ocorra algo similar a uma involução ou retrocesso tecnológico, em que se produz menos com a mesma quantidade de mão-de-obra.

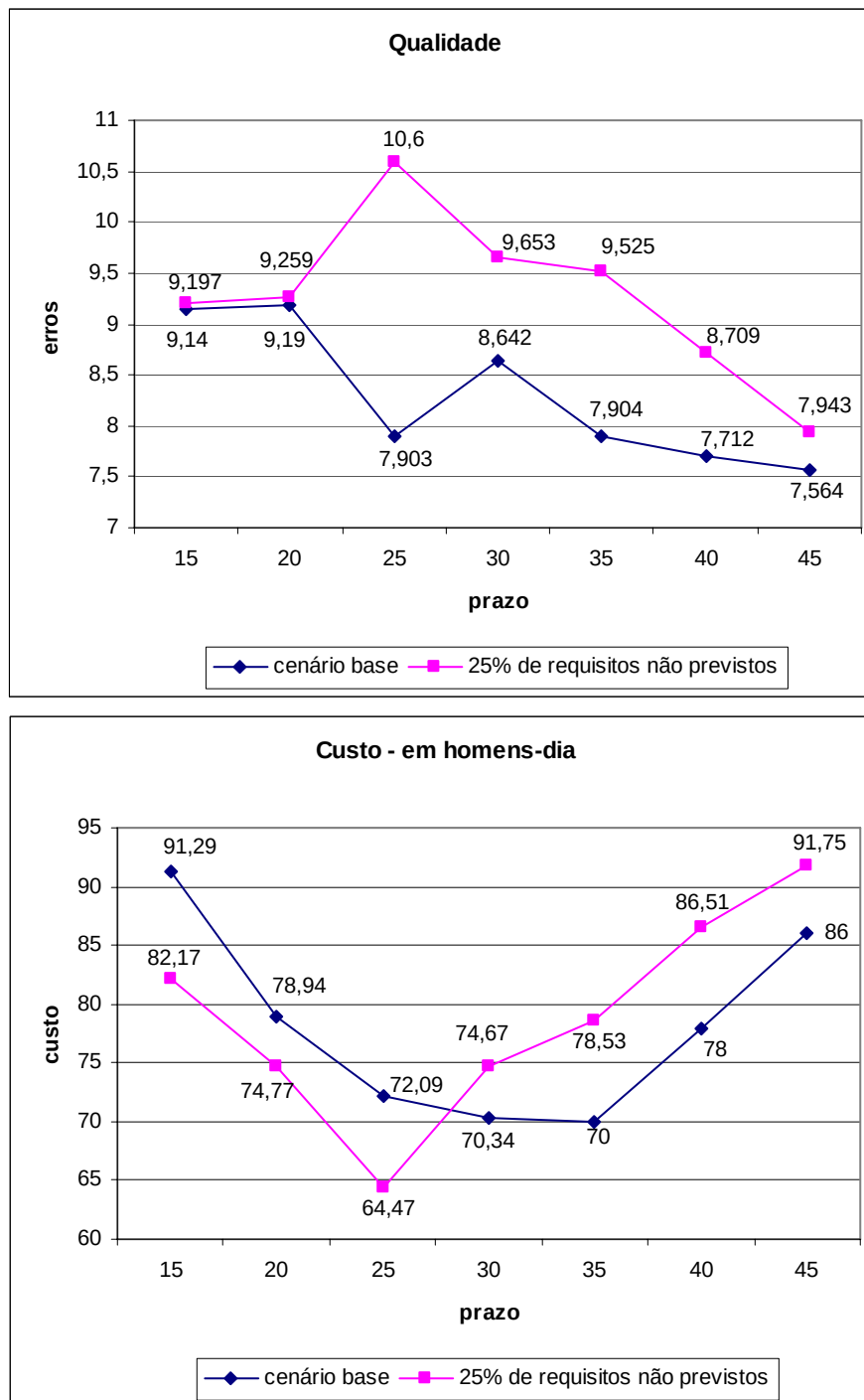
No cenário pessimista, o valor do custo mínimo tende a ser maior que no cenário mais provável, devido ao aumento na quantidade de retrabalho. Além disso, o ponto de custo mínimo no cenário pessimista é obtido em um determinado prazo planejado que é maior que o prazo correspondente ao custo mínimo no cenário mais provável. Isso ocorre porque diante do aumento na quantidade de retrabalho, para evitar o uso excessivo de esforço extra (o que poderia aumentar o custo), é necessário aumentar o prazo.

Um comportamento inverso ocorre no cenário otimista. Nesse cenário, o valor do custo mínimo tende a ser menor que no cenário mais provável. Além disso, o ponto de custo mínimo no cenário otimista é obtido em um determinado prazo planejado que é menor que o prazo correspondente ao custo mínimo no cenário mais provável. Isso ocorre porque diante de uma menor quantidade de retrabalho, para evitar que a equipe absorva uma porção do excesso de prazo trabalhando com menor produtividade (o que poderia aumentar o custo), é necessário reduzir o prazo.

Nas simulações apresentadas nessa seção ocorre exatamente o que foi explicado nos dois parágrafos anteriores. O gráfico de custo na Figura 5.8 mostra que o custo mínimo nos cenários otimista, mais provável e pessimista é 60, 70 e 82,93 homens-dia, respectivamente. E o custo mínimo para esses três cenários é obtido quando o prazo planejado é 30, 35 e 40 dias, respectivamente.

5.1.6 Erro na estimativa da quantidade de requisitos

Os valores nos gráficos apresentados na Figura 5.10 permitem comparar dois cenários: o cenário base em que não há erro na estimativa da quantidade de requisitos, e outro cenário em que 25% dos requisitos não são previstos inicialmente, que é um valor baixo de acordo com os intervalos definidos na Tabela 4.1.



Fonte: Elaborado pelo Autor

Figura 5.10. Impactos do erro na estimativa da quantidade de requisitos

O gráfico que compara a qualidade mostra que o erro na estimativa da quantidade de requisitos contribui para aumentar a quantidade de requisitos especificados incorretamente. O aumento no uso de esforço extra, que é necessário quando os requisitos não previstos são descobertos no decorrer da especificação, causa fadiga e cansaço na equipe, o que pode contribuir para aumentar a quantidade

de erros cometidos (COLLOFELLO *et al.*, 1998; LIN; ABDEL-HAMID; SHERIF, 1997).

O gráfico de custo mostra que quando o prazo planejado é menor ou igual a 25 dias, o erro na estimativa da quantidade de requisitos contribui para uma redução no custo. Isso pode ser explicado, analisando-se as variáveis envolvidas. Até que todos os requisitos sejam descobertos, a quantidade de requisitos que se pretende especificar é menor. Enquanto os demais requisitos não são descobertos, uma equipe pequena é suficiente para realizar a especificação. À medida que os requisitos não previstos são descobertos, torna-se necessário usar esforço extra da equipe. Se isso não for suficiente, novos profissionais podem ser alocados na equipe. No entanto, em geral, o uso de uma pequena quantidade de esforço extra é suficiente durante grande parte do tempo. A tendência é que a taxa com que os requisitos são descobertos seja inicialmente baixa e aumente com o progresso da especificação porque segundo (ABDEL-HAMID; MADNICK, 1991), com o tempo, o conhecimento da equipe sobre o sistema aumenta. Com isso, grande parte dos requisitos é descoberta próximo ao fim do prazo planejado. Nesse caso, o uso de esforço extra e o aumento do tamanho da equipe podem não ser mais suficientes para evitar o atraso, sendo necessário adiar o prazo.

Apesar de haver uma redução no custo porque a equipe se manteve pequena e foi utilizado pouco esforço extra, ocorre um atraso no término da especificação. Nas simulações realizadas, o erro na estimativa da quantidade de requisitos causou atrasos de até 4 dias para o término da especificação. A Tabela 5.2 permite comparar o tempo gasto para terminar a especificação, ao considerar os dois cenários.

Tabela 5.2. Tempo gasto para terminar a especificação em função do prazo planejado

Prazo planejado	15	20	25	30	35	40	45
Tempo gasto no cenário base	16	20	25	32	35	39	43
Tempo gasto no cenário em que 25% dos requisitos não são previstos	19	23	28	33	36	41	45

Fonte: Elaborado pelo Autor

Quando o prazo aumenta, as duas curvas no gráfico de custo se cruzam em algum ponto entre os prazos 25 e 30 dias, e então o erro na estimativa da quantidade de requisitos passa a contribuir para o aumento do custo. Até que todos os requisitos

sejam descobertos, a capacidade de trabalho da equipe é subutilizada porque o excesso aparente de prazo disponível é absorvido por meio de uma redução da produtividade. À medida que os demais requisitos são descobertos, é necessário aumentar o tamanho da equipe e utilizar esforço extra, o que aumenta o custo.

Para os demais cenários simulados e analisados na seção 5.1, o tempo gasto para terminar a especificação não diverge do tempo gasto no cenário base. Isso ocorre porque, como foi informado na seção 5.1.1, as variáveis do modelo foram configuradas para considerar um gerenciamento que evita alterar o prazo planejado, usando esforço extra dos profissionais e aumentando o tamanho da equipe, quando necessário. Por isso, uma tabela semelhante à Tabela 5.2 não aparece nas demais análises apresentadas na seção 5.1.

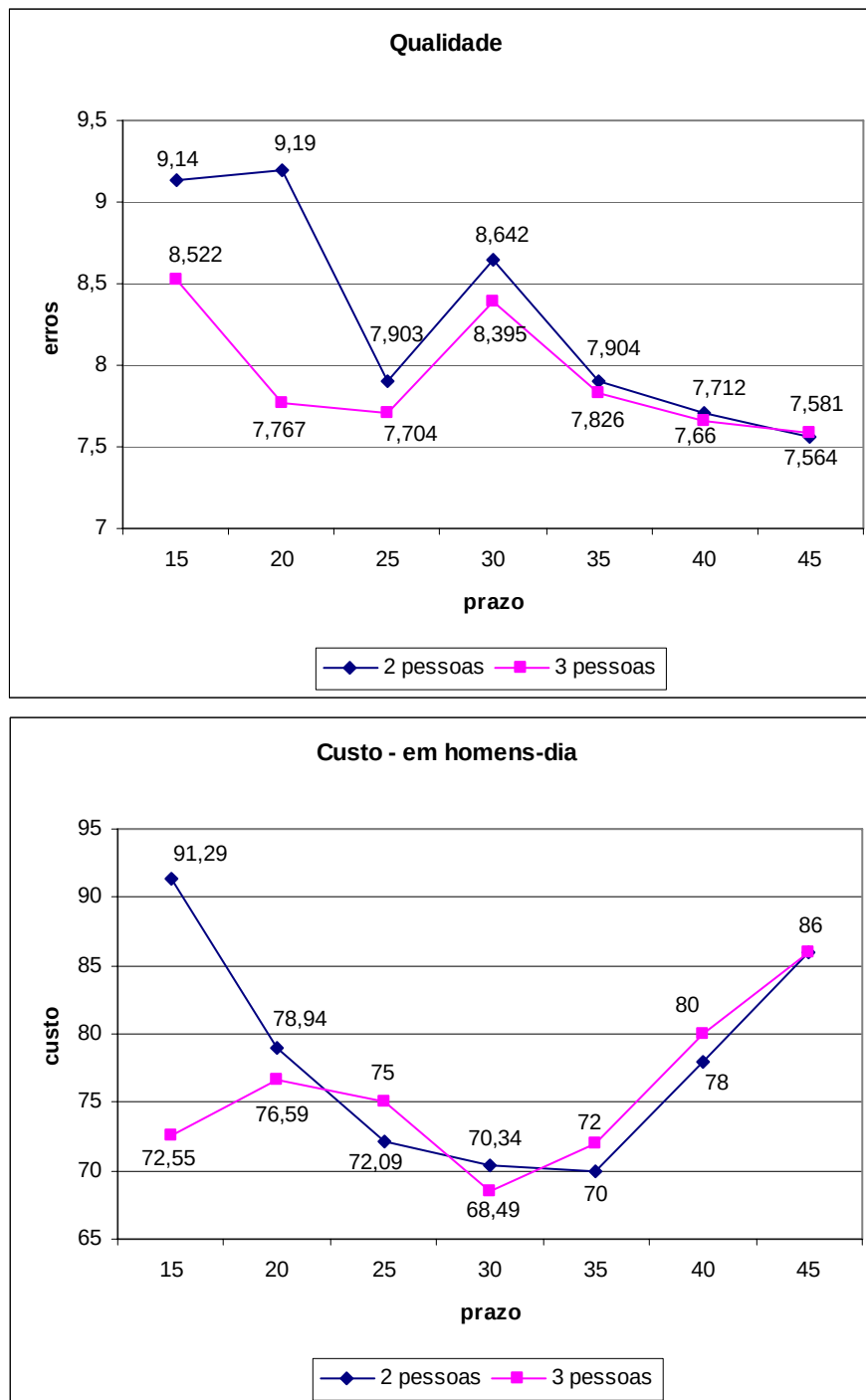
5.1.7 Modificações no tamanho da equipe

Os valores apresentados nos gráficos da Figura 5.11 permitem comparar dois cenários: o cenário base em que a especificação é iniciada com 2 pessoas na equipe, e outro cenário em que a especificação é iniciada com 3 pessoas na equipe.

O gráfico que compara a qualidade mostra que iniciar a especificação com uma equipe maior contribui para diminuir a quantidade de requisitos especificados incorretamente. O aumento no tamanho da equipe diminui a necessidade de utilizar esforço extra, o que diminui a taxa com que os erros são cometidos (COLLOFELLO *et al.*, 1998; LIN; ABDEL-HAMID; SHERIF, 1997).

As curvas no gráfico de custo se cruzam em três pontos. Esses pontos de cruzamento devem ser analisados pelos tomadores de decisão, pois neles ocorre uma inversão nos efeitos das decisões gerenciais. Para um prazo menor ou igual a 20 dias, o gráfico de custo mostra que iniciar a especificação com 3 pessoas na equipe contribui para que o custo seja menor. Isso ocorre porque diante de um prazo menor, iniciar o trabalho com uma equipe maior evita que novos profissionais precisem ser alocados no decorrer da especificação. Além disso, a quantidade de esforço extra utilizado é menor e não chega a causar exaustão da equipe.

Mas, quando o prazo aumenta para 35 dias ou mais, tornando-se maior que o necessário, iniciar a especificação com 3 pessoas na equipe tende a aumentar o custo. Com uma equipe maior, a folga no prazo torna-se mais evidente. Isso faz com que a queda na produtividade, para absorver uma parte do excesso de tempo disponível, seja mais acentuada, aumentando o custo.



Fonte: Elaborado pelo Autor

Figura 5.11. Impactos de modificações no tamanho da equipe

Quando as curvas, em um gráfico que compara os efeitos de decisões gerenciais, se cruzam em um único ponto, o tomador de decisão entende que da extremidade esquerda do gráfico até o ponto de cruzamento, uma determinada decisão gerencial é mais adequada; e a partir do ponto até a extremidade direita do gráfico, uma outra decisão gerencial é mais adequada. No entanto, ao organizar os resultados das simulações em um gráfico, as oscilações nas curvas podem criar mais

de um ponto de cruzamento. Isso ocorre com o gráfico de custo na Figura 5.11. Os cruzamentos entre as curvas no gráfico indicam que para o prazo de 25 dias, a decisão de iniciar a especificação com 3 pessoas provoca o aumento do custo. Quando o prazo é 30 dias, essa decisão contribui para reduzir o custo. E ao aumentar o prazo para 35 dias ou mais, essa decisão volta a aumentar o custo. Ao analisar os efeitos de diferentes decisões gerenciais, os tomadores de decisão devem estar atentos às oscilações nos gráficos e aos pontos de cruzamento que podem ser criados.

Ao simular o modelo considerando um prazo de 25 dias, foi verificado que no cenário em que a especificação é iniciada com 3 pessoas na equipe, a equipe permanece com 3 pessoas até o término da especificação. No cenário base, em que a especificação é iniciada com 2 pessoas na equipe, o tamanho da equipe aumenta para 3 no quarto dia de trabalho e assim permanece até o término da especificação. Como no cenário base o tamanho da equipe é menor nos três primeiros dias de trabalho, é necessário utilizar uma quantidade maior de esforço extra. No entanto, quase todo o esforço extra utilizado é obtido durante o horário normal de trabalho da equipe, sem exigir o uso de horas-extras. Isso contribui para que o custo no cenário base seja menor, quando o prazo planejado é 25 dias.

Ao simular o modelo considerando um prazo de 30 dias, foi verificado que no cenário em que a especificação é iniciada com 3 pessoas na equipe, o tamanho da equipe diminui para 2 no nono dia de trabalho e assim permanece até o término da especificação. No cenário base, em que a especificação é iniciada com 2 pessoas na equipe, o tamanho da equipe aumenta para 3 no 27º dia de trabalho, e no 28º dia aumenta para 4. Como a equipe permanece com 2 pessoas até o 27º dia, é necessário utilizar uma quantidade maior de esforço extra, que nesse caso chega a causar exaustão da equipe no 24º dia, contribuindo para que a especificação seja completada somente no 32º dia de trabalho. A necessidade de utilizar uma grande quantidade de esforço extra, causando a exaustão da equipe, e o aumento no tamanho da equipe próximo ao término da especificação contribuem para que o custo no cenário base seja maior, quando o prazo planejado é 30 dias.

5.1.8 Alocação de esforço para as atividades de garantia de qualidade

Para verificar os impactos da alocação de diferentes quantidades de esforço para as atividades de garantia de qualidade foram simulados três cenários: o cenário base em que a porcentagem desejada de homens-dia alocados para garantia de

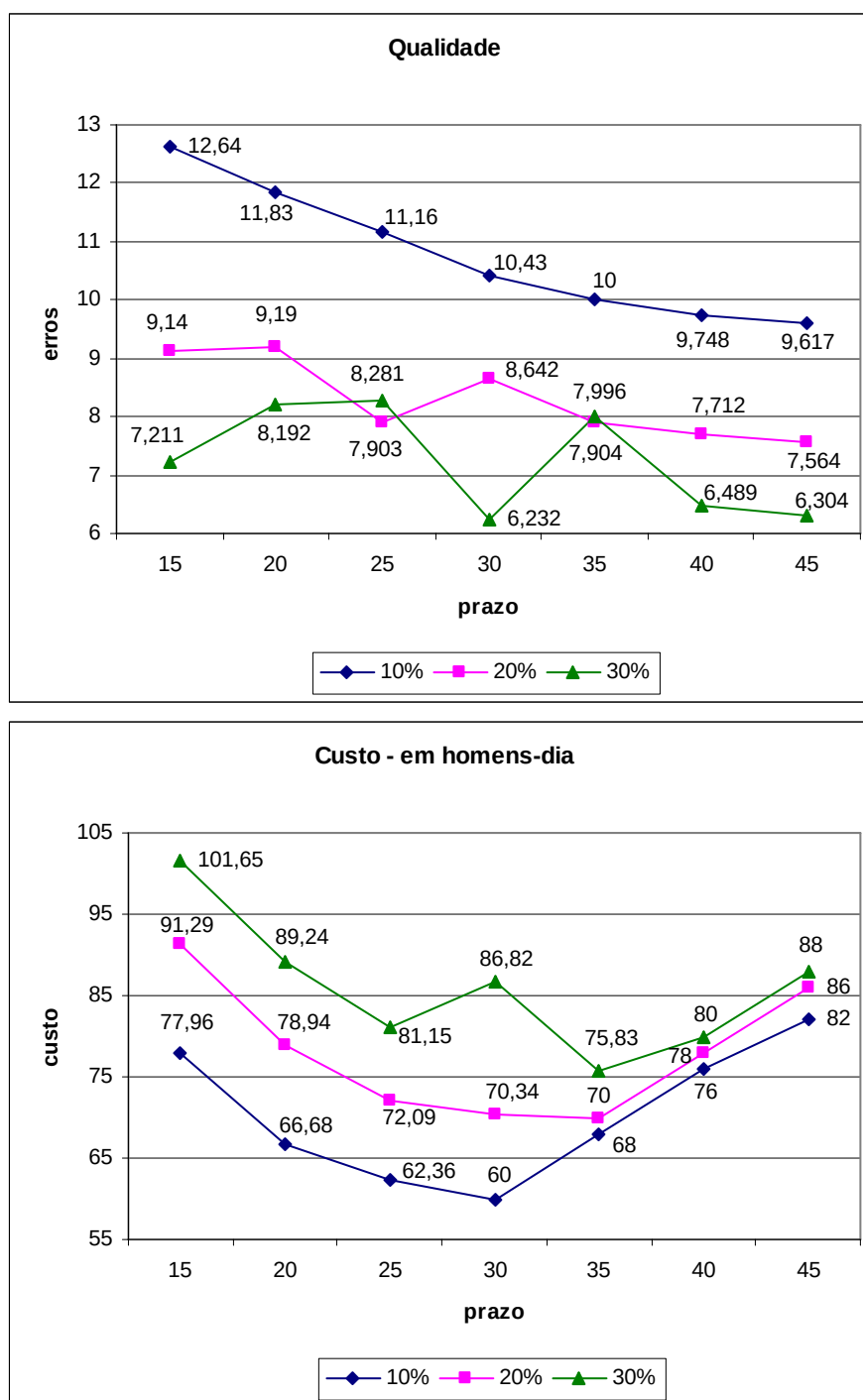
qualidade é 20%, um cenário em que essa porcentagem é 10%, e outro cenário em que essa porcentagem é 30%. Os valores são baseados nos intervalos de valores definidos na Tabela 4.1.

Os gráficos apresentados na Figura 5.12 mostram a qualidade e o custo da especificação de requisitos em função do prazo planejado. O gráfico que compara a qualidade mostra que o aumento na quantidade de esforço alocado para as atividades de garantia de qualidade tende a provocar uma redução na quantidade de requisitos especificados incorretamente. Isso é o esperado, porque os erros detectados são imediatamente corrigidos pela equipe.

Os formatos das curvas no gráfico de custo são parecidos, o que é esperado, pois as variáveis que determinam as demais decisões gerenciais recebem os mesmos valores nos três cenários simulados. O gráfico de custo mostra que aumentar a quantidade de esforço alocado para as atividades de garantia de qualidade provoca o aumento do custo. Isso ocorre porque o esforço utilizado em atividades de garantia de qualidade não contribui para o progresso do projeto. Além disso, o aumento na quantidade de esforço alocado para garantia de qualidade contribui para aumentar a quantidade de erros detectados. A gerência deve alocar esforço adicional para realizar as correções, o que também aumenta o custo.

Quando o prazo para o término da especificação aumenta, tornando-se maior que o necessário, as curvas no gráfico de custo tendem a se aproximar, ou seja, os custos nos três cenários tendem a se igualar. Quando o prazo é maior que o necessário e a porcentagem desejada de esforço alocado para garantia de qualidade aumenta de 20% para 30%, a equipe utiliza parte do tempo ocioso para realizar as atividades de garantia de qualidade, sem aumentar o custo da especificação.

Quando a porcentagem desejada de esforço alocado para garantia de qualidade diminui para 10%, a quantidade de esforço utilizado no progresso do projeto é maior e a equipe percebe uma folga maior no prazo planejado. Como a folga no prazo é maior, ocorrerá uma queda mais acentuada na produtividade da equipe para que seja absorvida uma parte do excesso de tempo disponível. Isso faz com que se aproximem, quando o prazo aumenta, os custos obtidos nos cenários em que se deseja que 10% e 20% do esforço sejam alocados para garantia de qualidade.



Fonte: Elaborado pelo Autor

Figura 5.12. Variação na quantidade de esforço alocado para as atividades de garantia de qualidade

5.1.9 Uso de esforço extra

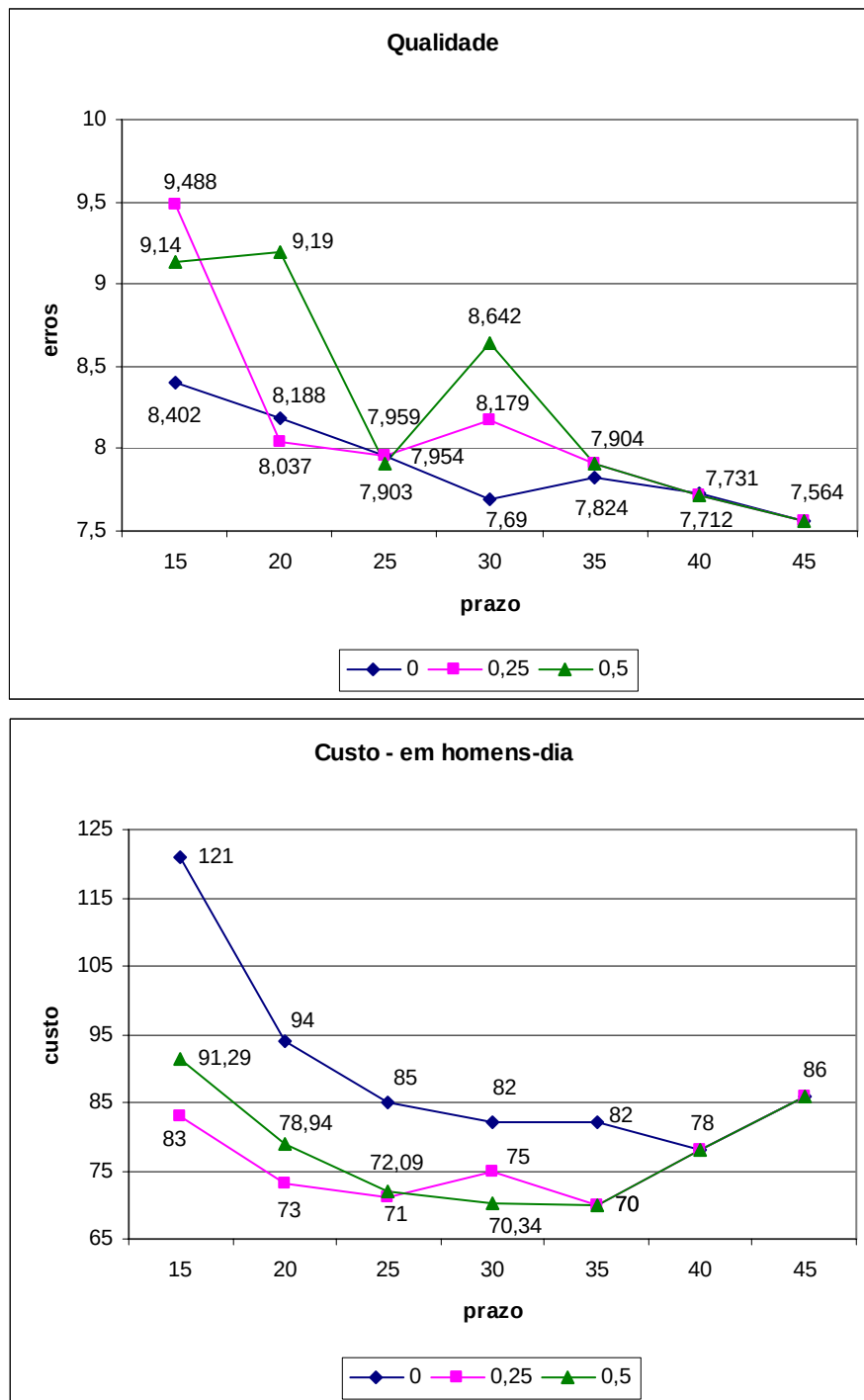
Visando a verificar os impactos das diferentes políticas gerenciais acerca da quantidade de esforço extra utilizado quando há risco de atraso no projeto, foram simulados três cenários: o cenário base em que a quantidade máxima de homens-dia extras disponibilizados por um membro experiente é 0,5 homens-dia (ou seja, os

membros da equipe são incentivados a disponibilizar até 50% a mais de esforço), um cenário em que esse valor é 0,25 homens-dia (até 25% a mais de esforço), e outro cenário em que esse valor é 0 homens-dia (não há uso de esforço extra). Os valores são baseados nos intervalos de valores definidos na Tabela 4.1.

Os gráficos apresentados na Figura 5.13 mostram a qualidade e o custo da especificação de requisitos em função do prazo planejado. As curvas no gráfico que compara a qualidade possuem algumas oscilações. No entanto, é possível observar que o aumento na quantidade de esforço extra utilizado tende a provocar o aumento da quantidade de requisitos especificados incorretamente. A sobrecarga de trabalho pode causar estresse e fadiga da equipe, o que aumenta a quantidade de erros cometidos (COLLOFELLO *et al.*, 1998; LIN; ABDEL-HAMID; SHERIF, 1997).

O gráfico de custo mostra que utilizar uma quantidade média (até 0,25 homens-dia) de esforço extra contribui para obter um menor custo. Ao utilizar uma quantidade excessiva de esforço extra, pode ocorrer fadiga e estresse da equipe, o que pode provocar uma queda brusca na produtividade (HANNE; NEU, 2004; COLLOFELLO *et al.*, 1998). Em seguida, para evitar o atraso, pode ser necessário alocar novos profissionais na equipe e utilizar esforço extra novamente, aumentando o custo. Por outro lado, utilizar uma quantidade pequena de esforço extra também provoca o aumento do custo. Durante o trabalho, os profissionais utilizam parte de seu esforço em tarefas pessoais, como ver e-mails e conversar com os colegas (ABDEL-HAMID; MADNICK, 1991). Quando há pressões de prazo e não é utilizado esforço extra dos profissionais, eles continuam desperdiçando parte de seu esforço em suas atividades pessoais, aumentando o custo.

As curvas no gráfico de custo possuem formatos semelhantes e quando o prazo aumenta para 40 dias ou mais, as curvas se sobrepõem. Isso ocorre porque quando o prazo aumenta demasiadamente, não é necessário utilizar horas-extras dos profissionais e o custo passa a depender somente do tamanho da equipe, que é o mesmo nos três cenários simulados.



Fonte: Elaborado pelo Autor

Figura 5.13. Variação na quantidade de esforço extra utilizado

5.2 Ajustando o modelo com dados coletados em uma empresa de desenvolvimento de software

A empresa que forneceu os dados utilizados para configurar o modelo e realizar as simulações apresentadas nessa seção desenvolve sistemas de software para uso no domínio científico e tecnológico, e sua sede fica em Viçosa-MG. Para

coletar os dados foi elaborado um questionário em forma de tabela, cujos resultados são apresentados na Tabela 5.3.

Tabela 5.3. Questionário para coleta de dados acerca da fase de requisitos

Item	Baixo	Médio (Mais Provável)	Alto
Quantidade de requisitos especificados e entregues em uma liberação.	Não foi possível estimar	Não foi possível estimar	Não foi possível estimar
Tamanho inicial da equipe responsável por especificar os requisitos.	2	2	3
Produtividade média de um membro experiente da equipe (em requisitos especificados / homem-dia e pontos de função / homem-dia).	Não foi possível estimar	Não foi possível estimar	Não foi possível estimar
Probabilidade de cometer erros durante a especificação dos requisitos.	5%	15%	20%
Probabilidade de alteração de requisitos.	10%	20%	30%
Porcentagem dos requisitos de uma liberação que não são previstos antes de iniciar a especificação.	15%	20%	25%
Porcentagem do esforço disponibilizado pela equipe que é alocado para as atividades de garantia de qualidade, como revisões e inspeções.	5%	10%	15%
Aumento no esforço disponibilizado pelos membros da equipe quando há pressões de prazo e risco de atraso.	10%	15%	20%

Fonte: Elaborado pelo Autor

A empresa consultada não mantém uma base de dados acerca dos projetos realizados. Os valores informados na Tabela 5.3 foram estimados por um Diretor e por um Gerente de Projetos, a partir da experiência obtida com a participação em projetos desenvolvidos pela empresa. Para cada item contido no questionário, os profissionais foram instruídos a responder o valor mais provável, um valor baixo e um valor alto, ao considerar um projeto de tamanho médio. De acordo com a escala de tamanho de projetos de software, apresentada em (FILHO, 2003), um projeto de tamanho médio possui entre 1000 e 10000 linhas de código. Na empresa consultada, o tempo necessário para o desenvolvimento de um projeto de tamanho médio varia, em geral, de 12 a 18 meses.

5.2.1 Definição e simulação de um cenário base

Seguindo a mesma abordagem utilizada na seção 5.1, inicialmente foi definido e simulado um cenário base em que não é considerada a ocorrência de nenhum risco. Em seguida, outros cenários foram estabelecidos e simulados, e os resultados das simulações desses cenários foram comparados com os obtidos com a simulação do cenário base. Para configurar o cenário base foram utilizados os valores contidos na coluna *Médio (Mais Provável)* da Tabela 5.3.

Os valores das demais variáveis do painel de controle, que não podem ser definidos com os valores informados na Tabela 5.3, foram definidos a partir da experiência dos autores e de trocas de experiências não documentadas com profissionais de empresas de desenvolvimento de software. Isso não prejudica a realização de comparações entre os cenários apresentados, pois os valores atribuídos a essas variáveis são os mesmos em todos os cenários simulados. As Figuras 5.14 e 5.15 mostram o painel de controle do modelo. Na Figura 5.14 é possível verificar como as variáveis do modelo foram ajustadas para simular o cenário base. E na Figura 5.15 são apresentados os valores obtidos para as variáveis utilizadas nas análises realizadas após a simulação.

Os valores atribuídos às variáveis *tempo para aumento do uso de esforço extra*, *tempo para aumento do prazo* e *tempo para aumento do tamanho da equipe* indicam que a simulação irá considerar um gerenciamento que, diante de riscos de atraso no projeto, tende a aumentar o uso do esforço extra e evitar modificações no tamanho da equipe e no prazo. Se o uso de esforço extra não for suficiente para evitar o atraso, os valores atribuídos às variáveis indicam que a segunda opção da gerência é aumentar o tamanho da equipe. Modificações no prazo só serão feitas se forem estritamente necessárias.

Configuração			
máximo homens-dia extras disponibilizados sem custo adicional	nível máximo de exaustão suportada	tamanho médio de cada requisito em pontos de função	quantidade de requisitos especificados por homem-dia
máximo homens-dia não disponibilizados por um membro experiente	diminuição da produtividade após a exaustão	Requisitos Entregues Especificados	quantidade de alterações realizadas por homem-dia
porcentagem do excesso de prazo absorvido	multiplicador para a exaustão causada pelo aumento da jornada de trabalho	Requisitos Entregues com Erros	quantidade de erros detectados por homem-dia
relação: produtividade de um novato / produtividade de um experiente	Requisitos da Liberação	quantidade de erros descobertos por dia	quantidade de falhas corrigidas por homem-dia

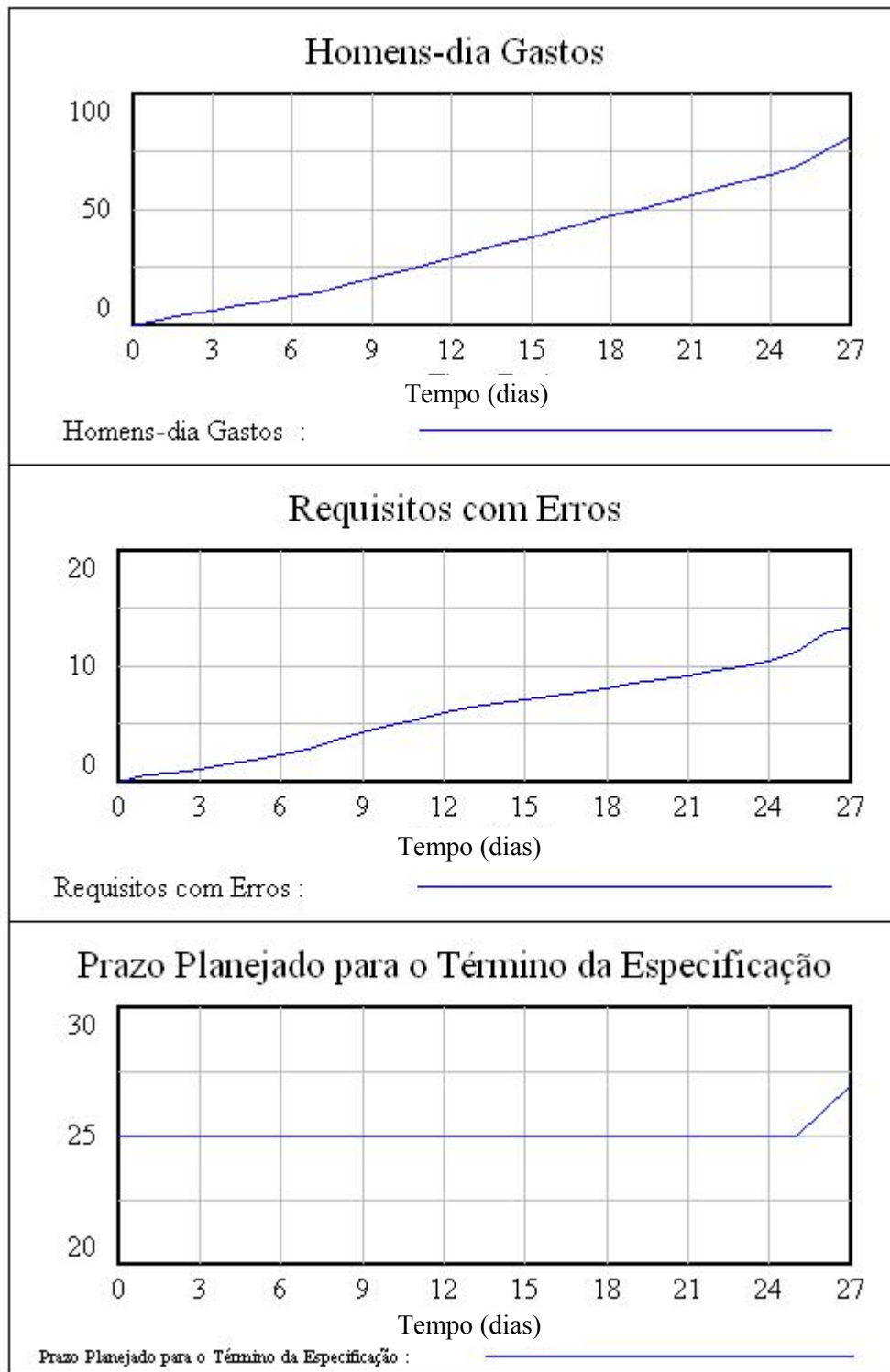
Análise - simular riscos			
porcentagem de requisitos não previstos	probabilidade de alteração de requisitos	probabilidade de alteração de requisitos entregues	probabilidade de cometer erros
atenuação para erros durante a correção e alteração	quantidade de profissionais que abandonam a equipe	tempo em que ocorre turnover	

Análise - determinar decisões e políticas gerenciais			
quantidade inicial de pessoas na equipe	porcentagem desejada de homens-dia alocados para garantia de qualidade	tempo desejado para realizar as alterações	tempo para aumento do prazo
quantidade mínima de pessoas na equipe	porcentagem da falta de homens-dia tratada com esforço extra	tempo desejado para realizar as correções	tempo para redução do prazo
força de trabalho reserva	máximo homens-dia extras disponibilizados por um membro experiente	tempo para aumento do uso de esforço extra	tempo para aumento do tamanho da equipe
Prazo (inicial) Planejado para o Término da Especificação	quantidade máxima de novatos que um membro experiente pode treinar	tempo para redução do uso de esforço extra	tempo para redução do tamanho da equipe

Fonte: Elaborado pelo Autor

Figura 5.14. Painel de controle – entrada de valores

Análise - verificação após a simulação



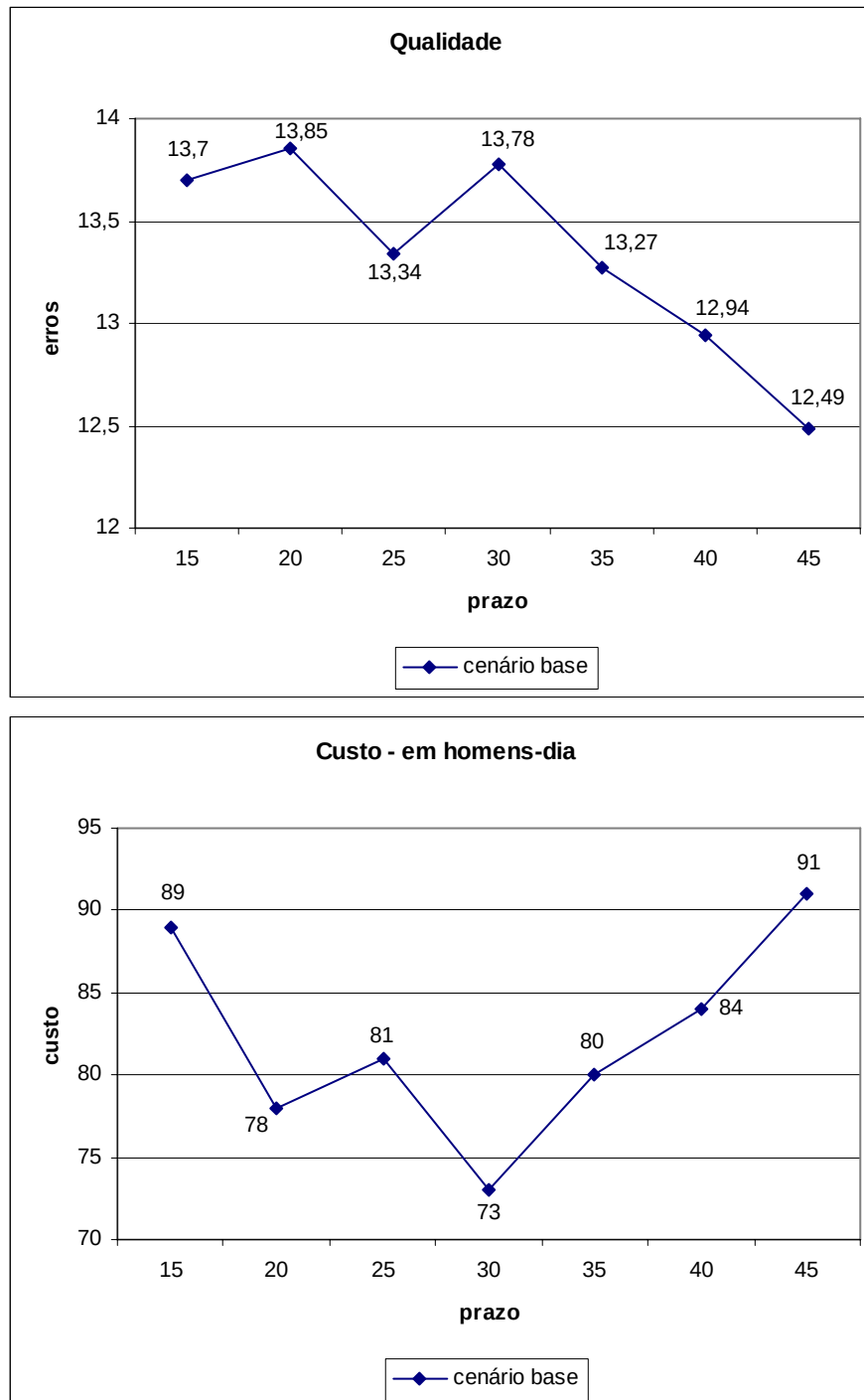
Observação: Os valores apresentados nos gráficos são os valores acumulados nos respectivos estoques em um determinado tempo, medido em dias, após o início da especificação.

Fonte: Elaborado pelo Autor

Figura 5.15. Painel de controle – saída de valores

Os gráficos apresentados na Figura 5.16 mostram a qualidade (valor do estoque *Requisitos com Erros*) e o custo (valor do estoque *Homens-dia Gastos*) da

especificação de requisitos em função do prazo planejado. Os valores nos gráficos foram obtidos ao simular o modelo considerando o cenário base descrito.



Fonte: Elaborado pelo Autor

Figura 5.16. Qualidade e custo obtidos com o cenário base

O gráfico que compara a qualidade mostra que a redução no prazo planejado tende a aumentar a quantidade de requisitos especificados incorretamente. Segundo (ABDEL-HAMID, 1996) e (LIN; ABDEL-HAMID; SHERIF, 1997), para terminar a

especificação em um tempo menor, pode ser necessário aumentar o tamanho da equipe e utilizar esforço extra dos profissionais, o que contribui para aumentar a quantidade de erros cometidos (COLLOFELLO *et al.*, 1998; LIN; ABDEL-HAMID; SHERIF, 1997).

Quando o prazo planejado é 30 dias, a quantidade de erros de especificação é maior se comparada à quantidade de erros quando o prazo é 25 ou 35 dias, o que provoca uma oscilação na curva do gráfico de qualidade. Ao simular o modelo considerando um prazo de 30 dias, as decisões gerenciais configuradas no modelo contribuem para o uso de uma maior quantidade de esforço extra, evitando o aumento do tamanho da equipe. O aumento no uso do esforço extra contribui para aumentar a quantidade de erros cometidos (COLLOFELLO *et al.*, 1998; LIN; ABDEL-HAMID; SHERIF, 1997), o que explica o aumento na quantidade de requisitos especificados incorretamente quando o prazo planejado é 30 dias, se comparado com os resultados obtidos quando o prazo é 25 ou 35 dias.

O gráfico de custo mostra que um prazo em torno de 30 dias estabelece o menor custo. Ao planejar e incentivar o término da especificação em um prazo menor, pode ser necessário aumentar o tamanho da equipe e/ou utilizar esforço extra dos profissionais (ABDEL-HAMID, 1996; LIN; ABDEL-HAMID; SHERIF, 1997), o que implica no aumento do custo. Por outro lado, quando o prazo para o término da especificação é superestimado, a equipe percebe que o projeto está adiantado em relação ao planejamento. Ao perceber a folga no prazo, a equipe irá trabalhar com menor produtividade, absorvendo uma parte do excesso de tempo disponível (ABDEL-HAMID; MADNICK, 1991). Isso explica o aumento que ocorre no custo quando o prazo planejado é maior que 30 dias.

Na simulação do cenário base, em que 20% dos requisitos não são previstos inicialmente, ocorrem atrasos de até 3 dias para o término da especificação. A Tabela 5.4 mostra o tempo gasto em função do prazo planejado.

Tabela 5.4. Tempo gasto para terminar a especificação em função do prazo planejado

Prazo planejado	15	20	25	30	35	40	45
Tempo gasto no cenário base em que 20% dos requisitos não são previstos	17	22	27	33	37	40	44

Fonte: Elaborado pelo Autor

Se forem comparadas as curvas dos gráficos apresentados nessa seção com as curvas dos gráficos apresentados na seção 5.1.1, pode-se perceber que, para o cenário base, as curvas correspondentes apresentam a mesma tendência de variação em função do prazo planejado. Ambos os gráficos de qualidade, apresentados nas Figuras 5.3 e 5.16, mostram que se o prazo for reduzido, há uma tendência de aumento na quantidade de requisitos especificados incorretamente. E ambos os gráficos de custo, também apresentados nas Figuras 5.3 e 5.16, mostram que existe um prazo em torno do qual se obtém um custo mínimo, e se o prazo for aumentado ou reduzido, há uma tendência de ocorrer aumento no custo para a realização da especificação dos requisitos.

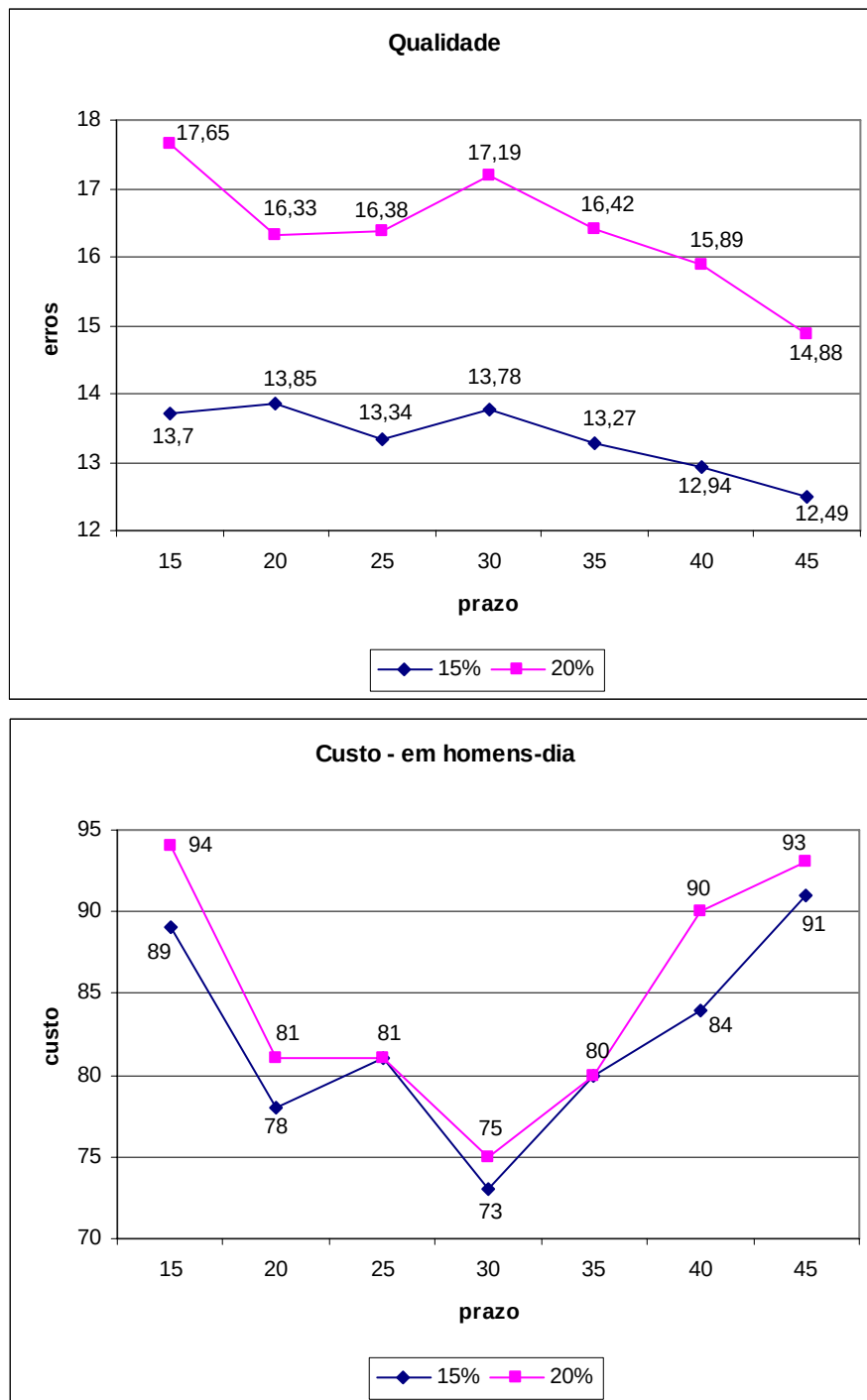
Ao analisar os gráficos apresentados nas seções seguintes (5.2.2 a 5.2.9), pode-se verificar que, diante da ocorrência de riscos ou da implementação de diferentes políticas gerenciais, o comportamento do modelo ao ajustá-lo utilizando-se dados coletados em uma empresa de desenvolvimento de software continua semelhante ao comportamento obtido quando foram utilizados dados disponibilizados na literatura.

As seções seguintes apresentam outros cenários, que consideram a ocorrência de riscos e permitem comparar os efeitos da implementação de diferentes políticas gerenciais. Para os cenários simulados nas seções seguintes, as variáveis que não possuem seus valores explicitamente informados recebem os mesmos valores definidos para a simulação do cenário base. Esses valores podem ser visualizados na Figura 5.14, que mostra o painel de controle configurado para simular o cenário base.

5.2.2 Aumento na quantidade de erros cometidos

Os gráficos apresentados na Figura 5.17 mostram a qualidade e o custo da especificação de requisitos em função do prazo planejado. Os valores permitem comparar dois cenários: o cenário base em que a probabilidade de cometer erros é 15%, e outro cenário em que a probabilidade de cometer erros é 20%, que é um valor alto de acordo com os dados informados na Tabela 5.3.

Em um projeto real, o aumento na quantidade de erros pode ser provocado por diversos fatores, tais como: dificuldade e complexidade de um determinado projeto, inexperiência e menor capacidade dos profissionais alocados na equipe, baixa confiabilidade das ferramentas utilizadas, etc.



Fonte: Elaborado pelo Autor

Figura 5.17. Aumento na quantidade de erros cometidos

Os formatos das curvas nos gráficos são parecidos, o que é esperado, pois as variáveis que determinam as decisões gerenciais recebem os mesmos valores nos dois cenários simulados. O gráfico que compara a qualidade mostra que o aumento na quantidade de erros cometidos contribui para o aumento da quantidade de requisitos especificados incorretamente. Esse comportamento também ocorre com as

simulações apresentadas na seção 5.1.2, que utilizaram dados disponibilizados na literatura para ajustar o modelo.

O gráfico de custo mostra que o aumento na quantidade de erros cometidos provoca o aumento do custo, o que também ocorre com as simulações apresentadas na seção 5.1.2.

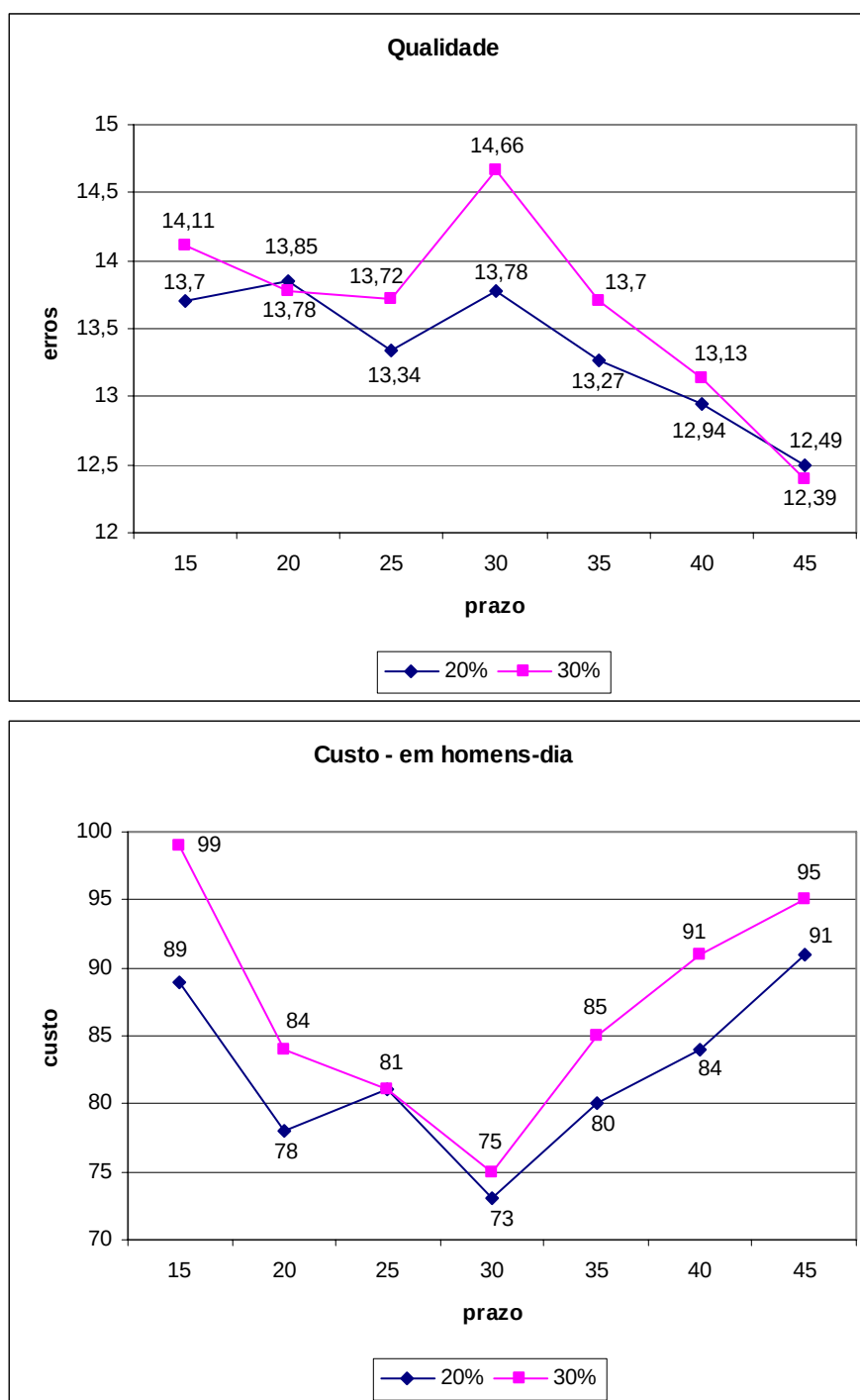
5.2.3 Aumento na volatilidade dos requisitos

Os valores apresentados nos gráficos da Figura 5.18 permitem comparar dois cenários: o cenário base em que a probabilidade de alteração de requisitos é 20%, e outro cenário em que a probabilidade de alteração de requisitos é 30%, que é um valor alto de acordo com os dados informados na Tabela 5.3.

De modo semelhante ao ocorrido na seção 5.2.2, os formatos das curvas nos gráficos são parecidos. A semelhança entre os formatos das curvas é esperada porque as variáveis que determinam as decisões gerenciais recebem os mesmos valores nos dois cenários simulados.

O gráfico que compara a qualidade mostra que o aumento na volatilidade dos requisitos tende a provocar um aumento na quantidade de requisitos especificados incorretamente. Um comportamento semelhante ocorre com as simulações apresentadas na seção 5.1.3, que utilizaram dados disponibilizados na literatura para ajustar o modelo. Quando o prazo planejado é 20 ou 45 dias, o aumento na volatilidade dos requisitos provoca uma pequena redução na quantidade de requisitos especificados incorretamente. Ao simular o cenário em que a volatilidade dos requisitos é 30%, considerando um prazo de 20 ou 45 dias, as decisões gerenciais configuradas no modelo contribuem para aumentar o tamanho da equipe, reduzindo a quantidade de esforço extra utilizado, o que acarreta uma redução na quantidade de erros cometidos.

O gráfico de custo mostra que o aumento na volatilidade dos requisitos provoca o aumento do custo, o que também ocorre com as simulações apresentadas na seção 5.1.3.



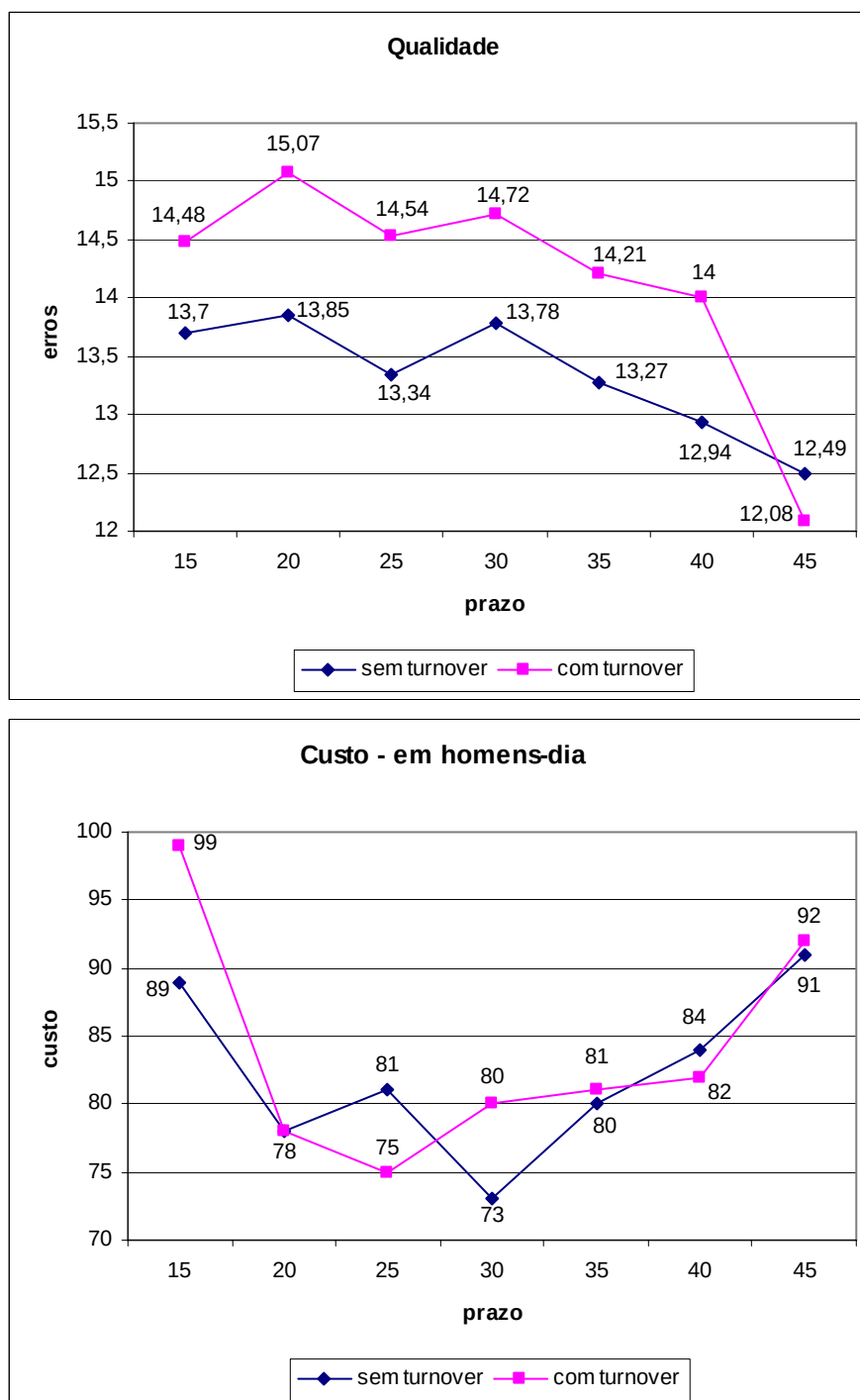
Fonte: Elaborado pelo Autor

Figura 5.18. Aumento na volatilidade dos requisitos

5.2.4 Turnover de pessoal

Os valores nos gráficos apresentados na Figura 5.19 permitem comparar dois cenários: o cenário base em que não ocorre *turnover*, e outro cenário em que ocorre *turnover*. Nesse caso, o *turnover* corresponde a uma pessoa abandonar a equipe

quando o tempo decorrido do início da especificação for igual à metade do prazo planejado.



Fonte: Elaborado pelo Autor

Figura 5.19. Impactos da ocorrência de *turnover* quando o tempo decorrido do início da especificação for igual à metade do prazo planejado

O gráfico que compara a qualidade mostra que quando ocorre *turnover*, há um aumento na quantidade de requisitos especificados incorretamente, exceto quando o prazo planejado é igual a 45 dias. Um comportamento semelhante ocorre com as

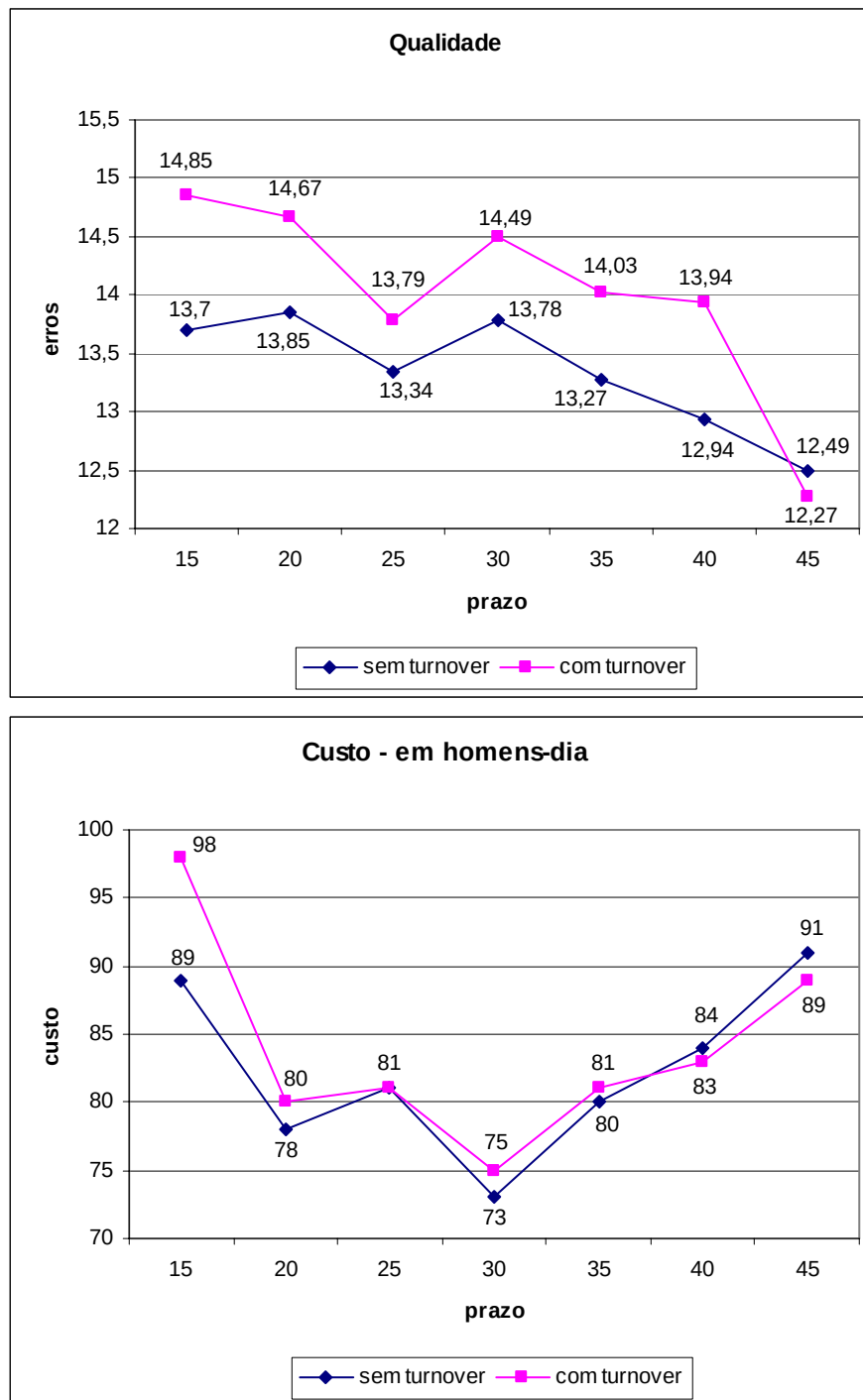
simulações apresentadas na seção 5.1.4, que utilizaram dados disponibilizados na literatura para ajustar o modelo.

As curvas no gráfico de custo se cruzam em diversos pontos, dificultando a realização de uma análise acerca dos efeitos do *turnover* sobre o custo da especificação dos requisitos. Os dois cenários simulados consideram que 20% dos requisitos não são previstos inicialmente. Nesse caso, até que todos os requisitos sejam descobertos, a quantidade de requisitos que se pretende especificar é menor. Então, a ocorrência de *turnover* quando o tempo decorrido do início da especificação for igual à metade do prazo planejado pode não ter um impacto sobre o custo porque a equipe tende a ficar folgada a partir desse momento (próximo à metade do prazo planejado) até próximo ao término da especificação, quando os demais requisitos são descobertos.

O *turnover* pode ter um impacto maior sobre o custo se o mesmo ocorrer quando o tempo decorrido do início da especificação for um pouco menor que a metade do prazo planejado ou em um tempo próximo ao término da especificação, quando os demais requisitos são descobertos. Para verificar essa hipótese, a Figura 5.20 apresenta os gráficos que permitem comparar dois cenários: o cenário base em que não ocorre *turnover*, e outro cenário em que ocorre *turnover*. Nesse caso, o *turnover* corresponde a uma pessoa abandonar a equipe quando o tempo decorrido do início da especificação for igual a 40% do prazo planejado, ou seja, um pouco antes da metade do prazo planejado. Por exemplo, se o prazo planejado for igual a 25 dias, o *turnover* corresponde a uma pessoa abandonar a equipe no décimo dia de trabalho.

O gráfico que compara a qualidade mostra que quando ocorre *turnover*, há um aumento na quantidade de requisitos especificados incorretamente, exceto quando o prazo planejado é igual a 45 dias. Um comportamento semelhante ocorre com as simulações apresentadas na seção 5.1.4, que utilizaram dados disponibilizados na literatura para ajustar o modelo.

O gráfico de custo mostra que a ocorrência de *turnover* aumenta o custo quando o prazo planejado é menor ou igual a 35 dias. No entanto, quando o prazo aumenta, as duas curvas se cruzam em um determinado ponto no gráfico e então o *turnover* passa a contribuir para uma redução no custo. Esse comportamento também ocorre com as simulações apresentadas na seção 5.1.4.

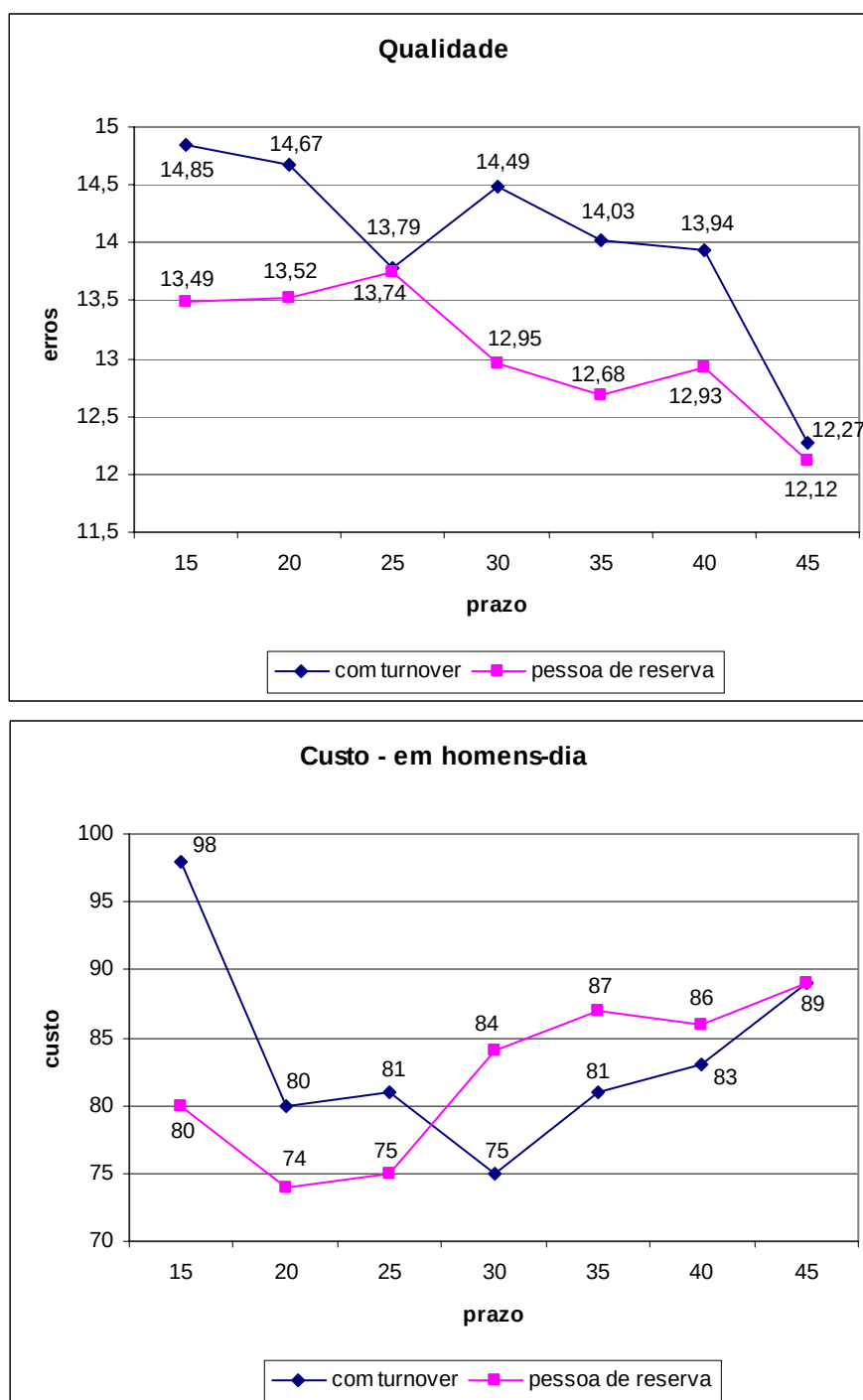


Fonte: Elaborado pelo Autor

Figura 5.20. Impactos da ocorrência de *turnover* quando o tempo decorrido do início da especificação for igual a 40% do prazo planejado

Na Figura 5.21, os gráficos comparam um cenário em que ocorre *turnover* com outro cenário em que a ocorrência de *turnover* é combinada com uma ação utilizada para mitigar o risco de *turnover* (DEMARCO; LISTER, 2003). Essa ação consiste em manter a equipe com um tamanho maior que o necessário. Nesse caso, se algum integrante abandonar a equipe, há outro profissional, que já conhece o domínio do

problema, disponível para assumir suas tarefas. Para simular o cenário em que essa ação gerencial é utilizada, as variáveis do modelo foram ajustadas para considerar que a gerência mantém uma pessoa de reserva na equipe, a partir do início da especificação e até a ocorrência do *turnover*.



Fonte: Elaborado pelo Autor

Figura 5.21. Efeitos de manter uma pessoa de reserva na equipe

O gráfico que compara a qualidade mostra que manter uma pessoa de reserva contribui para diminuir a quantidade de requisitos especificados incorretamente, o que também ocorre com as simulações apresentadas na seção 5.1.4, que utilizaram dados disponibilizados na literatura para ajustar o modelo.

O gráfico de custo mostra que, diante da ocorrência de *turnover*, a decisão de manter uma pessoa de reserva na equipe diminui o custo quando o prazo é menor que 30 dias. Mas, quando o prazo aumenta para 30 dias ou mais, as duas curvas se cruzam no gráfico, e a decisão de manter uma pessoa de reserva na equipe passa a ter efeito contrário, aumentando o custo. Um comportamento semelhante ocorre com as simulações apresentadas na seção 5.1.4.

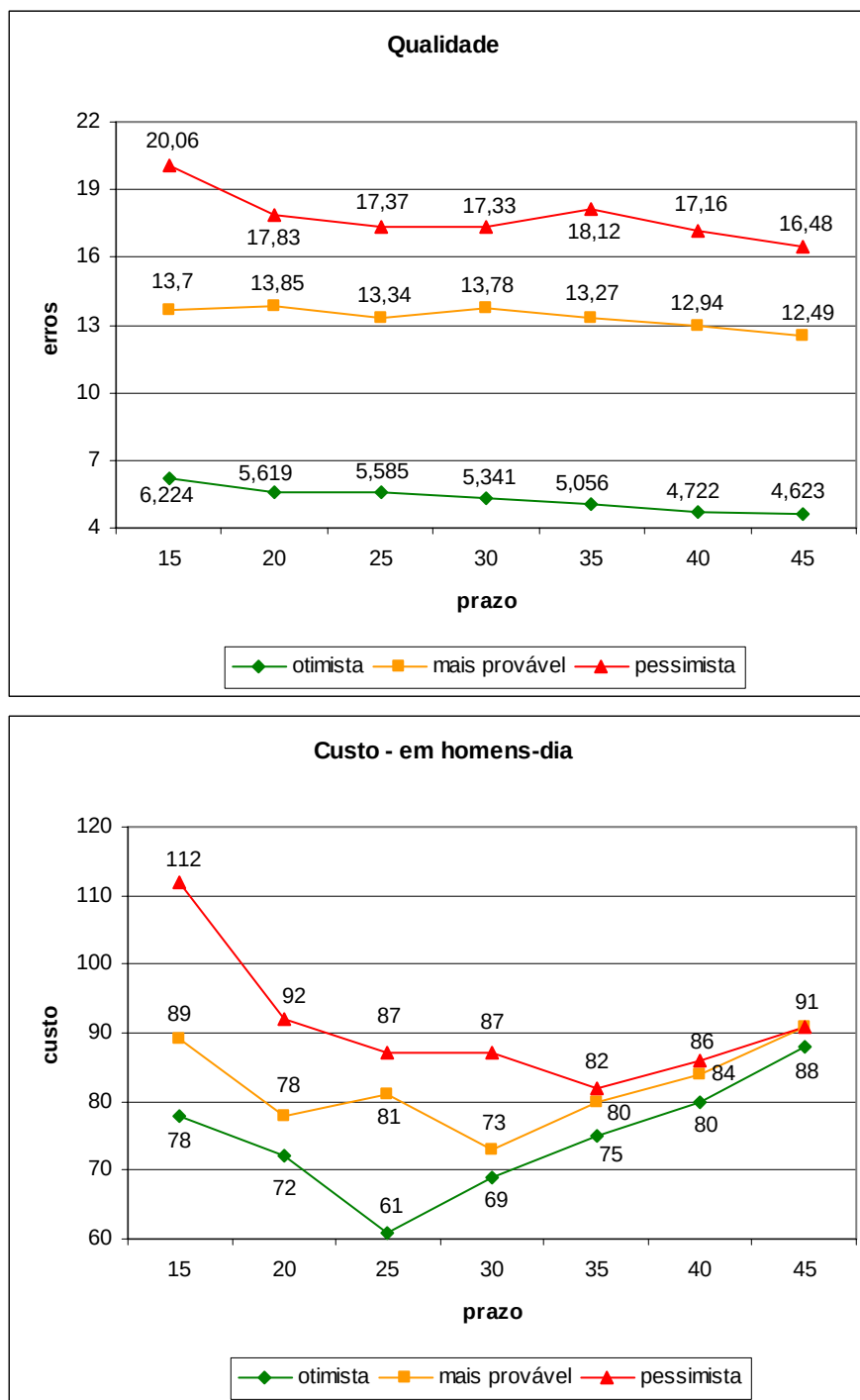
5.2.5 Comparação entre cenários: otimista, mais provável e pessimista

O cenário otimista corresponde a um cenário ideal em que os fatores são favoráveis ao sucesso do projeto. No cenário mais provável, as variáveis que representam os riscos recebem valores médios e mais prováveis. No cenário pessimista é considerada a materialização de riscos, tais como: aumento na quantidade de erros cometidos, alta volatilidade dos requisitos e ocorrência de *turnover*. As configurações para os três cenários, que são baseadas nos valores informados na Tabela 5.3, são:

- **otimista:** probabilidade de alteração de requisitos igual a 10%, probabilidade de cometer erros igual a 5% e sem ocorrência de *turnover*;
- **mais provável:** probabilidade de alteração de requisitos igual a 20%, probabilidade de cometer erros igual a 15% e também sem ocorrência de *turover*;
- **pessimista:** probabilidade de alteração de requisitos igual a 30%, probabilidade de cometer erros igual a 20% e ocorrência de *turnover*. Nesse caso, o *turnover* corresponde a uma pessoa abandonar a equipe quando o tempo decorrido do início da especificação for igual a 40% do prazo planejado. Por exemplo, se o prazo planejado for igual a 25 dias, o *turnover* corresponde a uma pessoa abandonar a equipe no décimo dia de trabalho.

Os gráficos apresentados na Figura 5.22 mostram a qualidade e o custo da especificação de requisitos em função do prazo planejado. Mais uma vez, os formatos das curvas nos gráficos são parecidos, o que é esperado, pois as variáveis que determinam as decisões gerenciais recebem os mesmos valores nos três cenários

simulados. O gráfico que compara a qualidade mostra que os cenários otimista e pessimista apresentam, respectivamente, a menor e a maior quantidade de requisitos especificados incorretamente. Esse comportamento também ocorre com as simulações apresentadas na seção 5.1.5, que utilizaram dados disponibilizados na literatura para ajustar o modelo.



Fonte: Elaborado pelo Autor

Figura 5.22. Comparação entre cenários

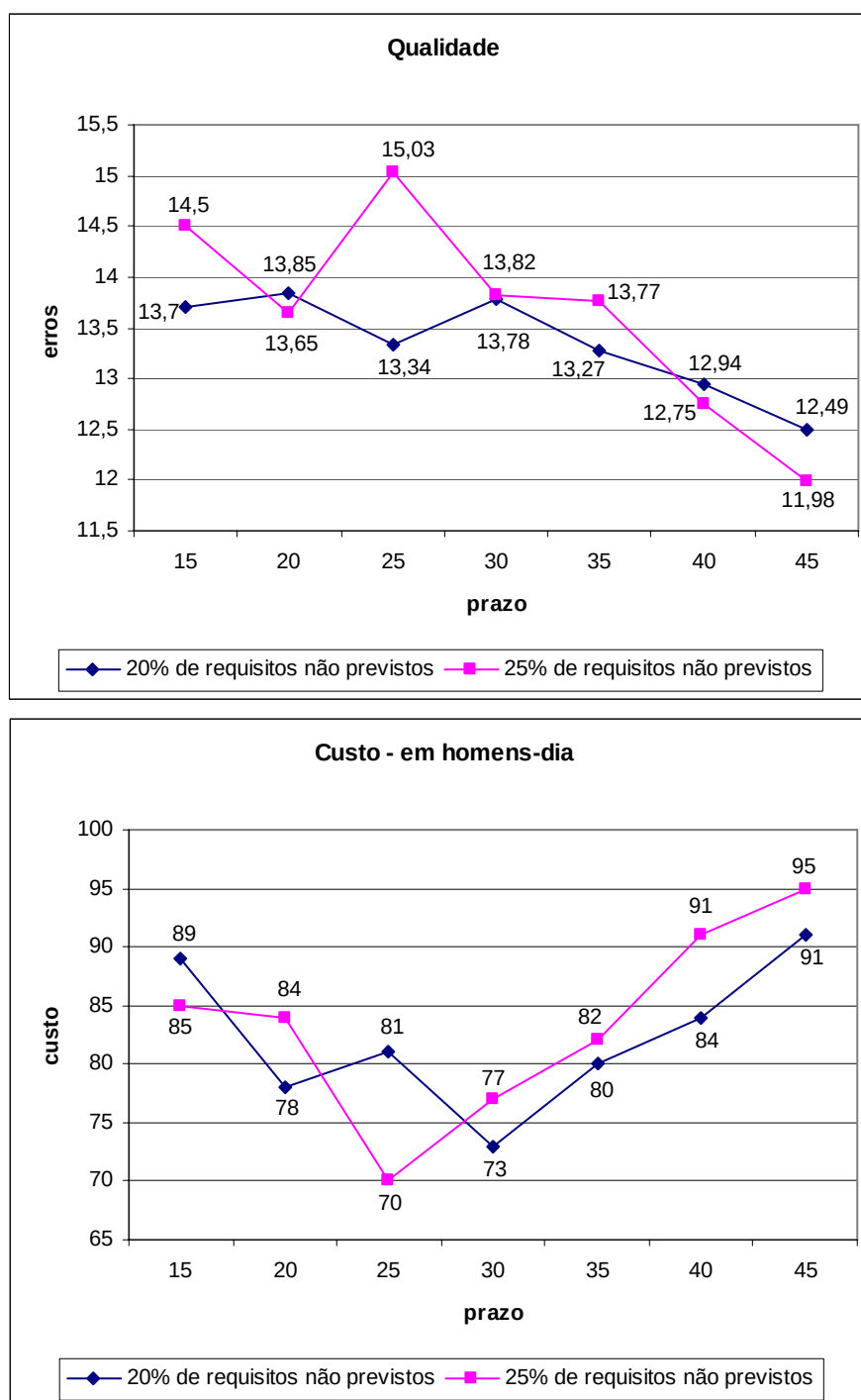
O gráfico de custo mostra que o cenário pessimista apresenta um custo maior que o cenário mais provável, que por sua vez apresenta um custo maior que o cenário otimista. Além disso, quando o prazo para o término da especificação é superestimado, tornando-se maior que o necessário, as curvas no gráfico de custo tendem a se aproximar, ou seja, os custos nos três cenários tendem a se igualar. Esse comportamento também ocorre com as simulações apresentadas na seção 5.1.5.

5.2.6 Erro na estimativa da quantidade de requisitos

Os valores nos gráficos apresentados na Figura 5.23 permitem comparar dois cenários: o cenário base em que 20% dos requisitos não são previstos inicialmente, e outro cenário em que 25% dos requisitos não são previstos inicialmente, que é um valor alto de acordo com os dados informados na Tabela 5.3.

As curvas no gráfico de qualidade se cruzam em diversos pontos, dificultando a realização de uma análise acerca dos efeitos do aumento da quantidade de requisitos não previstos sobre a qualidade da especificação de requisitos.

O gráfico de custo mostra que quando o prazo planejado é maior ou igual a 30 dias, o aumento na quantidade de requisitos não previstos contribui para aumentar o custo. O gráfico também mostra que quando o prazo planejado é menor ou igual a 25 dias, o aumento na quantidade de requisitos não previstos tende a diminuir o custo. Um comportamento semelhante ocorre com as simulações apresentadas na seção 5.1.6, que utilizaram dados disponibilizados na literatura para ajustar o modelo.

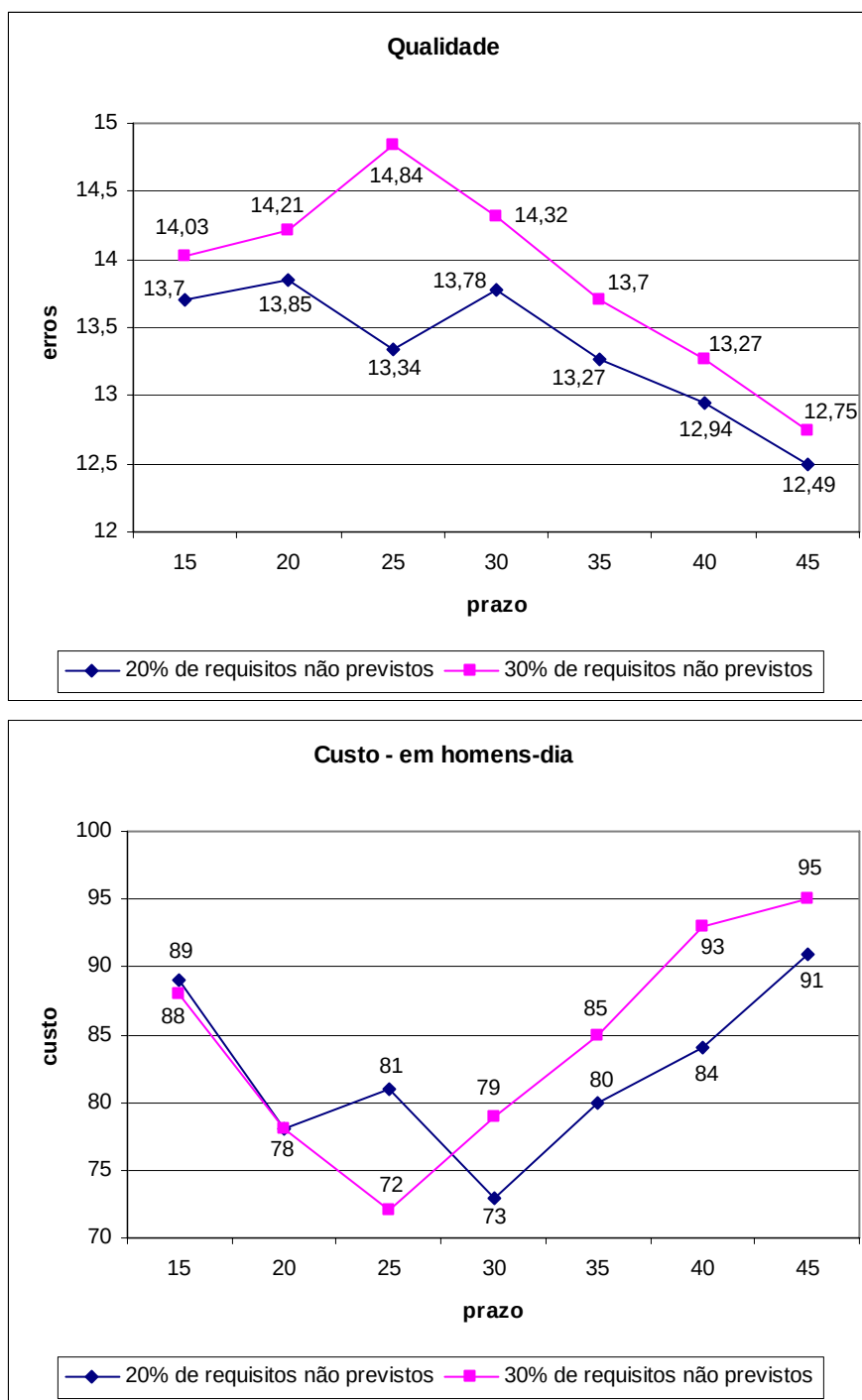


Fonte: Elaborado pelo Autor

Figura 5.23. Impactos do erro na estimativa da quantidade de requisitos

Para verificar se os diversos cruzamentos entre as curvas no gráfico de qualidade foram causados pela proximidade dos valores definidos nos dois cenários (20% e 25% de requisitos não previstos), foi definido um outro cenário em que 30% dos requisitos não são previstos inicialmente. A Figura 5.24 apresenta os gráficos que permitem comparar dois cenários: o cenário base em que 20% dos requisitos não

são previstos inicialmente, e outro cenário em que 30% dos requisitos não são previstos inicialmente.



Fonte: Elaborado pelo Autor

Figura 5.24. Impactos do erro na estimativa da quantidade de requisitos

O gráfico que compara a qualidade mostra que o aumento da quantidade de requisitos não previstos contribui para aumentar a quantidade de requisitos especificados incorretamente. Esse comportamento também ocorre com as

simulações apresentadas na seção 5.1.6, que utilizaram dados disponibilizados na literatura para ajustar o modelo.

O gráfico de custo mostra que quando o prazo planejado é menor ou igual a 25 dias, o aumento na quantidade de requisitos não previstos tende a diminuir o custo. Quando o prazo aumenta, as duas curvas no gráfico de custo se cruzam em algum ponto entre os prazos 25 e 30 dias, e então o aumento na quantidade de requisitos não previstos passa a contribuir para o aumento do custo. Esse comportamento também ocorre com as simulações apresentadas na seção 5.1.6, que utilizaram dados disponibilizados na literatura para ajustar o modelo.

Na simulação do cenário em que 30% dos requisitos não são previstos inicialmente, ocorrem atrasos de até 5 dias para o término da especificação. A Tabela 5.5 mostra o tempo gasto para terminar a especificação em função do prazo planejado.

Tabela 5.5. Tempo gasto para terminar a especificação em função do prazo planejado

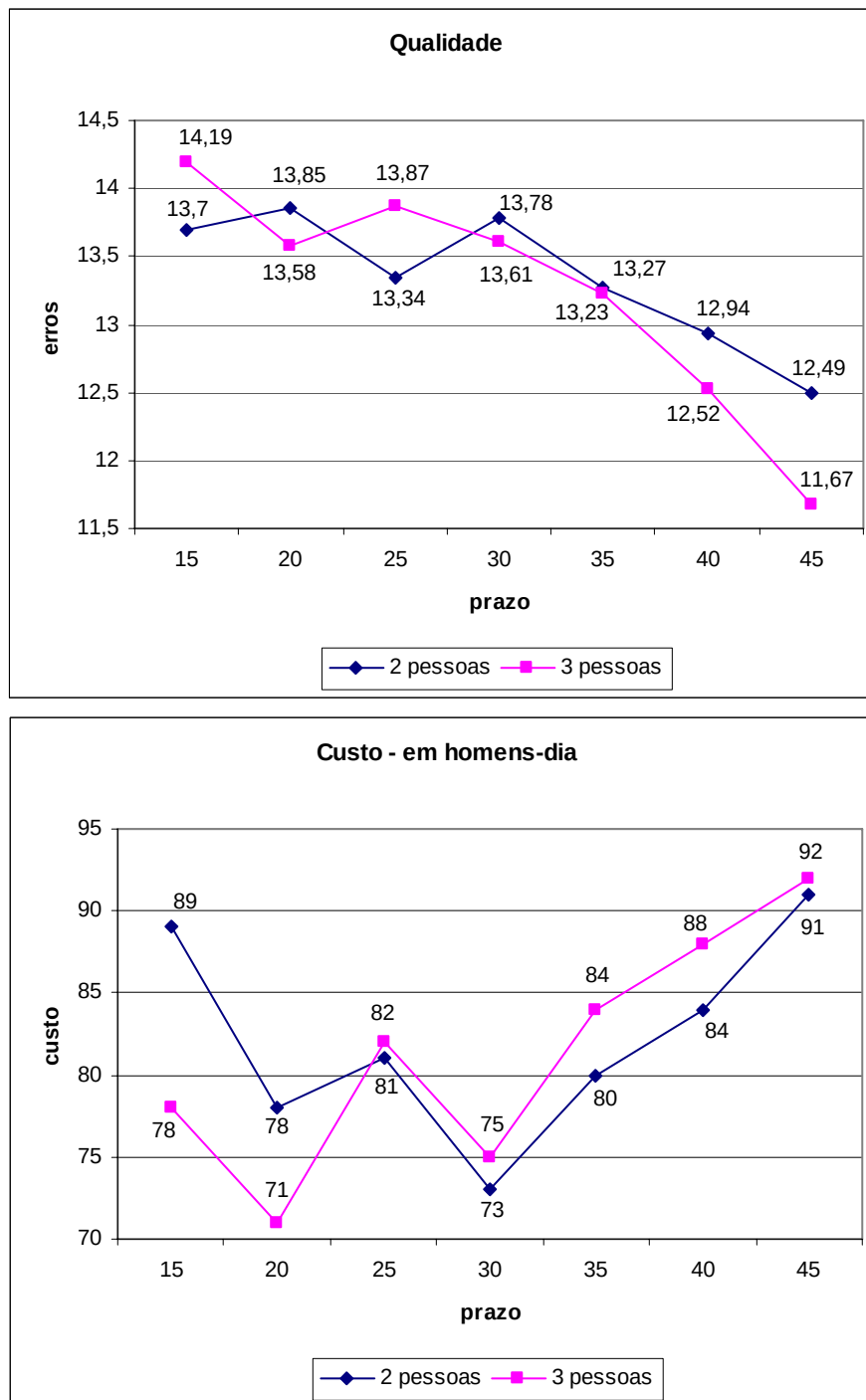
Prazo planejado	15	20	25	30	35	40	45
Tempo gasto no cenário base em que 20% dos requisitos não são previstos	17	22	27	33	37	40	44
Tempo gasto no cenário em que 30% dos requisitos não são previstos	18	24	30	34	38	42	45

Fonte: Elaborado pelo Autor

Para os demais cenários simulados e analisados na seção 5.2, o tempo gasto para terminar a especificação não diverge do tempo gasto no cenário base. Isso ocorre porque, como foi informado, as variáveis do modelo foram configuradas para considerar um gerenciamento que evita alterar o prazo planejado, usando esforço extra dos profissionais e aumentando o tamanho da equipe, quando necessário. Por isso, uma tabela semelhante à Tabela 5.5 não aparece nas demais análises apresentadas na seção 5.2.

5.2.7 Modificações no tamanho da equipe

Os valores apresentados nos gráficos da Figura 5.25 permitem comparar dois cenários: o cenário base em que a especificação é iniciada com 2 pessoas na equipe, e outro cenário em que a especificação é iniciada com 3 pessoas na equipe, que é um valor alto de acordo com os dados informados na Tabela 5.3.



Fonte: Elaborado pelo Autor

Figura 5.25. Impactos de modificações no tamanho da equipe

O gráfico que compara a qualidade mostra que iniciar a especificação com uma equipe maior tende a diminuir a quantidade de requisitos especificados incorretamente. Um comportamento semelhante ocorre com as simulações apresentadas na seção 5.1.7, que utilizaram dados disponibilizados na literatura para ajustar o modelo. O aumento no tamanho da equipe diminui a necessidade de utilizar

esforço extra, o que diminui a taxa com que os erros são cometidos (COLLOFELLO *et al.*, 1998; LIN; ABDEL-HAMID; SHERIF, 1997).

No entanto, quando o prazo planejado é 15 ou 25 dias, iniciar a especificação com 3 pessoas na equipe contribui para aumentar a quantidade de requisitos especificados incorretamente, provocando cruzamentos entre as curvas no gráfico de qualidade. Isso ocorre devido às interações dinâmicas entre as variáveis e aos vários laços de realimentação (*feedback*) existentes no modelo. Nesse caso, quando o prazo planejado é 15 ou 25 dias e a especificação inicia com 3 pessoas na equipe, à medida que os requisitos não previstos inicialmente são descobertos próximo ao término da especificação, é utilizada uma quantidade maior de esforço extra, evitando o aumento do tamanho da equipe. O aumento no uso de esforço extra contribui para aumentar a taxa com que os erros são cometidos (COLLOFELLO *et al.*, 1998; LIN; ABDEL-HAMID; SHERIF, 1997).

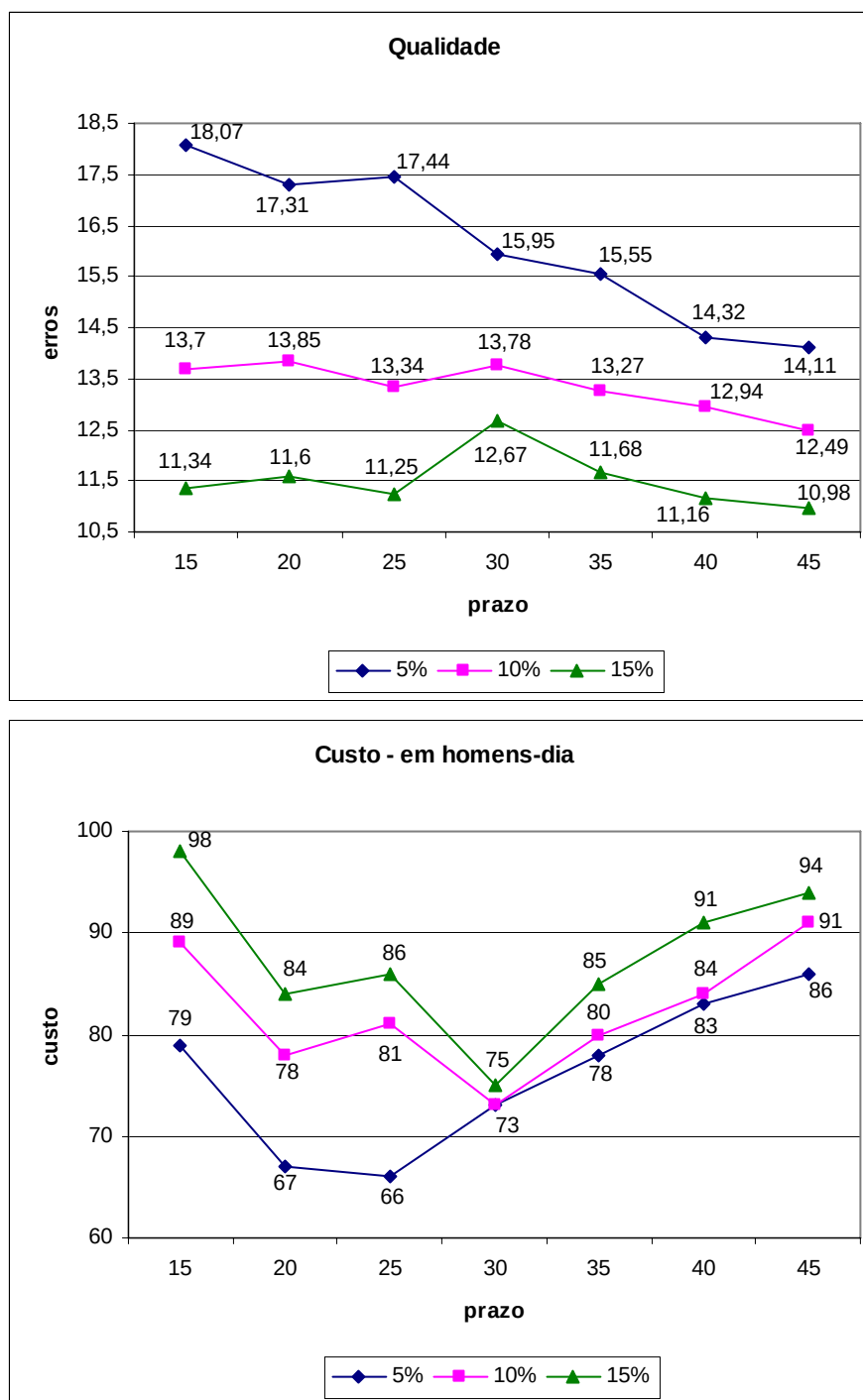
Os pontos de cruzamento no gráfico de custo devem ser analisados pelos tomadores de decisão, pois neles ocorre uma inversão nos efeitos das decisões gerenciais. Para um prazo menor que 25 dias, o gráfico de custo mostra que iniciar a especificação com 3 pessoas na equipe contribui para que o custo seja menor. Mas, quando o prazo aumenta para 25 dias ou mais, iniciar a especificação com 3 pessoas na equipe provoca o aumento do custo. Um comportamento semelhante ocorre com as simulações apresentadas na seção 5.1.7.

5.2.8 Alocação de esforço para as atividades de garantia de qualidade

Para verificar os impactos da alocação de diferentes quantidades de esforço para as atividades de garantia de qualidade foram simulados três cenários: o cenário base em que a porcentagem desejada de homens-dia alocados para garantia de qualidade é 10%, um cenário em que essa porcentagem é 5%, e outro cenário em que essa porcentagem é 15%. Os valores são baseados nos dados informados na Tabela 5.3.

Os gráficos apresentados na Figura 5.26 mostram a qualidade e o custo da especificação de requisitos em função do prazo planejado. O gráfico que compara a qualidade mostra que o aumento na quantidade de esforço alocado para as atividades de garantia de qualidade provoca uma redução na quantidade de requisitos especificados incorretamente. Esse comportamento também ocorre com as

simulações apresentadas na seção 5.1.8, que utilizaram dados disponibilizados na literatura para ajustar o modelo.



Fonte: Elaborado pelo Autor

Figura 5.26. Variação na quantidade de esforço alocado para as atividades de garantia de qualidade

Os formatos das curvas no gráfico de custo são parecidos, o que é esperado, pois as variáveis que determinam as demais decisões gerenciais recebem os mesmos valores nos três cenários simulados. O gráfico de custo mostra que aumentar a

quantidade de esforço alocado para as atividades de garantia de qualidade provoca o aumento do custo. Além disso, quando o prazo para o término da especificação aumenta, tornando-se maior que o necessário, as curvas no gráfico de custo tendem a se aproximar, ou seja, os custos nos três cenários tendem a se igualar. Esse comportamento também ocorre com as simulações apresentadas na seção 5.1.8.

5.2.9 Uso de esforço extra

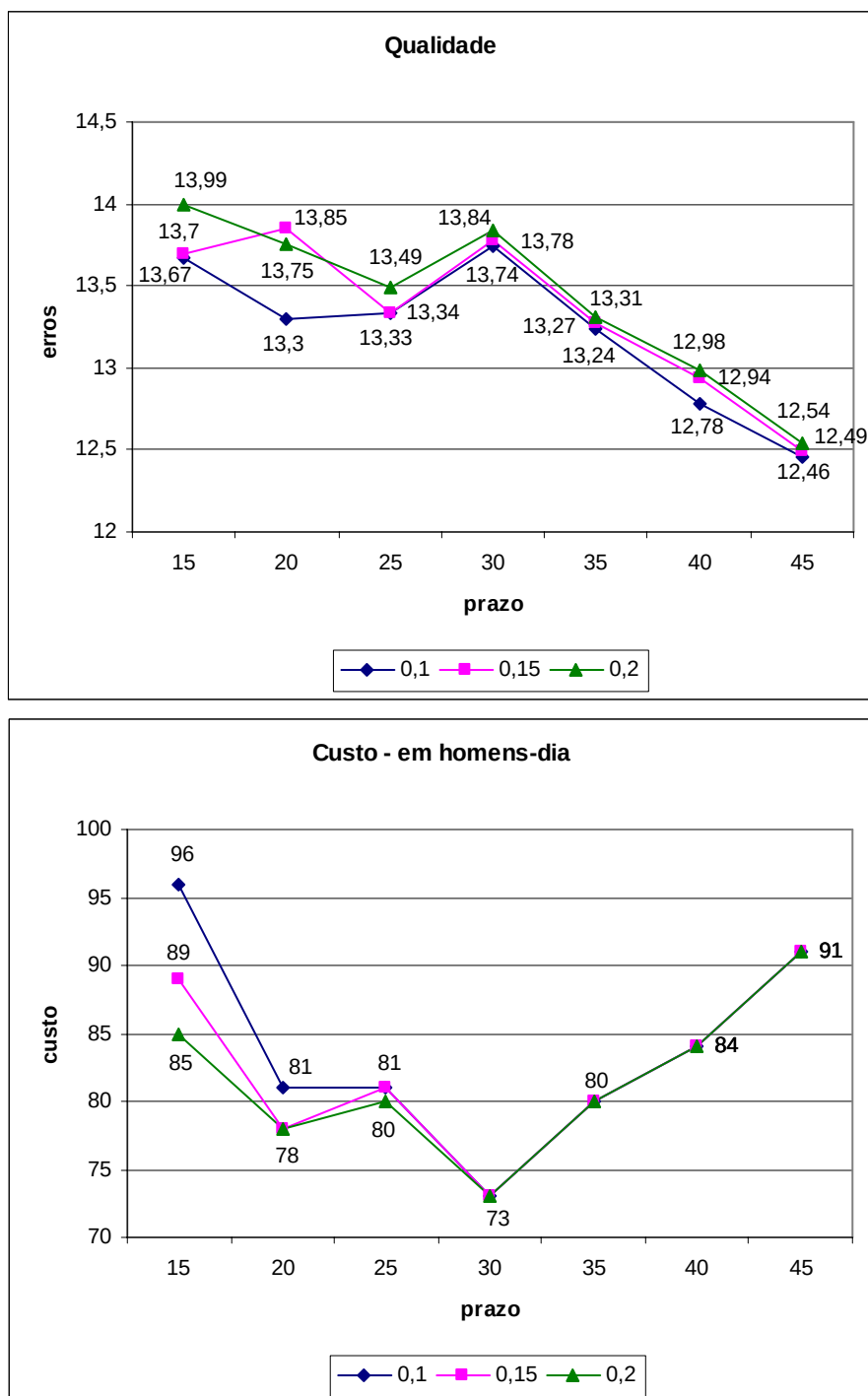
Visando a verificar os impactos das diferentes políticas gerenciais acerca da quantidade de esforço extra utilizado quando há risco de atraso no projeto, foram simulados três cenários: o cenário base em que a quantidade máxima de homens-dia extras disponibilizados por um membro experiente é 0,15 homens-dia (ou seja, os membros da equipe são incentivados a disponibilizar até 15% a mais de esforço), um cenário em que esse valor é 0,10 homens-dia (até 10% a mais de esforço), e outro cenário em que esse valor é 0,20 homens-dia (até 20% a mais de esforço). Os valores são baseados nos dados informados na Tabela 5.3.

Os gráficos apresentados na Figura 5.27 mostram a qualidade e o custo da especificação de requisitos em função do prazo planejado. As curvas no gráfico que compara a qualidade possuem algumas oscilações. No entanto, é possível observar que o aumento na quantidade de esforço extra utilizado tende a provocar o aumento da quantidade de requisitos especificados incorretamente. Esse comportamento também ocorre com as simulações apresentadas na seção 5.1.9, que utilizaram dados disponibilizados na literatura para ajustar o modelo.

O gráfico de custo mostra que o aumento no uso do esforço extra contribui para reduzir o custo. É necessário ressaltar que isso ocorre porque nos três cenários simulados, a quantidade de esforço extra utilizado é pequena, se for classificada de acordo os intervalos de valores definidos na Tabela 4.1. A empresa consultada evita utilizar horas-extras dos funcionários, e todo o esforço extra disponibilizado é obtido a partir de um maior empenho dos profissionais durante o horário normal de trabalho. Para os três cenários simulados, a quantidade de esforço extra utilizado não foi suficiente para que o nível de exaustão da equipe atingisse o limite máximo suportado.

O uso de uma maior quantidade de esforço extra (até um limite que não provoca a exaustão da equipe) pode evitar o aumento no tamanho da equipe, contribuindo para reduzir o custo. Durante o trabalho, os profissionais utilizam parte

de seu esforço em tarefas pessoais, como ver e-mails e conversar com os colegas (ABDEL-HAMID; MADNICK, 1991). Quando há pressões de prazo e não é utilizado esforço extra dos profissionais, eles continuam desperdiçando parte de seu esforço em suas atividades pessoais, aumentando o custo.



Fonte: Elaborado pelo Autor

Figura 5.27. Variação na quantidade de esforço extra utilizado

As curvas no gráfico de custo possuem formatos semelhantes e quando o prazo aumenta para 30 dias ou mais, as curvas se sobrepõem. Isso ocorre porque quando o prazo aumenta nos três cenários simulados, é necessário utilizar apenas uma pequena quantidade de esforço extra (no máximo 0,1 homens-dia) para evitar o aumento do tamanho da equipe. Então, o custo passa a depender somente do tamanho da equipe, que é o mesmo nos três cenários simulados.

5.3 Considerações sobre as simulações

A semelhança entre o comportamento do modelo nas simulações apresentadas na seção 5.1 (que utilizaram dados disponibilizados na literatura para ajustar o modelo) e o comportamento obtido nas simulações apresentadas na seção 5.2 (que utilizaram dados coletados em uma empresa de desenvolvimento de software), constitui uma evidência de que o comportamento do modelo é mais geral e menos dependente das características de empresas e de projetos, em particular.

O comportamento do modelo e as tendências e formatos das curvas nos gráficos apresentados podem ser extrapolados e generalizados obedecendo-se a determinados limites e restrições. Esses limites e restrições não são investigados profundamente nesse trabalho. No entanto, é esperado que a generalização possa ser feita para projetos que estejam inseridos em um contexto (formado pelas características da organização, da equipe e do projeto) semelhante ao contexto dos projetos que serviram como base para que outros grupos de pesquisa (referenciados ao longo do texto) disponibilizassem as informações e os dados utilizados para definir e quantificar os relacionamentos existentes no modelo.

O mais importante a ser analisado nos gráficos não são os valores numéricos apresentados, mas sim a direção das mudanças nos valores ao comparar os gráficos dos cenários simulados. Modelos que não são bons para prever valores numéricos específicos podem ser válidos e úteis se eles mostram a direção e a magnitude das mudanças dos valores (MCCARL, 1984). É importante que o modelo forneça uma compreensão do comportamento do sistema e não uma informação numérica de alta precisão.

A discussão apresentada deixa evidentes as vantagens de se utilizar o modelo e as simulações para apoiar o gerenciamento das atividades realizadas durante a fase de requisitos. Diante da impossibilidade de se reproduzir um projeto de software para se estudar as consequências das modificações em alguns fatores e variáveis que o

afetam, os modelos surgem como uma alternativa viável para a criação dos “laboratórios de aprendizagem” (STERMAN, 1992) nas organizações, permitindo aos gerentes testar e verificar os efeitos de diferentes suposições que podem impactar no processo. Os modelos possibilitam aos gerentes aprender com as simulações, sem incorrer nos riscos e prejuízos que podem ocorrer em um projeto real.

6 CONCLUSÕES

A partir dos estudos e pesquisas realizadas, foi construído um modelo de dinâmica de sistemas para a fase de requisitos em processos de desenvolvimento de software. A discussão acerca dos relacionamentos e das interações entre as variáveis envolvidas, explicitados no modelo, contribuem para aumentar a compreensão acerca dos aspectos gerenciais que afetam o trabalho realizado durante a fase de requisitos. Com isso, o objetivo geral, proposto nessa pesquisa, foi atingido.

Os objetivos específicos também foram alcançados. A estrutura e a descrição do modelo, apresentadas no Capítulo 3, abordam relações entre variáveis que muitas vezes não são consideradas pelos gerentes. Essas relações, que foram apresentadas e discutidas, explicam o comportamento dinâmico dos requisitos e de algumas variáveis envolvidas na fase de requisitos.

Para usar o modelo como um veículo de experimentação, foi definido um painel de controle contendo algumas variáveis que permitem ajustar o modelo ao realizar as simulações. As variáveis de configuração permitem configurar o modelo de acordo com as características da organização, da equipe e do projeto que definem o contexto da simulação. E as variáveis de análise permitem estabelecer os riscos e as decisões gerenciais que serão analisadas durante a simulação.

O modelo permite realizar experimentos controlados, acerca da fase de requisitos em projetos de software, que são inviáveis de serem feitos em ambientes reais devido ao custo e ao tempo necessários. As simulações possibilitam aos gerentes verificar os impactos da materialização de riscos e as consequências dinâmicas das decisões gerenciais planejadas, antes que elas sejam implementadas em um projeto real.

A capacidade de configurar o modelo, por meio de um painel de controle, torna possível a sua utilização como uma ferramenta para apoiar os gerentes na tomada de decisão no decorrer da fase de requisitos. O modelo permite estudar e antever as interações entre as variáveis mediante diversos cenários, e auxilia na descoberta da ação gerencial mais adequada para lidar com um determinado problema.

Nos diversos cenários simulados, o padrão de variação e o valor de algumas variáveis (como esforço utilizado, quantidade de erros, prazo planejado e tamanho da equipe) mostraram-se de acordo com o conhecimento comum da área de Engenharia de Software. Isso mostra a capacidade do modelo em reproduzir o comportamento dinâmico gerado pelas interações entre as variáveis envolvidas na fase de requisitos.

As análises das simulações apresentadas mostraram que o comportamento do modelo ao ajustá-lo utilizando-se dados coletados em uma empresa de desenvolvimento de software não diverge do comportamento obtido quando são utilizados dados disponibilizados na literatura. Esse fato evidencia a robustez do modelo e a possibilidade de generalizar os resultados obtidos e as análises apresentadas.

As análises gerenciais suportadas pelo modelo consistem em verificar:

- os impactos da materialização de riscos como uma alta taxa de alteração de requisitos, uma alta taxa de *turnover* e o aumento na quantidade de erros cometidos;
- os impactos de cometer erros ao estimar a quantidade de requisitos da liberação;
- os efeitos do planejamento de um determinado prazo para o término da especificação e da alocação de uma determinada quantidade de pessoas na equipe para iniciar a especificação;
- os efeitos da alocação de uma determinada porcentagem dos recursos para as atividades de garantia de qualidade e do uso de uma determinada quantidade de esforço extra para evitar alterações no prazo planejado;
- os efeitos de uma maior ou menor tendência em: alterar o prazo planejado, modificar o tamanho da equipe e utilizar esforço extra da equipe.

Os relacionamentos entre as variáveis do modelo foram definidos e quantificados a partir de informações publicadas na literatura. Isso torna a estrutura do modelo embasada no conhecimento disponibilizado pela comunidade científica, garantindo uma maior confiabilidade aos resultados obtidos durante as simulações.

O modelo foi construído por meio de diversos refinamentos. Cada versão obtida do modelo proporcionava um maior entendimento sobre os relacionamentos entre as variáveis, provocando modificações no modelo e originando uma nova versão. Ao utilizar a dinâmica de sistemas para modelar um problema ou sistema, o próprio processo de construção do modelo gera um processo de aprendizagem que

gera uma nova construção e assim sucessivamente, constituindo um ciclo de reforço positivo.

Por exemplo, a simulação da primeira versão do modelo chegou ao fim informando que a quantidade de requisitos especificados havia sido maior que a quantidade de requisitos da liberação corrente. Esses valores são inconsistentes porque a quantidade de requisitos especificados deve ser igual à quantidade de requisitos da liberação. Essa inconsistência ocorreu devido a um relacionamento incorreto entre elementos do modelo que foi identificado e corrigido.

6.1 Trabalhos futuros

O modelo construído não abrange todos os fatores envolvidos na fase de requisitos. Uma possível extensão para esse trabalho é a realização de novos refinamentos no modelo com o objetivo de incluir novas variáveis e relacionamentos, oferecendo suporte a novos cenários. Também podem ser construídos novos modelos que permitam analisar outros aspectos da fase de requisitos. Por exemplo, podem ser identificadas a influência da prototipação e as consequências da utilização de processos controlados. Podem ser explorados também os fatores subjetivos que afetam o trabalho realizado durante a fase de requisitos como a capacidade e a experiência dos profissionais da equipe, conhecimento do domínio do problema e outros.

A partir do modelo, podem ser definidos indicadores de decisão para realizar avaliação de riscos durante a fase de requisitos, como, por exemplo, analisar o impacto de uma alta taxa de alteração de requisitos. Esses indicadores podem ser embutidos em uma ferramenta de apoio à tomada de decisão para permitir a visualização dos indicadores de decisão, representando o comportamento do modelo ao longo do tempo.

O modelo também pode ser utilizado como base para a implementação de um simulador gerencial para treinar gerentes no processo de tomada de decisão durante a fase de requisitos em projetos de software. Inicialmente, o gerente deveria configurar o simulador de acordo com as características do projeto que será simulado. O simulador apresentaria as informações sobre o projeto e, por meio de um painel de controle, solicitaria que o gerente informasse as decisões gerenciais que devem ser empregadas: alterações no tamanho da equipe, modificações no prazo, quantidade de esforço extra que deve ser usado, etc. Em seguida, seria simulado um intervalo de

tempo do projeto e as informações sobre o projeto seriam novamente apresentadas para que o gerente pudesse tomar as decisões para o próximo intervalo de tempo. Esse processo continuaria até o término da simulação do projeto, o que pode ocorrer quando todos os requisitos forem especificados ou quando um prazo máximo for alcançado.

O simulador gerencial poderia permitir que os gerentes verificassem o seu desempenho por meio de uma análise da quantidade de esforço utilizado, quantidade de tempo gasto e quantidade de erros cometidos no decorrer da especificação dos requisitos. Poderiam ser comparados também os desempenhos obtidos antes e após os gerentes serem apresentados ao modelo e entenderem as variáveis e os relacionamentos existentes.

Outra sugestão para trabalhos futuros consiste em implementar e utilizar redes neurais para ajustar os valores iniciais de algumas variáveis e quantificar alguns relacionamentos presentes no modelo. As redes neurais também poderiam ser utilizadas para alertar os gerentes quando ocorrerem situações críticas, que exigem a tomada de decisões para evitar o fracasso do projeto.

Visando a promover o reuso e a facilitar a instanciação de modelos de dinâmica de sistemas para o gerenciamento de projetos, podem ser construídos modelos genéricos que descrevem a estrutura principal do processo. Esses modelos poderiam então, ser especializados e detalhados para representar as particularidades do problema ou da situação que se deseja analisar.

Outra atividade que poderia ampliar as contribuições desse trabalho consiste em identificar os limites e as restrições que devem ser obedecidas para que se possa generalizar o comportamento do modelo e as análises apresentadas no Capítulo 5.

APÊNDICE A

A.1 Glossário com a definição dos elementos do Modelo Nível 1

ajuste do prazo: conversor – representa o ajuste que deve ser feito no prazo planejado. O seu valor é dado pela diferença entre os valores da variável *tempo necessário para terminar* e do estoque *Prazo Planejado para o Término da Especificação*.

aprendizado: conversor – representa o aprendizado que ocorre sobre o sistema e o domínio do problema à medida que as pessoas trabalham em um projeto. Esse aprendizado aumenta a produtividade da equipe.

densidade de erros: conversor – representa a densidade de erros que corresponde à relação entre a quantidade de erros não descobertos e a quantidade de requisitos especificados.

densidade de erros nos requisitos entregues: conversor – representa a densidade de erros nos requisitos entregues que corresponde à relação entre a quantidade de erros não descobertos e a quantidade de requisitos entregues.

Exaustão: estoque – representa o nível de exaustão da equipe devido à sobrecarga de trabalho.

Falhas Descobertas pelos Usuários: estoque – representa todos os requisitos, entregues em liberações anteriores, que apresentam erros descobertos pelos usuários.

Falhas Detectadas: estoque – representa todos os requisitos especificados que apresentam erros já detectados pelas atividades de garantia de qualidade.

falta de homens-dia percebida: conversor – representa a quantidade de esforço, medido em homens-dia, que irá faltar para a realização das atividades, impedindo que a especificação termine no prazo planejado.

força de trabalho alcançável: conversor – representa a quantidade de pessoas que podem constituir a equipe no momento, obedecendo às restrições impostas.

Força de Trabalho Experientes: estoque – representa os membros experientes que compõem a equipe.

força de trabalho necessária: conversor – representa a quantidade de pessoas que devem compor a equipe para que a especificação termine no prazo planejado.

Força de Trabalho Novatos: estoque – representa os membros novatos que compõem a equipe.

força de trabalho total: conversor – representa a quantidade total de pessoas que compõem a equipe.

homens-dia alocados para alteração de requisitos: representa a quantidade de esforço, medido em homens-dia, alocado para realizar alterações de requisitos.

homens-dia alocados para correção: conversor – representa a quantidade de esforço, medido em homens-dia, alocado para corrigir as falhas detectadas.

homens-dia alocados para especificação de requisitos: conversor – representa a quantidade de esforço, medido em homens-dia, alocado para especificar novos requisitos.

homens-dia alocados para garantia de qualidade: conversor – representa a quantidade de esforço, medido em homens-dia, alocado para as atividades de garantia de qualidade.

homens-dia disponibilizados por um membro experiente: conversor – representa a quantidade de esforço, medido em homens-dia, que um membro experiente da equipe dedica à especificação em um dia de trabalho.

homens-dia disponibilizados por um membro novato: conversor – representa a quantidade de esforço, medido em homens-dia, que um membro novato da equipe dedica à especificação em um dia de trabalho.

homens-dia extras disponibilizados por um membro experiente: conversor – representa a quantidade de esforço, medido em homens-dia, obtido com o esforço extra de um membro experiente da equipe.

homens-dia extras obtidos por meio de horas-extras: conversor – representa a quantidade de esforço extra, medido em homens-dia, que um membro experiente da equipe dedica à especificação, por meio do aumento na jornada de trabalho.

Homens-dia Gastos: estoque – representa a quantidade de esforço, medido em homens-dia, utilizado até o momento.

perdas devido à comunicação: conversor – representa a queda na produtividade que ocorre devido ao tempo gasto com comunicação e reuniões envolvendo os membros da equipe, o que diminui a quantidade de homens-dia utilizados nas atividades de especificação.

potencial de detecção de erros: conversor – representa o potencial de detecção de erros da equipe.

Prazo Planejado para o Término da Especificação: estoque – representa o prazo em dias estimado para o término da especificação dos requisitos.

pressões de prazo: conversor – representa as pressões de prazo sobre a equipe quando há risco de atraso no projeto.

probabilidade de cometer erros: conversor – representa a probabilidade com que os erros são cometidos durante a realização das atividades na fase de requisitos.

produtividade percebida: conversor – representa a produtividade percebida da equipe no decorrer da especificação dos requisitos.

quantidade de homens-dia necessários para terminar a especificação: conversor – representa a quantidade de esforço, medido em homens-dia, necessário para terminar a especificação dos requisitos.

relação produtividade de um novato / produtividade de um experiente: conversor – representa a relação entre a produtividade de um membro novato da equipe e a produtividade de um membro experiente.

Requisitos Aguardando Alteração: estoque – representa todos os requisitos especificados que necessitam ser alterados para atender às novas necessidades do cliente.

Requisitos com Erros: estoque – representa todos os requisitos especificados que apresentam erros ainda não detectados pelas atividades de garantia de qualidade.

Requisitos da Liberação: estoque – representa todos os requisitos que devem ser especificados e entregues na próxima liberação.

Requisitos Entregues Aguardando Alteração: estoque – representa todos os requisitos, entregues em liberações anteriores, que necessitam ser alterados para atender às solicitações do cliente.

Requisitos Entregues com Erros: estoque – representa todos os requisitos que foram especificados incorretamente e entregues em liberações anteriores.

Requisitos Entregues Especificados: estoque – representa todos os requisitos que foram corretamente especificados e entregues em liberações anteriores.

Requisitos Entregues Re-especificados: estoque – representa todos os requisitos entregues que foram re-especificados. A re-especificação é realizada quando é necessário corrigir ou alterar algum requisito especificado anteriormente.

Requisitos Especificados: estoque – representa todos os requisitos que foram corretamente especificados.

Requisitos não Previstos: estoque – representa os requisitos, não identificados inicialmente, que poderão ser descobertos após o início da especificação.

Requisitos Re-especificados: estoque – representa todos os requisitos que foram re-especificados. A re-especificação é realizada quando é necessário corrigir ou alterar algum requisito especificado anteriormente.

restrições de alocação: conversor – representa as restrições para a alocação de pessoas na equipe. Essas restrições correspondem ao tempo necessário para a gerência remanejar profissionais de outras equipes ou contratar profissionais disponíveis no mercado. Além disso, a quantidade máxima de novatos que podem existir na equipe pode ser limitada pela quantidade de membros experientes e pela quantidade de novatos que cada membro experiente pode auxiliar por meio de *mentoring* ou acompanhamento.

saldo da força de trabalho: conversor – representa o saldo da força de trabalho. O seu valor é determinado pela diferença entre os valores das variáveis *força de trabalho total* e *força de trabalho alcançável*.

tamanho do trabalho em pontos de função: conversor – representa a quantidade de trabalho que deve ser realizado. Corresponde a todos os requisitos que devem ser especificados.

Taxa de Alocação: fluxo – representa a alocação de novas pessoas na equipe.

Taxa de Alteração de Requisitos: fluxo – representa a taxa com que os requisitos são alterados conforme as solicitações do cliente.

Taxa de Alteração de Requisitos Entregues: fluxo – representa a taxa com que os requisitos entregues em liberações anteriores são alterados pela equipe.

Taxa de Aprendizado: fluxo – representa o aprendizado que ocorre durante o trabalho.

Taxa de Aumento do Prazo: fluxo – representa o aumento no prazo planejado.

Taxa de Correção: fluxo – representa a taxa com que as falhas detectadas pelas atividades de garantia de qualidade são corrigidas pela equipe.

Taxa de Correção dos Erros Descobertos: fluxo – representa a taxa com que os erros descobertos e reportados pelos usuários, são corrigidos pela equipe.

Taxa de Descanso: fluxo – representa o descanso da equipe, o que diminui o nível de exaustão da equipe.

Taxa de Descoberta de Requisitos: fluxo – representa a taxa com que os requisitos, não previstos inicialmente, são descobertos e incluídos entre os requisitos que serão entregues na próxima liberação.

Taxa de Descoberta de Erros: fluxo – representa a taxa com que os erros de requisitos são descobertos pelos usuários ao utilizarem o software.

Taxa de Detecção de Erros: fluxo – representa a taxa com que os erros cometidos durante a especificação são detectados pelas atividades de garantia de qualidade, como revisões e inspeções.

Taxa de Erros: fluxo – representa a taxa com que os erros são cometidos durante a especificação de novos requisitos.

Taxa de Erros Durante a Correção e Alteração: fluxo – representa a taxa com que os erros são cometidos durante a correção e/ou alteração de requisitos.

Taxa de Especificação Correta: fluxo – representa a taxa com que os requisitos solicitados pelo cliente são especificados corretamente.

Taxa de Exaustão: fluxo – representa a taxa com que o nível de exaustão da equipe aumenta devido à sobrecarga de trabalho.

Taxa de Gasto de Homens-dia: fluxo – representa a quantidade de esforço utilizado por dia para a realização das atividades durante a fase de requisitos.

Taxa de Redução do Prazo: fluxo – representa a redução no prazo planejado.

Taxa de Remoção de Experientes: fluxo – representa a remoção de membros experientes da equipe.

Taxa de Remoção de Novatos: fluxo – representa a remoção de membros novatos da equipe.

Taxa de Solicitação de Alteração: fluxo – representa a taxa com que as alterações de requisitos são solicitadas pelo cliente.

Taxa de Solicitação de Alteração de Requisitos Entregues: fluxo – representa a taxa com que são solicitadas as alterações nos requisitos entregues em liberações anteriores.

Taxa de Turnover: fluxo – representa o *turnover* de profissionais alocados na equipe.

tempo necessário para terminar: conversor – representa a quantidade de dias necessários para o término da especificação.

total de homens-dia disponíveis diariamente: conversor – representa a quantidade de esforço, medido em homens-dia, disponibilizado diariamente pela equipe para a realização das atividades durante a fase de requisitos.

trabalho realizado em pontos de função: conversor – representa a quantidade de trabalho que já foi realizado. Corresponde a todos os requisitos que já foram especificados.

trabalho restante em pontos de função: conversor – representa a quantidade de trabalho necessário para especificar os requisitos que ainda não foram especificados.

REFERÊNCIAS BIBLIOGRÁFICAS

ABDEL-HAMID, T. K.; MADNICK, S. E. Lessons Learned from Modeling the Dynamics of Software Development. **Communication of the ACM**, v. 32, n. 12, p. 1426-1438, 1989.

ABDEL-HAMID, T. K. Investigating the cost/schedule trade-off in software development. **IEEE Software**, v. 7, n. 1, p. 97-105, 1990.

ABDEL-HAMID, T. K.; MADNICK, S. E. **Software Project Dynamics: an Integrated Approach**. Englewood Cliffs: Prentice Hall, 1991. 264 p.

ABDEL-HAMID, T. K. Adapting, correcting, and perfecting software estimates: a maintenance metaphor. **IEEE Computer**, v. 26, n. 3, p. 20-29, 1993.

ABDEL-HAMID, T. K. The slippery path to productivity improvement. **IEEE Software**, v. 13, n. 4, p. 43-52, 1996.

ABDEL-HAMID, T. K.; SENGUPTA, K.; SWETT, C. The Impact of Goals on Software Project Management: An Experimental Investigation. **MIS Quarterly**, v. 23, n. 4, p. 531-555, 1999.

ANDERSON, S.; FELICI, M. Quantitative Aspects of Requirements Evolution. In: COMPUTER SOFTWARE AND APPLICATIONS CONFERENCE, 26., 2002, Oxford. **Proceedings...** Oxford: IEEE Computer Society, 2002. p. 27-32.

ANG, S.; SLAUGHTER, S. Turnover of information technology professionals: the effects of internal labor market strategies. **ACM SIGMIS Database**, v. 35, n. 3, p. 11-27, 2004.

BARROS, M. O.; WERNER, C. M. L.; TRAVASSOS, G. H. Risk Analysis: a Key Success Factor for Complex System Development. In: INTERNATIONAL CONFERENCE ON SOFTWARE AND SYSTEMS ENGINEERING AND THEIR APPLICATIONS, 12., 1999, Paris. **Proceedings...** Paris: CNAM, 1999.

BARROS, M. O.; WERNER, C. M. L.; TRAVASSOS, G. H. Explaining the Behavior of System Dynamics Model. In: INTERNATIONAL CONFERENCE

OF THE SYSTEM DYNAMICS SOCIETY, 19., 2001a, Atlanta. **Abstract...**
Albany: System Dynamics Society, 2001. p. 34.

BARROS, M. O.; WERNER, C. M. L.; TRAVASSOS, G. H. From Metamodels to Models: Organizing and Reusing Domain Knowledge in System Dynamics Model Development. In: INTERNATIONAL CONFERENCE OF THE SYSTEM DYNAMICS SOCIETY, 19., 2001b, Atlanta. **Abstract...** Albany: System Dynamics Society, 2001. p. 34.

BARROS, M. O.; WERNER, C. M. L.; TRAVASSOS, G. H. Scenario Oriented Project Management Knowledge Reuse within a Risk Analysis Process. In: INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING AND KNOWLEDGE ENGINEERING, 13., 2001c, Buenos Aires. **Proceedings...** Buenos Aires, 2001. p. 37-44.

BARROS, M. O.; WERNER, C. M. L.; TRAVASSOS, G. H. System Dynamics Extension Modules for Software Process Modeling. In: PROSIM - INTERNATIONAL WORKSHOP ON SOFTWARE PROCESS SIMULATION AND MODELING, 4., 2003, Oregon.

BASIL, V. R.; BRIAND, L. C.; MELO, W. L. How reuse influences productivity in object-oriented systems. **Communications of the ACM**, v. 39, n. 10, p. 104-116, 1996.

BERGERON, F.; ST-ARNAUD, J. Y. Estimation of Information Systems Development Efforts: A Pilot Study. **Information and Management**, v. 22, n. 4, p. 239-254, 1992.

BLACKBURN, M. R.; BUSSE, R.; NAUMAN, A. Removing Requirement Defects and Automating Test. In: INTERNATIONAL CONFERENCE ON SOFTWARE TESTING ANALYSIS E REVIEW, 2001, Orlando.

BOEHM, B. W. **Software Engineering Economics**. Englewood Cliffs: Prentice-Hall, 1981. 767 p.

BOEHM, B. W. A Spiral Model of Software Development and Enhancement. **IEEE Computer**, v. 21, n. 5, p. 61-72, 1988.

BOEHM, B. W.; PAPACCIO, P. N. Understanding and Controlling Software Cost. **IEEE Transactions on Software Engineering**, v. 14, n. 10, p. 1462-1477, 1988.

BOEHM, B.; TURNER, R. **Balancing Agility and Discipline: A Guide for the Perplexed**. Boston: Addison Wesley, 2003a. 304 p.

- BOEHM, B.; TURNER, R. Using risk to balance agile and plan-driven methods. **IEEE Computer**, v. 36, n. 6, p. 57-66, 2003b.
- BRAGA, J. L. *et al.* Modelagem com dinâmica de sistemas. In: SANTOS, M. L.; VIEIRA, W. (Eds.). **Métodos Quantitativos em Economia**. Viçosa, MG: UFV, 2004. p. 411-434.
- BROOKS, F. P. **The Mythical Man-Month: Essays on Software Engineering**. 20th Anniversary Edition. Boston: Addison-Wesley, 1995. 322 p.
- CHENG, B. H. C.; ATLEE, J. M. Research Directions in Requirements Engineering. In: INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING, 29., 2007, Minneapolis. **Proceedings...** Minneapolis: IEEE Computer Society, 2007. p. 285-303.
- COCKBURN, A. **Agile Software Development**. Boston: Addison-Wesley, 2001. 256 p.
- COLLOFELLO, J. *et al.* A system dynamics software process simulator for staffing policies decision support. In: HAWAII INTERNATIONAL CONFERENCE ON SYSTEM SCIENCES, 31., 1998, Hawaii. **Proceedings...** Hawaii: IEEE Computer Society, 1998. p. 103-111.
- DAHLSTEDT, A. G.; PERSSON, A. Requirements Interdependencies – Moulding the State of Research into a Research Agenda. In: INTERNATIONAL WORKSHOP ON REQUIREMENTS ENGINEERING, 9., 2003, Austria. **Proceedings...** Austria, 2003. p. 71-80.
- DAMIAN, D.; CHISAN, J. An Empirical Study of the Complex Relationships between Requirements Engineering Processes and Other Processes that Lead to Payoffs in Productivity, Quality, and Risk Management. **IEEE Transactions On Software Engineering**, v. 32, n. 7, p. 433-453, 2006.
- DAYASINDHU, N. Embeddedness, knowledge transfer, industry clusters and global competitiveness: a case study of the Indian software industry. **Technovation**, v. 22, n. 9, p. 551-560, 2002.
- DEMARCO, T.; LISTER, T. **Waltzing with Bears: Managing Risk on Software Projects**. New York: Dorset House Publishing Co Inc, 2003. 144 p.
- DEMIRORS, O.; GENCEL, C.; TARHAN, A. Utilizing Business Process Models for Requirements Elicitation. In: EUROMICRO CONFERENCE, 29., 2003,

- Ankara, Turkey. **Proceedings...** Ankara, Turkey: IEEE Computer Society, 2003. p. 409-412.
- FILHO, W. P. P. **Engenharia de Software: Fundamentos, Métodos e Padrões**. Segunda Edição. Rio de Janeiro: LTC, 2003. 602 p.
- FORRESTER, J. W. **Industrial Dynamics**. Portland: Productivity Press, 1961. 479 p.
- GOODMAN, M. *et al.* **Designing a Systems Thinking Intervention**. Waltham, MA: Pegasus Communications, 1997. 16 p.
- GREER, D.; RUHE, G. Software release planning: an evolutionary and iterative approach. **Information and Software Technology**, v. 46, n. 4, p. 243-253, 2004.
- HANNE, T.; NEU, H. **Simulating Human Resources in Software Development Processes**. Berichte des Fraunhofer ITWM, Nr. 64, 2004. Disponível em: <<http://www.itwm.fhg.de/zentral/download/berichte/bericht64.pdf>>. Acesso em: 08 abr. 2008.
- HENDIS, K. S. Quantifying software quality. In: ACM ANNUAL CONFERENCE/ANNUAL MEETING, 1981, New York. **Proceedings...** New York: ACM, 1981. p. 268-273.
- HICKEY, A. M.; DAVIS, A. M. Requirements elicitation and elicitation technique selection: model for two knowledge-intensive software development processes. In: ANNUAL HAWAII INTERNATIONAL CONFERENCE ON SYSTEM SCIENCES, 36., 2003, Hawaii. **Proceedings...** Hawaii: IEEE Computer Society, 2003. 10 p.
- HOOD, D. J.; HOOD, C. S. Teaching Software Project Management Using Simulations. In: ANNUAL JOINT CONFERENCE INTEGRATING TECHNOLOGY INTO COMPUTER SCIENCE EDUCATION, 11., 2006, Bologna. **Proceedings...** Bologna: ACM, 2006. p. 289-293.
- HOUSTON, D. X. *et al.* Behavioral characterization: finding and using the influential factors in software process simulation models. **Journal of Systems and Software**, v. 59, n. 3, p. 259-270, 2001.
- HUGHES, J. *et al.* Presenting ethnography in the requirements process. In: IEEE INTERNATIONAL SYMPOSIUM ON REQUIREMENTS ENGINEERING, 2., 1995, York, U.K. **Proceedings...** York, U.K.: IEEE Computer Society, 1995. p. 27-34.

HUMPHREY, W. S. **The Personal Software Process (PSP)**. Technical report CMU/SEI-2000-TR-022. 2000a. 55 p. Disponível em: <<http://www.sei.cmu.edu/pub/documents/00.reports/pdf/00tr022.pdf>>. Acesso em: 08 abr. 2008.

HUMPHREY, W. S. **The Team Software Process (TSP)**. Technical report CMU/SEI-2000-TR-023. 2000b. 51 p. Disponível em: <<http://www.sei.cmu.edu/pub/documents/00.reports/pdf/00tr023.pdf>>. Acesso em: 08 abr. 2008.

JAVED, T.; MANZIL-E-MAQSOOD; DURRANI, Q. S. A Study to Investigate the Impact of Requirements Instability on Software Defects. **ACM SIGSOFT Software Engineering Notes**, v. 29, n. 3, p. 1-7, 2004.

JONES, C. Software change management. **IEEE Computer**, v. 29, n. 2, p. 80-82, 1996a.

JONES, C. **Applied Software Measurement: Assuring Productivity and Quality**. New York: McGraw-Hill, 1996b. 618 p.

JONES, C. Strategies for managing requirements creep. **IEEE Computer**, v. 29, n. 6, p. 92-94, 1996c.

JONES, C. **Estimating Software Costs**. New York: McGraw-Hill, 1998a. 724 p.

JONES, C. **The Year 2000 Software Problem: Quantifying the Costs and Assessing the Consequences**. Reading: Addison-Wesley, 1998b. 335 p.

JONES, C. **Software Assessments, Benchmarks, and Best Practices**. Boston: Addison-Wesley, 2000. 688 p.

JONES, C. **Conflict and Litigation Between Software Clients and Developers**. Technical note. 2001. Disponível em: <<http://www.spr.com/news/ConflictLitigationArticle.pdf>>. Acesso em: 08 abr. 2008.

KAPPELMAN, L. A.; MCKEEMAN, R.; ZHANG, L. Early Warning Signs of it Project Failure: The Dominant Dozen. **Information Systems Management**, v. 23, n. 4, p. 31-36, 2006.

- KING, S. F.; BURGESS, T. F. Beyond critical success factors: A dynamic model of enterprise system innovation. **International Journal of Information Management**, v. 26, n. 1, p. 59–69, 2006.
- KIRNER, T. G.; ABIB, J. C. Inspection of software requirements specification documents: a pilot study. In: ANNUAL INTERNATIONAL CONFERENCE ON COMPUTER DOCUMENTATION, 15., 1997, Utah. **Proceedings...** New York: ACM, 1997. p.161-171.
- KOTONYA, G.; SOMMERVILLE, I. **Requirements Engineering: process and techniques**. New York: John Wiley and Sons Ltd, 1998. 294 p.
- KRUCHTEN, P. **The Rational Unified Process: An Introduction**. Second Edition. Reading: Addison-Wesley, 2000. 320 p.
- KULK, G. P.; VERHOEF, C. **Quantifying Requirements Volatility**. VU University Amsterdam, Department of Computer Science, 2008. Disponível em: <<http://www.few.vu.nl/~erald/qrv/qrv.pdf>>. Acesso em: 08 abr. 2008.
- LIN, C. Y.; ABDEL-HAMID, T.; SHERIF, J. S. Software-Engineering Process Simulation Model (SEPS). **Journal of Systems and Software**, v. 38, n. 3, p. 263-277, 1997.
- MADACHY, R.; TARBET, D. Case Studies in Software Process Modeling with System Dynamics. **Software process: Improvement and Practice**, v. 5, n. 2-3, p. 133-146, 2000.
- MATSON, J. E.; BARRET, B. E.; MELLICHAMP, J. M. Software Development Cost Estimation Using Function Points. **IEEE Transactions on Software Engineering**, v. 20, n. 4, p. 275-287, 1994.
- MAXWELL, K. D.; FORSELIUS, P. Benchmarking software development productivity. **IEEE Software**, v. 17, n. 1, p. 80-88, 2000.
- MCCARL, B. A. Model Validation: An Overview with some Emphasis on Risk Models. **Review of Marketing and Agricultural Economics**, v. 52, n. 3, p. 153-173, 1984.
- MOLOEKKEN-OESTVOLD, K. *et al.* A survey on software estimation in the Norwegian industry. In: INTERNATIONAL SYMPOSIUM ON SOFTWARE METRICS, 10., 2004, Chicago. **Proceedings...** Chicago: IEEE Computer Society, 2004. p. 208-219.

- MOLOKKEN, K.; JORGENSEN, M. A Review of Surveys on Software Effort Estimation. In: INTERNATIONAL SYMPOSIUM ON EMPIRICAL SOFTWARE ENGINEERING, 2., 2003, Rome. **Proceedings...** Rome: IEEE Computer Society, 2003. p. 223-230.
- NUSEIBEH, B.; EASTERBROOK, S. Requirements engineering: a roadmap. In: INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING, 22., 2000, Limerick, Ireland. **Proceedings...** Limerick, Ireland: ACM, 2000. p. 35-46.
- PERLOW, L. A. Time to Coordinate: Toward an Understanding of Work-Time Standards and Norms in a Multicountry Study of Software Engineers. **Work and Occupations**, v. 28, n. 1, p. 91-111, 2001.
- PHAN, D. **Information Systems Project Management: an Integrated Resource Planning Perspective Model**. Thesis (Ph.D.) - Department of Management and Information Systems, University of Arizona, Arizona, 1990.
- PINDYCK, R. S.; RUBINFELD, D. L. **Microeconomia**. Sexta Edição. São Paulo: Pearson Prentice Hall, 2006. 672 p.
- PINHEIRO, F. A. C. Requirements Traceability. In: LEITE, J. C. S. P.; DOORN, J. H. (Eds.). **Perspectives on Software Requirements**. Norwell, Massachusetts: Kluwer Academic Publishers, 2004. p. 91-113.
- RACCOON, L. B. S. A Learning Curve Primer for Software Engineers. **ACM Sigsoft Software Engineering Notes**, v. 21, n. 1, p. 77-86, 1996.
- RAMESH, B.; JARKE, M. Towards Reference Models for Requirements Traceability. **IEEE Transactions on Software Engineering**, v. 27, n. 1, p. 58-93, 2001.
- RICHMOND, B. M. **An Introduction to Systems Thinking**. Lebanon, NH: High Performance Systems, 2001. 173 p.
- ROBERTS, E. B. **The Dynamics of Research and Development**. Thesis (Ph.D. in Economics) - Department of Economics, Massachusetts Institute of Technology, Cambridge, MA, 1964.
- RODRIGUES, A. G. Managing and Modelling Project Risk Dynamics: A System Dynamics-based Framework. In: EUROPEAN PROJECT MANAGEMENT CONFERENCE, PMI, 4., 2001, London.

- RODRIGUES, A.; BOWERS, J. The role of system dynamics in project management. **International Journal of Project Management**, v. 14, n. 4, p. 213-220, 1996.
- SAIEDIAN, H.; DALE, R. Requirements engineering: making the connection between the software developer and customer. **Information and Software Technology**, v. 42, n. 6, p. 419-428, 2000.
- SAYÃO, M.; LEITE, J. C. S. P. Rastreabilidade de Requisitos. **Revista de Informática Teórica e Aplicada**, v. 13, n. 1, p. 57-86, 2006.
- SBC – Sociedade Brasileira de Computação. **Grandes Desafios da Pesquisa em Computação no Brasil – 2006 – 2016**. 2006. Disponível em: <<http://www.sbc.org.br/index.php?language=1&content=downloads&id=272>>. Acesso em: 08 abr. 2008.
- SEI – Software Engineering Institute. **CMMI Web Site**. Disponível em: <<http://www.sei.cmu.edu/cmmi/>>. Acesso em: 08 abr. 2008.
- SENGE, P. **The Fifth Discipline: The Art and Practice of the Learning Organization**. New York: Currency Doubleday, 1990. 371 p.
- SENGUPTA, K.; ABDEL-HAMID, T. K.; BOSLEY, M. Coping with staffing delays in software project management: an experimental investigation. **IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans**, v. 29, n. 1, p. 77-91, 1999.
- SHEPPERD, M.; SCHOFIELD, C.; KITCHENHAM, B. Effort estimation using analogy. In: INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING, 18., 1996, Berlin. **Proceedings...** Berlin: IEEE Computer Society, 1996. p. 170-178.
- SIMMONS, D. B. Communications: a software group productivity dominator. **Software Engineering Journal**, v. 6, n. 6, p. 454-462, 1991.
- SIMMONS, D. B. Measuring and Tracking Distributed Software Development Projects. In: IEEE WORKSHOP ON FUTURE TRENDS OF DISTRIBUTED COMPUTING SYSTEMS, 9., 2003, San Juan. **Proceedings...** San Juan: IEEE Computer Society, 2003. p. 63-69.
- SOFTEX. **MPS.BR - Melhoria de Processo do Software Brasileiro – Guia Geral Versão 1.2**. 2007. 52 f. Disponível em:

<http://www.softex.br/mpsbr/%5Fguias/MPS.BR_Guia_Geral_V1.2.pdf>.
Acesso em: 08 abr. 2008.

SOMMERVILLE I. **Software Engineering**. Sixth Edition. Reading: Addison-Wesley, 2000. 693 p.

SOMMERVILLE, I.; SAWYER, P. **Requirements Engineering – a good practice guide**. New York: John Wiley and Sons Ltd, 1997. 404 p.

STANDBRIDGE J.; AUTREY, R. Rapid Skill Obsolescence In An IT Company: A Case Study Of Acxiom Corporation. **Journal of Organizational Excellence**, v. 20, n. 3, p. 3-9, 2001.

STERMAN, J. D. **System Dynamics modeling for project management**. Massachusetts Institute of Technology. 1992. Disponível em: <<http://web.mit.edu/jsterman/www/SDG/project.pdf>>. Acesso em: 08 abr. 2008.

STERMAN, J. D. **Business dynamics: systems thinking and modeling for a complex world**. Boston: Irwin McGraw-Hill, 2000. 982 p.

SUAREZ, A. J. *et al.* Qualitative Simulation of Human Resources Subsystem in Software. In: INTERNATIONAL WORKSHOP ON QUALITATIVE REASONING, 16., 2002, Barcelona.

SUDHAKAR, M. **Managing the Impact of Requirements Volatility**. 42 f. Thesis (Master in Computer Science) - Department of Computing Science, Umea University, Umea, Sweden, 2005.

TAPIA, A. H. The power of myth in the IT workplace: Creating a 24-hour workday during the dot-com bubble. **Information Technology and People**, v. 17, n. 3, p. 303-326, 2004.

VENSIM. **Vensim from Ventana Systems, Inc.** Disponível em: <<http://www.vensim.com>>. Acesso em: 08 abr. 2008.

VILLER, S.; SOMMERVILLE, I. Social Analysis in the Requirements Engineering Process: From Ethnography to Method. In: IEEE INTERNATIONAL SYMPOSIUM ON REQUIREMENTS ENGINEERING, 4., 1999, Ireland. **Proceedings...** Ireland: IEEE Computer Society, 1999. p. 6-13.

ZAHEDI, F. **Quality information Systems**. Denver: Boyd & Frazer Publishing Company, 1995. 450 p.

ZHANG, H. *et al.* Qualitative simulation model for software engineering process. In: AUSTRALIAN SOFTWARE ENGINEERING CONFERENCE, 2006, Sydney. **Proceedings...** Sydney: IEEE Computer Society, 2006. p. 391-400.

ZOWGHI, D.; OFFEN, R. A logical framework for modeling and reasoning about the evolution of requirements. In: IEEE INTERNATIONAL SYMPOSIUM ON REQUIREMENTS ENGINEERING, 3., 1997, Annapolis. **Proceedings...** Annapolis: IEEE Computer Society, 1997. p. 247-257.