

Um Modelo Dinâmico para Análise dos Impactos da Rotatividade de Pessoal Durante a Fase de Requisitos

Bernardo Giori Ambrósio¹, José Luis Braga¹, Alcione de Paiva Oliveira¹

¹Departamento de Informática – Universidade Federal de Viçosa (UFV)
Av. P. H. Rolfs, s/n – 36570-000 – Viçosa – MG – Brasil

bgambrosio@gmail.com, zeluisbraga@ufv.br, alcione@dpi.ufv.br

Resumo. *Visando a aumentar a compreensão acerca dos aspectos gerenciais envolvidos na fase de requisitos em processos de desenvolvimento de software, este trabalho apresenta e discute um modelo de dinâmica de sistemas que descreve as interações dinâmicas entre as principais variáveis envolvidas na fase de requisitos. As simulações realizadas com o modelo permitiram verificar e antever principalmente os impactos da rotatividade (turnover) de pessoal sobre a qualidade e o custo do trabalho realizado durante a fase de requisitos. Os efeitos econômicos de manter uma equipe com sobra de pessoal, como uma forma de mitigar riscos durante essa fase, foram analisados via simulações com os modelos.*

Abstract. *This paper presents and discusses the construction of system dynamics models focused on the main managerial decision variables of the requirements phase in software development. The models establish the relationships between those variables, making it possible to analyse and to understand better their mutual influences. Simulations conducted with the model made it possible to verify and to foresee the consequences of people turnover on quality and on cost of work at software requirements phase. Results of decision making regarding people turnover risks mitigation are also presented and discussed.*

1. Introdução

Como um esforço para trazer disciplina para a atividade de desenvolvimento de software, muitas organizações, na década de 70, começaram a aplicar a disciplina da engenharia no processo de desenvolvimento de software, dando origem à Engenharia de Software [Filho, 2003]. Surgiram então novas técnicas que foram aplicadas pela indústria, tais como: projeto e programação estruturada, verificação formal, programação orientada a objetos, reutilização, dentre outras.

No entanto, em parte devido ao enorme número de problemas com a produção de software de qualidade, pouca atenção tem sido dada aos aspectos gerenciais do processo de desenvolvimento de software ([Humphrey, 1990], [Abdel-Hamid e Madnick, 1991] e [Kappelman, Mckeeman e Zhang, 2006]). Os problemas de decisão nessa área 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, 2000] é uma dessas ferramentas,

constituindo 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 e Madnick, 1991], definiram modelos de dinâmica de sistemas para descrever aspectos do processo de gerenciamento de projetos de software e tentar aumentar a compreensão sistêmica sobre esse processo e suas implicações. Entretanto, não foram encontrados trabalhos que incluam as principais variáveis da fase de requisitos em seu escopo. Em muitas organizações, o processo de requisitos é deixado em plano secundário, em parte devido às pressões de mercado e ao tamanho das empresas, que não dispõem de recursos humanos em quantidade e qualidade suficientes. Jones (1996) relata que o processo de requisitos é 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. Assim, é necessário empenho para que os requisitos sejam compreendidos e especificados corretamente desde o início do projeto [Kappelman, Mckeeman e Zhang, 2006], pois 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 [Blackburn, Busser e Nauman, 2001].

Este trabalho apresenta e discute um modelo de dinâmica de sistemas que descreve as interações dinâmicas entre as principais variáveis envolvidas na fase de requisitos em processos de desenvolvimento de software. As simulações realizadas com o modelo permitiram antever os impactos da rotatividade de profissionais (*turnover*) alocados ao projeto sobre a qualidade e o custo do trabalho realizado durante a fase de requisitos. Também foi possível antecipar, por meio de simulações, os possíveis efeitos da decisão gerencial de manter a equipe com um tamanho maior que o necessário, visando a mitigar os impactos do *turnover*.

Segundo [Cheng e Atlee, 2007], a contribuição deste trabalho situa-se na Gerência de Requisitos, apoiando o gerenciamento das atividades realizadas durante a fase de requisitos. Com relação aos Grandes Desafios da Pesquisa em Computação no Brasil 2006-2016 [SBC, 2006], este trabalho se enquadra no quinto desafio, “Desenvolvimento tecnológico de qualidade: sistemas disponíveis, corretos, seguros, escaláveis, persistentes e ubíquos”.

Na seção 2 são apresentados conceitos sobre *turnover* de pessoal e dinâmica de sistemas, e uma descrição sobre trabalhos correlatos. A seção 3 descreve o modelo de dinâmica de sistemas construído. A seção 4 apresenta os resultados de simulações realizadas com o modelo. Finalmente, a seção 5 apresenta as conclusões obtidas.

2. Contexto

2.1. *Turnover* de pessoal em tecnologia da informação

A expansão da tecnologia da informação e da economia digital impõem desafios para as organizações manterem seus profissionais mais capacitados. 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 e Slaughter, 2004], causando o denominado *turnover*.

Diversos fatores podem determinar a ocorrência de *turnover* entre profissionais do setor de tecnologia da informação. Em [Igarria e Guimarães, 1999] é investigada a

relação entre a satisfação profissional, o grau de confiança na empresa e a taxa de *turnover* entre os profissionais. Ang e Slaughter (2004) discutem os efeitos da política de recursos humanos sobre o *turnover* e, a partir de coletas de dados e entrevistas com gerentes de RH, afirmam que os cargos técnicos apresentam uma taxa de *turnover* maior que os cargos gerenciais.

DeMarco e Lister (2003) classificam o *turnover* entre os cinco tipos de risco mais recorrentes em projetos de software. Quando o *turnover* ocorre, pode ser necessário adicionar novas pessoas na equipe. No entanto, a alocação de pessoas não ocorre imediatamente [Sengupta, Abdel-Hamid e Bosley, 1999]. É necessário algum tempo para remanejar profissionais de outras equipes ou contratar profissionais disponíveis no mercado. Além disso, os membros novatos da equipe, que no contexto deste trabalho não são necessariamente inexperientes, apresentam uma produtividade menor e cometem mais erros que os membros experientes [Lin, Abdel-Hamid e Sherif, 1997]. O efeito final pode ser o atraso dos compromissos, o aumento no custo, e/ou o aumento na quantidade de erros cometidos.

2.2. Dinâmica de sistemas

A dinâmica de sistemas é uma técnica descritiva, utilizada para modelagem e simulação de sistemas, baseada no pensamento e análise sistêmicos e na teoria matemática dos Sistemas Dinâmicos [Sterman, 2000]. 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], permitindo antecipar colapsos.

Um modelo de dinâmica de sistemas possui dois elementos principais: os estoques, que são os recursos acumuláveis do sistema, e os fluxos, que são responsáveis pela variação dos estoques. Tomando como exemplo a Figura 2, são exemplos de estoques: *Força de Trabalho Novatos* e *Força de Trabalho Experientes*; são exemplos de fluxos: *Taxa de Alocação*, *Taxa de Aprendizado*, *Taxa de Turnover*, *Taxa de Remoção de Novatos* e *Taxa de Remoção de Experientes*. Há também os conversores ou variáveis, que são os elementos do modelo que exercem influência sobre os valores dos fluxos. Ainda na Figura 2, são exemplos de variáveis: *força de trabalho total*, *força de trabalho alcançável* e *saldo da força de trabalho*. Os dois traços paralelos que cruzam algumas ligações entre os elementos indicam que o efeito de um elemento sobre o outro só é observável com uma defasagem de tempo.

Recomenda-se obter diagramas de influência [Senge, 1990] nas fases iniciais da modelagem, pois eles permitem entender melhor o relacionamento entre as variáveis do problema e a identificação de laços de realimentação no modelo. Um diagrama de influência pode ser visualizado na Figura 1. Em 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).

2.3. Trabalhos correlatos

Entre os vários trabalhos que relatam a aplicação da dinâmica de sistemas na modelagem de aspectos do processo de gerenciamento de projetos de software, o descrito em [Abdel-Hamid e Madnick, 1991], que investiga as relações entre diversas

variáveis que afetam o desenvolvimento de software, é um dos mais referenciados. Outros trabalhos realizados com o mesmo objetivo são: [Abdel-Hamid, 1993], [Abdel-Hamid, 1996], [Lin, Abdel-Hamid e Sherif, 1997], [Collofello *et al.*, 1998], [Sengupta, Abdel-Hamid e Bosley, 1999], [Zhang *et al.*, 2006], dentre outros. No entanto, esses trabalhos não incluíram a modelagem das variáveis e dos relacionamentos envolvidos na fase de requisitos dos processos de desenvolvimento de software.

3. Um modelo dinâmico para análise dos impactos do *turnover*

O modelo de dinâmica de sistemas apresentado neste trabalho abrange as principais variáveis envolvidas na fase de requisitos e alguns dos relacionamentos existentes entre elas. As relações entre as variáveis foram definidas, quantificadas e fundamentadas a partir de informações disponibilizadas em artigos científicos citados ao longo do texto. O escopo do modelo é geral, incluindo todas as atividades e artefatos produzidos durante a fase de requisitos de projetos de software. Para construir o modelo e realizar as simulações, foi utilizada uma versão gratuita, para uso acadêmico, da ferramenta Vensim [Vensim, 2008].

Visando a facilitar o entendimento do modelo, os nomes dos seus elementos são escritos em *itálico*, sendo que os nomes dos estoques e dos fluxos iniciam com caractere maiúsculo, e os nomes dos conversores (variáveis) iniciam com caractere minúsculo. O modelo utiliza a medida *pontos de função* [Matson, Barret e Mellichamp, 1994] para estimar o tamanho e a complexidade do trabalho a ser realizado. Para estimar o esforço necessário, é utilizada a medida *homens-dia*. O modelo adapta a formulação apresentada em [Filho, 2003], relacionando essas duas medidas, segundo a equação: Esforço (em *homens-dia*) = Tamanho (em *pontos de função*) / Produtividade (em *pontos de função por homem-dia*).

3.1. Impactos do *turnover* sobre a qualidade

O modelo utiliza o estoque *Requisitos com Erros* para medir a qualidade da especificação de requisitos. Esse estoque representa todos os requisitos especificados que apresentam erros ainda não detectados. Após o término da simulação, o valor desse estoque corresponde à quantidade residual de requisitos especificados incorretamente que serão entregues para as fases seguintes do processo de desenvolvimento.

Como pode ser visto no diagrama de influência da Figura 1 e nas subseções seguintes, o *turnover* contribui para aumentar a quantidade de erros de requisitos devido a: - aumento na porcentagem de membros novatos na equipe (região 1); - aumento no uso do esforço extra da equipe devido às pressões de prazo (região 2); - redução no esforço alocado para as atividades de garantia de qualidade (região 3).

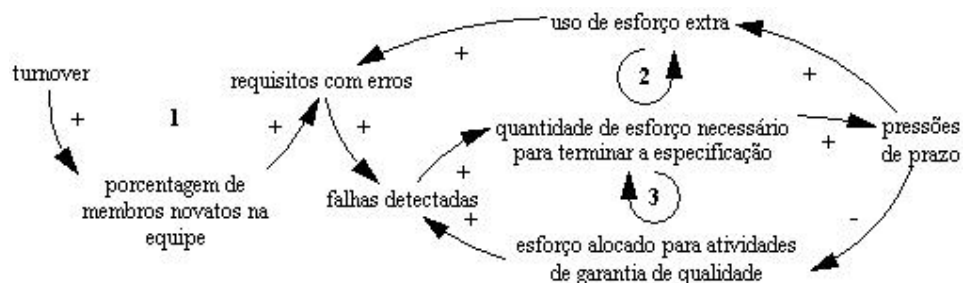


Figura 1. Modelo utilizado para entender os impactos do *turnover* sobre a qualidade

membros novatos na equipe, o que aumenta também o valor da variável *força de trabalho total*, formando um *loop* de balanço ou de equilíbrio.

As relações entre as variáveis nos modelos podem formar dois tipos de *loops*: *loop* de reforço e *loop* de balanço. O *loop* de reforço corresponde a uma realimentação positiva, em que o valor de uma ou mais variáveis aumenta ou diminui indefinidamente. Mas na prática, existe um limite para o crescimento ou decrescimento do valor da(s) variável(is). O *loop* de balanço é responsável por equilibrar os efeitos do *loop* de reforço, permitindo obter uma situação de equilíbrio no sistema.



Figura 3. Modelo utilizado para analisar o *loop* de balanço que equilibra os efeitos do *turnover*

Para equilibrar os efeitos do *turnover* são alocados membros novatos na equipe. Isso contribui para aumentar a quantidade de erros cometidos ([Abdel-Hamid e Madnick, 1991] e [Lin, Abdel-Hamid e Sherif, 1997]). Na Figura 4, o aumento no valor da variável *porcentagem de membros novatos na equipe* acarreta um aumento no valor da variável *probabilidade de cometer erros*.

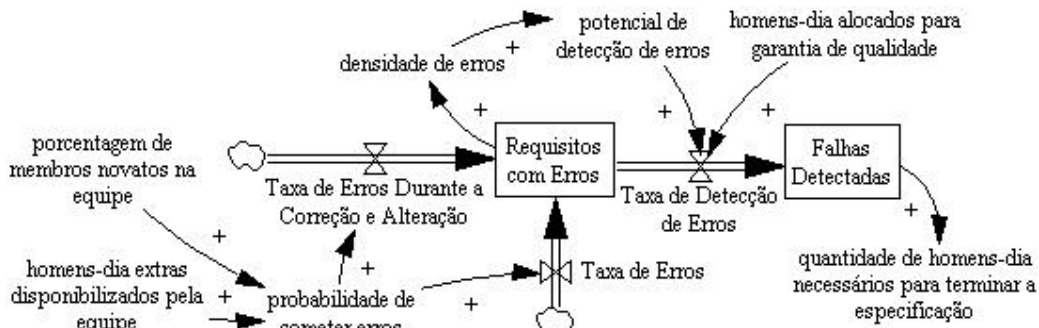


Figura 4. Modelo utilizado para analisar o aumento na quantidade de erros cometidos

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. Esses fluxos acumulam requisitos no estoque *Requisitos com Erros*. As subseções que seguem exploram outras relações no modelo, em uma sequência semelhante à utilizada neste item, e serão abreviadas por questões de espaço.

3.1.2. Aumento no uso do esforço extra da equipe

A porção do modelo apresentada na Figura 4 descreve como o aumento na quantidade de erros cometidos acarreta um aumento na quantidade de esforço necessário para terminar a especificação dos requisitos. As atividades de garantia de qualidade visam a detectar os erros cometidos, evitando a propagação de defeitos ao longo do projeto. 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 e o valor da variável *potencial de detecção de erros* determinam o valor do fluxo *Taxa de Detecção de Erros*. Esse fluxo representa a taxa com que os erros são detectados pela equipe e transfere requisitos do estoque *Requisitos com Erros* para o estoque *Falhas Detectadas*. O valor da variável *densidade de erros* afeta o valor da variável *potencial de detecção de erros* [Lin, Abdel-Hamid e Sherif, 1997].

A variável *quantidade de homens-dia necessários para terminar a especificação* representa o esforço necessário para terminar a especificação dos requisitos. Considerando que as falhas detectadas devem ser corrigidas pela equipe, o aumento no valor do estoque *Falhas Detectadas* acarreta um aumento no valor da variável *quantidade de homens-dia necessários para terminar a especificação*.

O modelo na Figura 5 permite entender como o aumento na quantidade de esforço necessário pode provocar um aumento nas pressões de prazo sobre a equipe. Os valores da variável *quantidade de homens-dia necessários para terminar a especificação* e do estoque *Prazo Planejado* determinam o valor da variável *força de trabalho necessária* [Abdel-Hamid, 1993]. Essa variável representa a quantidade de pessoas que devem compor a equipe para que a especificação termine no prazo planejado.



Figura 5. Modelo que descreve o aumento nas pressões de prazo sobre a equipe

Devido a restrições, tais como o limite na quantidade de membros novatos na equipe [Abdel-Hamid e Madnick, 1991] e o tempo necessário para a alocação de novos profissionais [Sengupta, Abdel-Hamid e Bosley, 1999], a quantidade de pessoas que pode compor a equipe em algum instante é representada pela variável *força de trabalho alcançável*. Após a alocação ou 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*. 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 que irá faltar para a realização das atividades, impedindo que a especificação termine no prazo planejado.

Quando há risco de o projeto atrasar e o prazo não pode ser adiado, as pressões de prazo aumentam e os membros da equipe são incentivados a trabalhar com mais empenho ([Abdel-Hamid, 1996] e [Lin, Abdel-Hamid e Sherif, 1997]), disponibilizando esforço extra. Isso é representado pela influência da variável *pressões de prazo* sobre a variável *homens-dia extras disponibilizados pela equipe*. O aumento no uso do esforço

extra acarreta um aumento na quantidade de esforço alocado para realizar as atividades, o que reduz a quantidade de esforço necessário para o término da especificação. Isso diminui as pressões de prazo sobre a equipe, formando um *loop* de balanço.

A sobrecarga de trabalho acarreta estresse e exaustão da equipe, contribuindo para aumentar a quantidade de erros cometidos ([Lin, Abdel-Hamid e Sherif, 1997] e [Collofello *et al.*, 1998]). Conforme pode ser visto no modelo da Figura 4, o aumento no valor da variável *homens-dia extras disponibilizados pela equipe* provoca um aumento no valor da variável *probabilidade de cometer erros*.

3.1.3. Redução no esforço alocado para as atividades de garantia de qualidade

Segundo [Abdel-Hamid e Madnick, 1991] e [Abdel-Hamid, 1993], quando o projeto está atrasado, os recursos são concentrados na recuperação do atraso, e as atividades de garantia de qualidade são deixadas em plano secundário. As pressões de prazo provocam uma redução no esforço alocado para as atividades de garantia de qualidade, que visam a detectar os erros cometidos, contribuindo para um maior acúmulo no estoque *Requisitos com Erros*.

3.2. Impactos do *turnover* sobre o custo

Para medir o custo para a realização da especificação de requisitos, o modelo utiliza o estoque *Homens-dia Gastos*, apresentado na Figura 6. Esse estoque armazena a quantidade de esforço, medido em homens-dia, utilizado até o momento. O fluxo *Taxa de Gasto de Homens-dia* representa a quantidade de homens-dia utilizados por dia.

De acordo com os elementos do modelo, o *turnover* pode acarretar um aumento no custo do trabalho realizado devido ao: - **aumento no uso de esforço extra:** 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; - **aumento na quantidade de esforço necessário:** como o *turnover* acarreta um aumento na quantidade de erros cometidos, deve ser alocado esforço adicional para realizar as correções à medida que os erros são detectados.

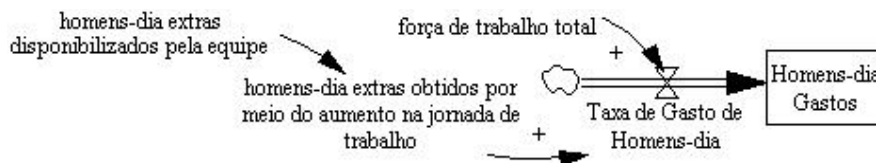


Figura 6. Elementos do modelo que determinam o custo do trabalho realizado

3.2.1. Aumento no uso de esforço extra

Na seção 3.1.1 foi explicado que o *turnover* inicialmente provoca uma redução do tamanho da equipe, e após ocorrer a alocação de novos profissionais, para assumir as atividades dos que saíram, ocorre um aumento na porcentagem de membros novatos na equipe. Os membros novatos, em geral, possuem uma produtividade menor que os membros experientes [Lin, Abdel-Hamid e Sherif, 1997], o que contribui para um declínio na taxa de produtividade com que as atividades são realizadas [Zhang *et al.*, 2006].

Com a redução da produtividade da equipe, pode surgir um risco de atraso no projeto, o que aumenta as pressões de prazo sobre a equipe. Conforme foi explicado na

seção 3.1.2, diante do aumento nas pressões de prazo, os membros da equipe são incentivados a trabalhar com mais empenho, e a jornada de trabalho pode ser aumentada, se for necessário. O modelo na Figura 6 mostra que o uso do esforço extra obtido por meio do aumento da jornada de trabalho contribui para aumentar o custo.

3.2.2. Aumento na quantidade de esforço necessário

Na seção 3.1 foi descrito como o *turnover* provoca um aumento na quantidade de erros cometidos. À medida que os erros são detectados pela equipe, deve ser alocado esforço para realizar as correções. No modelo apresentado na Figura 4, isso é representado pela relação entre o estoque *Falhas Detectadas* e a variável *quantidade de homens-dia necessários para terminar a especificação*. Para tratar o aumento na quantidade de esforço necessário, pode ser preciso: alocar novos profissionais na equipe, utilizar esforço extra, e/ou adiar o prazo. Todas as alternativas provocam o aumento do custo.

4. Simulações utilizando o modelo

O modelo construído pode ser utilizado como uma ferramenta para apoiar os gerentes na tomada de decisão durante a fase de requisitos. As simulações realizadas com o modelo permitem avaliar e antever os impactos da materialização de riscos e os efeitos das decisões gerenciais planejadas, antes de elas serem implementadas.

A seguir são apresentados os resultados de algumas simulações do modelo que permitem verificar os impactos do *turnover* durante a fase de requisitos e as conseqüências da decisão gerencial de manter a equipe com um tamanho maior que o necessário, visando a mitigar os impactos do *turnover*. Inicialmente são apresentados os resultados obtidos ao utilizar dados disponibilizados na literatura para ajustar o modelo. Em seguida, são apresentados os resultados obtidos quando são utilizados dados coletados em uma empresa de desenvolvimento de software.

4.1. Simulações com dados disponibilizados na literatura

Antes de executar uma simulação, é necessário definir os valores de algumas variáveis do modelo para ajustá-lo de acordo com: o cenário que será simulado e as características da organização, da equipe e do projeto que definem o contexto da simulação. A Tabela 1 apresenta os valores atribuídos a algumas variáveis do modelo.

Tabela 1. Valores utilizados para configurar o modelo

1- <i>Quantidade de requisitos especificados e entregues em uma liberação</i> : 120 requisitos estimados em 120 pontos de função
2- <i>Produtividade média dos integrantes da equipe</i> : 2 pontos de função / homem-dia
3- <i>Tamanho inicial da equipe responsável pela especificação</i> : 2
4- <i>Probabilidade de cometer erros durante a especificação dos requisitos</i> : 12%
5- <i>Probabilidade de alteração de requisitos</i> : 3%
6- <i>Porcentagem dos requisitos de uma liberação que não são previstos antes de iniciar a especificação</i> : 0% (todos os requisitos são previstos)
7- <i>Porcentagem do esforço disponibilizado pela equipe que é alocado para as atividades de garantia de qualidade</i> : 20%
8- <i>Aumento no esforço disponibilizado pelos membros da equipe quando há pressões de prazo e risco de atraso</i> : 0,5 homens-dia (até 50% a mais de esforço)

Os valores definidos para as variáveis 1 e 2 são baseados na experiência dos autores e em trocas de informações não documentadas com profissionais de empresas de desenvolvimento de software. Para definir os valores das variáveis 3 e 4 foram utilizados dados fornecidos por uma empresa média, local, de desenvolvimento de software. Os valores utilizados para configurar as demais variáveis (5 a 8) foram definidos a partir de dados disponibilizados na literatura, relatados por outros grupos de pesquisa.

Alguns trabalhos que podem direcionar a definição dos valores das variáveis 5 a 8, apresentadas na Tabela 1, são: - variável 5: [Jones, 2001], [Javed, Manzil-E-Maqsood e Durrani, 2004] e [Kulk e Verhoef, 2008]; - variável 6: [Molokken e Jorgensen, 2003] e [Molokken-Oestvold *et al.*, 2004]; - variável 7: [Abdel-Hamid e Madnick, 1991]; - variável 8: [Abdel-Hamid e Madnick, 1991], [Perlow, 2001] e [Tapia, 2004].

Os gráficos apresentados na Figura 7 mostram as variações na qualidade e no custo do trabalho em função do prazo planejado. Os valores nos gráficos permitem comparar dois cenários: um em que não ocorre *turnover* e outro 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.

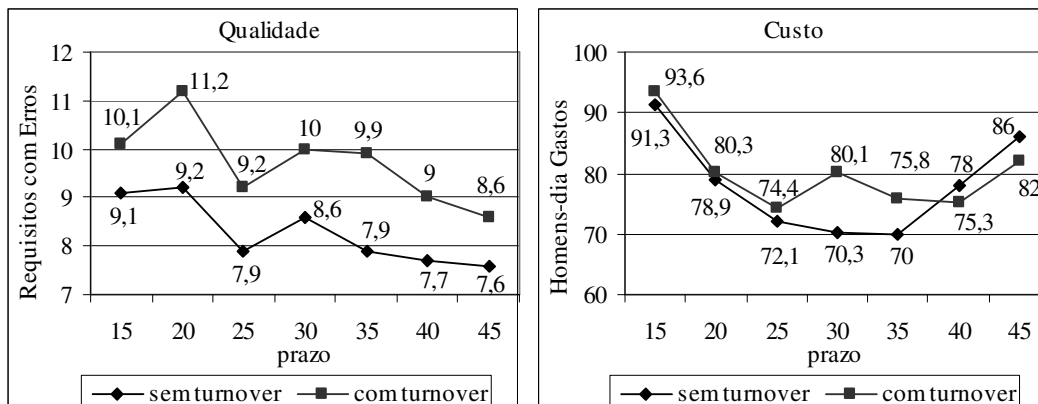


Figura 7. Resultados das simulações que verificam os impactos do *turnover*

O gráfico que descreve as variações na qualidade mostra que quando ocorre *turnover*, há um aumento na quantidade de erros de especificação que serão entregues para as fases seguintes do processo de desenvolvimento. Além disso, a redução no prazo planejado tende a aumentar a quantidade de erros de especificação nos dois cenários. Para terminar a especificação em um tempo menor, pode ser necessário aumentar o tamanho da equipe e/ou utilizar esforço extra dos profissionais ([Abdel-Hamid, 1996] e [Lin, Abdel-Hamid e Sherif, 1997]), o que contribui para aumentar a quantidade de erros cometidos ([Lin, Abdel-Hamid e Sherif, 1997] e [Collofello *et al.*, 1998]).

Para o cenário em que não ocorre *turnover*, o gráfico de variação no custo mostra que um prazo em torno de 30 a 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 horas-extras dos profissionais ([Abdel-Hamid, 1996] e [Lin, Abdel-Hamid e Sherif, 1997]), o que implica no aumento do custo. Mas, quando o prazo planejado é superestimado, tornando-se maior que o necessário, a equipe irá trabalhar com menor produtividade, absorvendo uma parte do

excesso de tempo disponível [Abdel-Hamid e Madnick, 1991]. Isso explica o aumento que ocorre no custo quando o prazo planejado é maior que 35 dias.

O custo mínimo ocorre próximo ao prazo de 30 dias devido aos valores atribuídos às variáveis 1, 2 e 3, na Tabela 1. 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 correções e alterações de requisitos e atividades de garantia de qualidade, o que aumenta o custo. No entanto, o maior empenho da equipe durante o horário normal de trabalho pode amenizar os impactos sobre o custo, mantendo o prazo ideal próximo a 30 dias.

O gráfico de custo mostra também 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. Quando o prazo estimado é maior que o necessário e um integrante da equipe abandona o projeto, pode não ser necessário alocar um novo membro para substituí-lo. Nesse caso, o *turnover* pode contribuir para diminuir o custo, como é mostrado no gráfico.

É comum ocorrerem oscilações nas curvas dos gráficos ao utilizá-los para organizar e visualizar os resultados das simulações do modelo. 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*), o que explica o comportamento observado e as oscilações nas curvas dos gráficos.

Na Figura 8, 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 os impactos do *turnover* [DeMarco e 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, até a ocorrência do *turnover*.

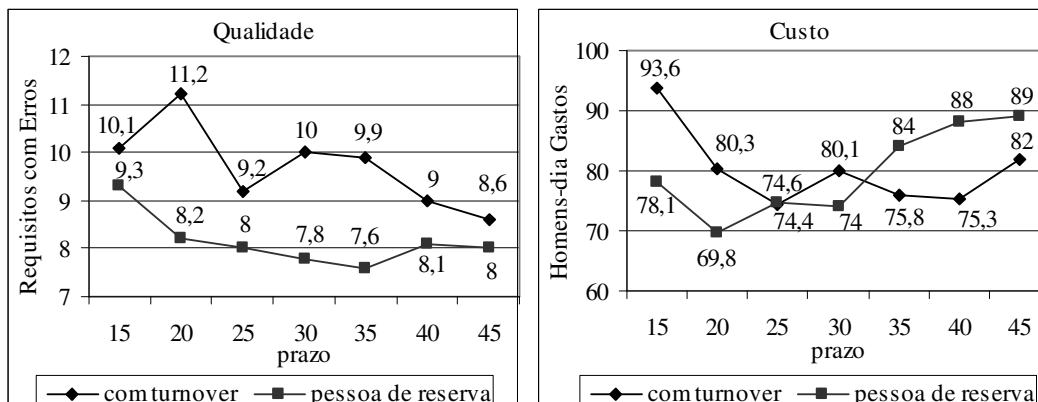


Figura 8. Resultados das simulações que verificam os efeitos de manter uma pessoa de reserva na equipe

O gráfico que descreve as variações na qualidade mostra que manter uma pessoa de reserva contribui para diminuir a quantidade de erros de especificação. 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 variação no custo mostra que, diante da ocorrência de *turnover*, a decisão de manter uma pessoa de reserva na equipe tende a diminuir o custo quando o prazo é menor que 35 dias. Mas, quando o prazo aumenta, essa decisão tende a aumentar o custo porque com uma equipe maior, a folga no prazo torna-se mais evidente. A equipe irá então trabalhar com menor produtividade, absorvendo parte do excesso de tempo disponível.

4.2. Simulações com dados coletados em uma empresa de desenvolvimento de software

Para configurar o modelo e realizar as simulações apresentadas nessa seção, foram utilizados dados coletados em uma empresa que desenvolve sistemas de software para uso no domínio científico e tecnológico. A empresa consultada não mantém uma base organizada de dados acerca dos projetos realizados. Os valores, que são baseados em um projeto de tamanho médio, foram estimados por um Diretor e por um Gerente de Projetos a partir da sua experiência obtida na participação em projetos. 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.

A Tabela 2 apresenta os valores baixo, médio (mais provável) e alto obtidos para algumas variáveis do modelo. Diante da impossibilidade de estimar os valores das variáveis 1 e 2, as mesmas receberam valores idênticos aos definidos na seção 4.1. A variável *quantidade de requisitos especificados e entregues em uma liberação* recebeu o valor 120, e a variável *produtividade média dos integrantes da equipe* recebeu o valor 2 pontos de função / homem-dia.

Tabela 2. Valores coletados em uma empresa de desenvolvimento de software

Variável	Baixo	Médio	Alto
1- <i>Quantidade de requisitos especificados e entregues em uma liberação</i>	Não foi possível estimar		
2- <i>Produtividade média dos integrantes da equipe</i>	Não foi possível estimar		
3- <i>Tamanho inicial da equipe responsável pela especificação</i>	2	2	3
4- <i>Probabilidade de cometer erros durante a especificação dos requisitos</i>	5%	15%	20%
5- <i>Probabilidade de alteração de requisitos</i>	10%	20%	30%
6- <i>Porcentagem dos requisitos de uma liberação que não são previstos antes de iniciar a especificação</i>	15%	20%	25%
7- <i>Porcentagem do esforço disponibilizado pela equipe que é alocado para as atividades de garantia de qualidade</i>	5%	10%	15%
8- <i>Aumento no esforço disponibilizado pelos membros da equipe quando há pressões de prazo e risco de atraso</i>	10%	15%	20%

Os gráficos apresentados na Figura 9 permitem comparar dois cenários configurados com os valores médios informados na Tabela 2: um em que não ocorre *turnover* e outro em que ocorre *turnover*. O gráfico de variação na qualidade mostra que quando ocorre *turnover*, há um aumento na quantidade de erros de especificação, exceto

quando o prazo planejado é igual a 45 dias. O gráfico de variação no 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. Um comportamento semelhante ocorre com as simulações apresentadas na seção anterior, que utilizaram dados disponibilizados na literatura para ajustar o modelo.

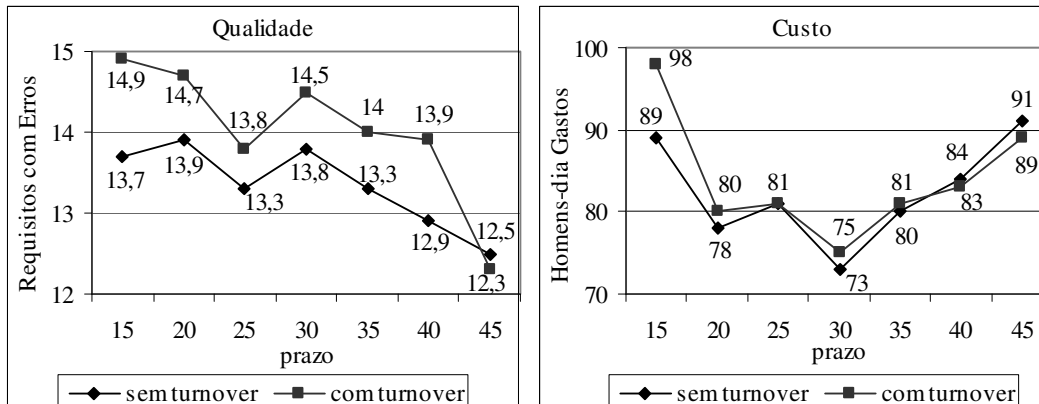


Figura 9. Resultados das simulações que verificam os impactos do *turnover*

Na Figura 10, os gráficos comparam um cenário em que ocorre *turnover* com outro cenário em que a ocorrência de *turnover* é combinada com a decisão gerencial de manter uma pessoa de reserva na equipe. Ambos os cenários foram configurados com os valores médios informados na Tabela 2. O gráfico de variação na qualidade mostra que manter uma pessoa de reserva contribui para diminuir a quantidade de erros de especificação. O gráfico de variação no 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, a decisão de manter uma pessoa de reserva na equipe contribui para aumentar o custo. Um comportamento semelhante ocorre com as simulações apresentadas na seção anterior, que utilizaram dados disponibilizados na literatura para ajustar o modelo.

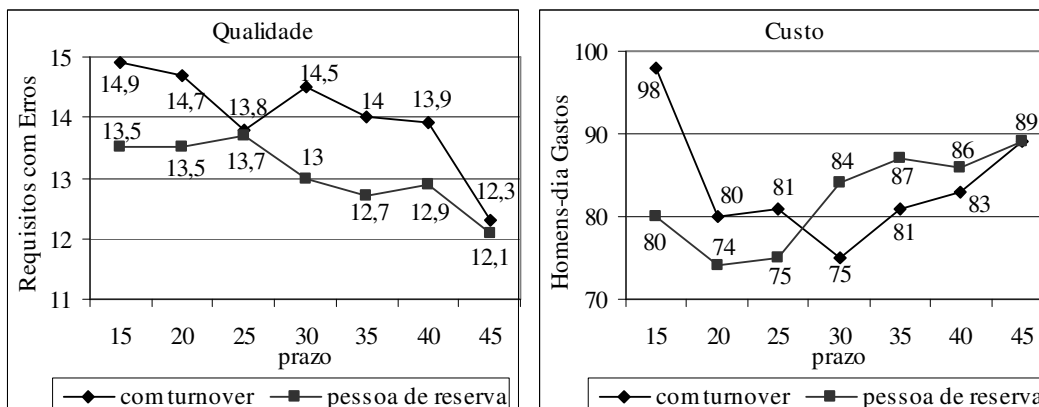


Figura 10. Resultados das simulações que verificam os efeitos de manter uma pessoa de reserva na equipe

5. Conclusões

As discussões e simulações apresentadas mostram como o modelo pode ser utilizado para analisar os impactos do *turnover*, durante a fase de requisitos. O modelo também permite realizar outras análises gerenciais que consistem em verificar: - os impactos da materialização de riscos como uma alta volatilidade dos requisitos e o aumento na quantidade de erros cometidos; - os impactos de cometer erros ao estimar a quantidade de requisitos da liberação; - os efeitos do prazo planejado 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; - os efeitos de uma maior ou menor tendência em alterar o prazo planejado, modificar o tamanho da equipe e utilizar esforço extra da equipe.

Em simulações realizadas com vários outros cenários, o padrão de variação e o valor de algumas variáveis tais como esforço utilizado, erros cometidos, prazo e tamanho da equipe, mostraram-se de acordo com o conhecimento disponível na literatura da área de Engenharia de Software. A semelhança entre o comportamento do modelo nas simulações que utilizaram dados disponibilizados na literatura e o comportamento obtido nas simulações que utilizaram dados coletados em uma empresa de desenvolvimento de software constitui uma evidência de que o comportamento do modelo é compatível com situações mais gerais, sendo menos dependente das características particulares de empresas e de projetos.

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 precisa.

Diante da impossibilidade de se reproduzir um projeto de software para se estudar as conseqüências das modificações em alguns fatores e variáveis que o afetam, os modelos surgem como uma alternativa viável para a criação dos “laboratórios de aprendizagem” [Serman, 2000] 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.

Como expansão deste trabalho, o modelo apresentado pode ser utilizado como base para a implementação de um simulador gerencial para treinar gerentes no processo de tomada de decisão. O simulador apresentaria as informações sobre o projeto sendo simulado 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, entre outras.

Agradecimentos

Este trabalho foi parcialmente financiado pela FAPEMIG, com recursos do projeto ForteSIG – Integração e Fortalecimento do Grupo de Pesquisa em Sistemas de Informação Geográfica da Área de Computação da UFV (CEX 80768/04).

XXII Simpósio Brasileiro de Engenharia de Software

Referências bibliográficas

- Abdel-Hamid, T. K. (1993). Adapting, correcting, and perfecting software estimates: a maintenance metaphor. *IEEE Computer*, v. 26, n. 3, p. 20-29.
- Abdel-Hamid, T. K. (1996). The slippery path to productivity improvement. *IEEE Software*, v. 13, n. 4, p. 43-52.
- Abdel-Hamid, T. K. e Madnick, S. E. (1991). *Software Project Dynamics: an Integrated Approach*. Englewood Cliffs: Prentice Hall. 264 p.
- Ang, S. e Slaughter, S. (2004). Turnover of information technology professionals: the effects of internal labor market strategies. *ACM SIGMIS Database*, v. 35, n. 3, p. 11-27.
- Blackburn, M. R., Busser, R. e Nauman, A. (2001). Removing Requirement Defects and Automating Test. In: *International Conference on Software Testing Analysis and Review*, Orlando.
- Cheng, B. H. C. e Atlee, J. M. (2007). Research Directions in Requirements Engineering. In: *International Conference on Software Engineering*, 29., Minneapolis. *Proceedings...* Minneapolis: IEEE Computer Society, 2007. p. 285-303.
- Collofello, J. *et al.* (1998). A system dynamics software process simulator for staffing policies decision support. In: *Hawaii International Conference on System Sciences*, 31., Hawaii. *Proceedings...* Hawaii: IEEE Computer Society, 1998. p. 103-111.
- DeMarco, T. e Lister, T. (2003). *Waltzing with Bears: Managing Risk on Software Projects*. New York: Dorset House Publishing Co Inc. 144 p.
- Filho, W. P. P. (2003). *Engenharia de Software: Fundamentos, Métodos e Padrões*. Segunda Edição. Rio de Janeiro: LTC. 602 p.
- Humphrey, W. S. (1990). *Managing the software process*. Reading, Massachusetts: Addison-Wesley. 512 p.
- Igbaria, M. e Guimaraes, T. (1999). Exploring Differences in Employee Turnover Intentions among Telecommuters and Non-Telecommuters. *Journal of Management Information Systems*, v. 16, n. 1, p. 147-164.
- Javed, T., Manzil-E-Maqsood e Durrani, Q. S. (2004). A Study to Investigate the Impact of Requirements Instability on Software Defects. *ACM SIGSOFT Software Engineering Notes*, v. 29, n. 3, p. 1-7.
- Jones, C. (1996). *Applied Software Measurement: Assuring Productivity and Quality*. New York: McGraw-Hill. 618 p.
- Jones, C. (2001). *Conflict and Litigation Between Software Clients and Developers*. Technical note. Disponível em: <<http://www.spr.com/news/ConflictLitigationArticle.pdf>>. Acesso em: 08 abr. 2008.
- Kappelman, L. A., Mckeeman, R. e Zhang, L. (2006). Early Warning Signs of it Project Failure: The Dominant Dozen. *Information Systems Management*, v. 23, n. 4, p. 31-36.

XXII Simpósio Brasileiro de Engenharia de Software

- Kulk, G. P. e Verhoef, C. (2008). *Quantifying Requirements Volatility*. VU University Amsterdam, Department of Computer Science. Disponível em: <<http://www.few.vu.nl/~erald/qrv/qrv.pdf>>. Acesso em: 08 abr. 2008.
- Lin, C. Y., Abdel-Hamid, T. e Sherif, J. S. (1997). Software-Engineering Process Simulation Model (SEPS). *Journal of Systems and Software*, v. 38, n. 3, p. 263-277.
- Matson, J. E., Barret, B. E. e Mellichamp, J. M. (1994). Software Development Cost Estimation Using Function Points. *IEEE Transactions on Software Engineering*, v. 20, n. 4, p. 275-287.
- McCarl, B. A. (1984). Model Validation: An Overview with some Emphasis on Risk Models. *Review of Marketing and Agricultural Economics*, v. 52, n. 3, p. 153-173.
- Moloekken-Oestvold, K. *et al.* (2004). A survey on software estimation in the Norwegian industry. In: *International Symposium on Software Metrics*, 10., Chicago. *Proceedings...* Chicago: IEEE Computer Society, 2004. p. 208-219.
- Molokken, K. e Jorgensen, M. (2003). A Review of Surveys on Software Effort Estimation. In: *International Symposium on Empirical Software Engineering*, 2., Rome. *Proceedings...* Rome: IEEE Computer Society, 2003. p. 223-230.
- Perlow, L. A. (2001). 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.
- Raccoon, L. B. S. (1996). A Learning Curve Primer for Software Engineers. *ACM SIGSOFT Software Engineering Notes*, v. 21, n. 1, p. 77-86.
- SBC – Sociedade Brasileira de Computação (2006). *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.
- Senge, P. (1990). *The Fifth Discipline: The Art and Practice of the Learning Organization*. New York: Currency Doubleday. 371 p.
- Sengupta, K., Abdel-Hamid, T. K. e Bosley, M. (1999). 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.
- Sterman, J. D. (2000). *Business dynamics: systems thinking and modeling for a complex world*. Boston: Irwin McGraw-Hill. 982 p.
- Tapia, A. H. (2004). 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.
- Vensim (2008). *Vensim from Ventana Systems, Inc.* Disponível em: <<http://www.vensim.com>>. Acesso em: 08 abr. 2008.
- Zhang, H. *et al.* (2006). Qualitative simulation model for software engineering process. In: *Australian Software Engineering Conference*, Sydney. *Proceedings...* Sydney: IEEE Computer Society, 2006. p. 391-400.