

# Modelagem da atividade de elicitação de requisitos utilizando a técnica de entrevista: uma abordagem utilizando dinâmica de sistemas

Victor O. Hermsdorf<sup>1</sup>, José Luis Braga<sup>1</sup>, Leonardo Carvalho<sup>1</sup>, Liziane S. Soares<sup>1</sup>,  
Bernardo G. Ambrósio<sup>2</sup>

<sup>1</sup>Universidade Federal de Viçosa, Viçosa, MG, Brasil

<sup>2</sup>Universidade Federal de Ouro Preto, João Monlevade, MG, Brasil

{ vhermsdorf,zeluis,lcarvalho}@dpi.ufv.br, liziane.soares@ufv.br, bernardo@decea.ufop.br

**Abstract.** O desenvolvimento de software é uma atividade que envolve riscos, parte deles relacionados a problemas de decisão, principalmente no nível gerencial. São muitas variáveis dinâmicas interconectadas a serem gerenciadas, sendo indispensável o uso de ferramentas que auxiliem os gerentes em suas tomadas de decisão, identificando os riscos antes de eles se materializarem. O objetivo deste trabalho é produzir um modelo de dinâmica de sistemas que aborde a atividade de elicitação de requisitos utilizando a técnica de entrevista, permitindo que gerentes possam utilizá-lo como uma ferramenta de apoio para auxiliá-los na análise e tomadas de decisão relativas à adoção desta técnica em projetos.

**Keywords:** requisito de software, dinâmica de sistemas, elicitação de requisitos

## 1 Introdução

Problemas de decisão são comuns em projetos de software, principalmente no nível gerencial. Decisões gerenciais devem ser tomadas cuidadosamente, analisando-se todo o contexto e as conseqüências de cada possibilidade. Muitas vezes essas decisões envolvem cenários complexos, com muitas variáveis para serem analisadas, o que leva a tomadas de decisão reativas, considerando apenas o problema presente sem relacioná-lo ao seu ambiente, suas variáveis e demais problemas correlacionados [1]. Isso se torna um risco considerável no projeto, visto que decisões gerenciais equivocadas podem representar grandes prejuízos no futuro.

Na extração de requisitos os impactos das decisões são ainda maiores, visto que os requisitos formam uma base de trabalho que irá direcionar todas as iterações seguintes do desenvolvimento [1]. Um erro proveniente da especificação de requisitos detectado em uma fase avançada do desenvolvimento irá exigir retrabalho, com possibilidade de introdução de novos erros. Planos, modelos, esquemas, documentação deverão ser modificados para que a consistência e a rastreabilidade sejam mantidas [2]. Se os requisitos não forem rastreáveis [3] o problema é agravado, pois a dificuldade será maior para identificar todos os artefatos que precisam ser alterados [1].

Dada a complexidade das relações entre as variáveis dinâmicas envolvidas na fase

de requisitos do processo de desenvolvimento de software, torna-se indispensável o uso de ferramentas que permitam visualizar o contexto como um todo [1].

A dinâmica de sistemas [1, 4, 5, 6] provê apoio a este tipo de problema, através da investigação dos aspectos do processo de desenvolvimento de software em níveis micro e macro. No nível micro, ela pode ser usada para comparar o ciclo de vida de diferentes processos, técnicas de detecção de erro, casos de negócio, decidir o “quanto é o bastante” em termos de testes ou outros problemas. Já no nível macro, organizações podem focar em aspectos específicos do custo de desenvolvimento, cronograma, qualidade do produto ou outras questões [4]. Sua grande utilidade consiste em viabilizar a compreensão de como as políticas adotadas, ou a própria estrutura do sistema, afetam ou determinam o seu comportamento dinâmico [6].

Na próxima seção serão apresentados o contexto e os trabalhos correlatos que foram tomados como base para o desenvolvimento deste trabalho. Na Seção 3 será apresentado um modelo estoque-fluxo da dinâmica de sistemas que representa a atividade de elicitação. As simulações realizadas e suas análises serão discutidas na Seção 4. Por fim, serão apresentados as conclusões e trabalhos futuros na Seção 5.

## 2 Contexto e Trabalhos Correlatos

O contexto do trabalho é a utilização da dinâmica de sistemas para modelar a fase de requisitos<sup>1</sup> no processo de desenvolvimento de software, e mais especificamente a atividade de elicitação de requisitos, que é uma das atividades realizadas nessa etapa do processo de desenvolvimento.

Entre os diversos trabalhos relacionados à aplicação de dinâmica de sistemas na modelagem de processos de desenvolvimento de software, destacam-se os trabalhos de Abdel-Hamid e Madnick [7] e Madachy [4]. Ambos apresentam vários modelos de dinâmica de sistemas que abordam diferentes etapas do processo de desenvolvimento de software. Em relação à modelagem focada em requisitos de software, destacam-se os trabalhos de Ambrósio [1, 8].

Ambrósio [1, 8] propôs um modelo de dinâmica de sistemas para analisar efeitos mútuos de variáveis envolvidas na fase de requisitos do processo de desenvolvimento de software. Através desse modelo, gerentes de projeto podem buscar alternativas para avaliar o risco de suas decisões, fornecendo parâmetros que caracterizam sua equipe e executando simulações, com o intuito de verificar as possíveis consequências de suas decisões. São abordados no modelo temas como o *turnover* de pessoal, retrabalho devido a erros, falhas em planejamento, entre outros.

O modelo de Ambrósio [1, 8] é focado nas atividades de especificação e validação de requisitos, considerando-se como atividades da fase de requisitos a elicitação, análise, especificação e validação [9]. Não são retratadas no modelo as atividades anteriores à especificação. Considera-se que os requisitos do software vêm de um escopo externo.

Um complemento necessário ao modelo proposto por Ambrósio [1, 8] é a contextualização dos requisitos que alimentam o modelo nas atividades de elicitação e análise.

---

<sup>1</sup> O termo “fase de requisitos” foi utilizado para manter o contexto do trabalho de Ambrósio [1]

se. No presente trabalho, será abordada a contextualização dos requisitos na atividade de elicitação. Serão analisadas as variáveis dinâmicas que impactam nessa atividade e que devem ser levadas em consideração para a tomada de decisões nessa etapa do processo de desenvolvimento. Apesar de ser um complemento para o modelo de Ambrósio [1, 8], o modelo produzido neste trabalho é totalmente independente.

### 3 Modelagem da Atividade de Elicitação de Requisitos

O modelo de dinâmica de sistemas apresentado neste trabalho abrange as principais variáveis envolvidas na atividade de elicitação e alguns dos relacionamentos existentes entre elas. As relações entre as variáveis foram definidas e quantificadas de acordo com dados disponíveis na literatura, tomando como base os estudos de Dieste *et al.* [10, 11] e Davis *et al.* [12], além de diversos outros trabalhos citados ao longo do texto. Para construir o modelo foi utilizada a ferramenta Vensim [17], em sua versão gratuita para uso acadêmico.

Os parâmetros estudados estão relacionados à técnica de entrevista [9, 10, 11, 12, 13, 14], que é a técnica mais comumente utilizada na prática [12]. Ela consiste em diálogos entre elicidores e stakeholders sobre o domínio do problema no qual o software deve atuar. Neste trabalho foi considerada a técnica de entrevista não-estruturada, ou seja, um diálogo menos formal, onde não há um roteiro rígido a ser seguido, dando liberdade ao entrevistado de abordar os assuntos a seu próprio modo.

Para facilitar o entendimento, cada parte do modelo será apresentada separadamente, com explicações de como as variáveis foram definidas e quantificadas.

A primeira parte do modelo é apresentada na Fig. 1 e possui os três elementos principais da dinâmica de sistemas: os estoques (*Requisitos Repetidos*, *Requisitos Elicitados* e *Requisitos com Erros*), que são os recursos acumuláveis do sistema; os fluxos (*Taxa de Req. Repetidos*, *Taxa de Elicitação Correta* e *Taxa de Erros*), que são funções que representam as decisões ou políticas da empresa com relação aos usos e acúmulos dos estoques ou recursos; e as variáveis simples (demais variáveis), que são os elementos do modelo que exercem influência sobre os valores dos fluxos responsáveis pela variação dos estoques [1].

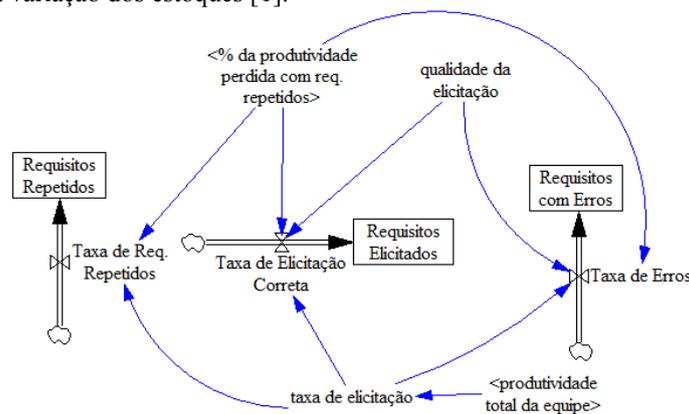


Fig. 1. Fluxo principal do modelo

Na Fig. 1, a “taxa de elicitação” depende da técnica de elicitação utilizada e da “produtividade total da equipe” alocada no projeto. Essa taxa influencia a quantidade de requisitos elicitados. Os requisitos elicitados podem ser repetidos ou novos requisitos, que ainda podem ter sido elicitados corretamente (“Taxa de Elicitação Correta”) ou não (“Taxa de Erros”), dependendo da “qualidade da elicitação”. A qualidade da elicitação, por sua vez, depende também da técnica de elicitação utilizada.

Corbridge *et al.* [18] e Burton *et al.* [19] realizaram experimentos de comparação entre técnicas de elicitação. Nesses experimentos, as sessões de elicitação foram gravadas e depois codificadas em “*Pseudo-English Production Rules* (PPRs)”. Essas regras (*rules*) são compostas por estruturas IF, AND e THEN, mas são independentes de implementação em software [19]. Regras como essas são úteis para avaliar o “ganho” da técnica, ou seja, a quantidade de conhecimento que é elicitada a partir dos *stakeholders*<sup>2</sup> utilizando uma técnica específica [18]. Porém, essa medida de ganho não reflete precisamente a quantidade de informações contidas nas regras, uma vez que uma regra pode conter muitas cláusulas [18]. Uma cláusula é definida como uma instrução condicional em uma regra de produção (PPR) [19]. Por exemplo, seja uma regra “IF a AND b AND c THEN x” e outra regra “IF d THEN y”. Apesar de ambas representarem apenas uma regra de produção, na primeira foi possível extrair mais informações do que na segunda, visto que a primeira contém mais cláusulas. Nos estudos citados foram considerados os números de cláusulas como medidas de “ganho” para a comparação entre as técnicas de elicitação.

Os estudos de Corbridge *et al.* [18] e Burton *et al.* [19, 20, 21] foram levados em consideração neste trabalho por terem sido bem avaliados em um dos trabalhos tomados como base ([10]). Como esses estudos compararam as técnicas em termos de número de cláusulas, essa também foi a medida adotada no presente trabalho.

A Tabela 1 mostra uma compilação de resultados obtidos nos experimentos no que diz respeito à técnica de entrevista. Os dados omitidos na tabela não foram divulgados nos trabalhos citados. Os campos se referem ao número de cláusulas obtidas, tempo total da elicitação em minutos, a completude dos requisitos em relação ao total (em porcentagem) para um *stakeholder* e para todos os *stakeholders* (32 ao todo). Os *stakeholders* eram estudantes “especialistas” em um domínio, e a porcentagem de completude alcançada por eles foi comparada à quantidade de cláusulas obtidas por um especialista sênior na área (considerado 100%).

**Tabela 1.** Resumo dos resultados dos experimentos de Corbridge *et al.* [18] e Burton *et al.* [19]

Medida	Exp 1 [20 apud 18]	Exp 2 [21 apud 18]	Exp 3 [19]	Exp 4 [19]	Exp 5 [18]
Cláusulas	94,4	--	270	317	274
Tempo (min)	--	--	217	240	39,5
Completude % (1)	27,9	33	--	--	--
Completude % (32)	79,5	94	--	--	--
Cláusulas/Tempo	--	--	1,2	1,3	6,9

A “taxa de elicitação” no modelo foi considerada baseando-se nas médias de cláusulas por tempo obtidas na Tabela 1. Uma observação importante é que essas médias

<sup>2</sup> No escopo deste trabalho, *stakeholders* são os profissionais entrevistados durante a elicitação.

foram baseadas a partir de *stakeholders* pouco experientes (estudantes).

Os dados divulgados na Tabela 1 pelo Exp. 5 foram muito superiores quando comparados aos divulgados nos experimentos anteriores, portanto esses resultados não foram considerados neste trabalho. Os valores elevados podem ter sido consequência do domínio do problema em questão (diagnóstico médico) e da forma como o experimento foi conduzido, que foi diferente dos experimentos anteriores.

A “qualidade da elicitação” foi baseada no trabalho de Burton *et al.* [19]. Um dos experimentos realizados nesse trabalho foi a classificação por parte dos especialistas (*stakeholders*) do conjunto de cláusulas elicitadas a partir deles mesmos pelos analistas. As cláusulas foram categorizadas em verdadeiras, falsas, triviais e parcialmente verdadeiras. As definições de verdadeiras e falsas são auto-explicativas. A definição de trivial dada foi que a cláusula não é relevante no diagnóstico de problemas no domínio em questão. Uma cláusula classificada como parcialmente verdadeira significa que se ela sofresse alguma modificação ela se tornaria verdadeira [19]. Os resultados desse experimento para a técnica de entrevista foram: 63% verdadeiras, 22% triviais, 8% parcialmente verdadeiras e 6% falsas.

Para quantificar a qualidade da elicitação foram consideradas como cláusulas elicitadas corretamente as cláusulas classificadas como verdadeiras e triviais, porém, foi dado um peso maior para as cláusulas classificadas como verdadeiras, já que são as cláusulas mais úteis no contexto do domínio.

A Fig. 2 exhibe as variáveis que impactam na produtividade.

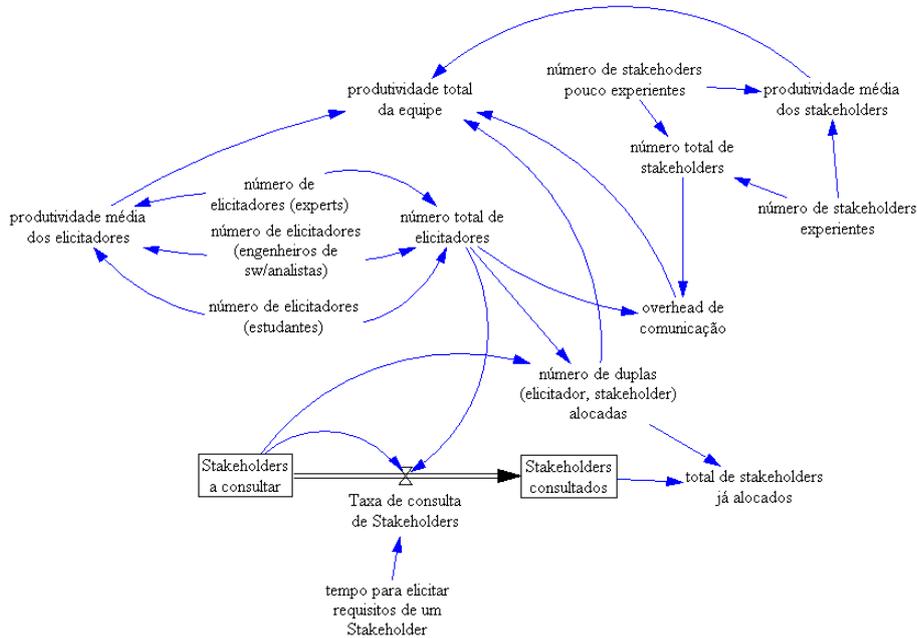


Fig. 2. Parte do modelo referente à produtividade

A relação entre a produtividade dos elicitadores<sup>3</sup> com o seu perfil (especialista, engenheiro ou estudante) foi quantificada baseando-se no experimento realizado por Griffin e Hauser [15]. Segundo eles, em média cada elicitador foi capaz de identificar 54% das necessidades dos *stakeholders*, variando-se entre 45 e 68%, sendo que os especialistas ficaram na extremidade baixa (45%) e a equipe de engenheiros ficou na extremidade alta (68%). Os estudantes ficaram na metade do intervalo (considerado 56%). Para o modelo proposto neste trabalho, foi considerada a diferença entre os valores, com as devidas adaptações para a área, obtendo que os especialistas no domínio do sistema têm 66% da produtividade dos engenheiros de software, enquanto os estudantes (estagiários) têm 82%.

A diferença entre a produtividade dos *stakeholders* experientes e pouco experientes, ou seja, com muito ou pouco conhecimento no domínio, foi baseada na Tabela 1. Nela é mostrada a completude alcançada por um estudante (menos experiente) com relação a um especialista sênior da área (muito experiente). O fato de os *stakeholders* mais experientes obterem maior produtividade do que os menos experientes pôde ser validado através do trabalho de Fowlkes *et al.* [16].

Apesar do acréscimo do número de elicitadores e *stakeholders* no projeto poder aumentar a produtividade percebida, quanto maior for o tamanho da equipe, maior será o tempo gasto com comunicação e reuniões envolvendo os membros da equipe [22], ou seja, maior será o *overhead*<sup>4</sup> de comunicação [4]. Esse *overhead* é dado em porcentagem e expresso como uma função não-linear do total de pessoas que precisam se comunicar [7].

No desenvolvimento do modelo, foi considerado que os elicitadores e *stakeholders* trabalham sempre em duplas (um elicitador com um *stakeholder*), até que todos os *stakeholders* sejam consultados. Dessa forma, o número de *stakeholders* inicial é dividido pelo número de elicitadores e essa “sessão de elicitação” tem um tempo estipulado. Passado o tempo de elicitação, os elicitadores começam uma nova sessão com os *stakeholders* que ainda não foram entrevistados.

A “produtividade total da equipe” em um dado momento pode ser expressa como uma função do número de duplas (elicitador e *stakeholder*) alocadas no momento multiplicado pelas produtividades médias dos *stakeholders* e elicitadores, descontando-se a produtividade perdida pelo *overhead* de comunicação.

Além de a produtividade do *stakeholder* estar relacionada com a sua experiência, existe uma relação não linear entre o número de *stakeholders* alocados no projeto e o percentual de necessidades identificadas [15]. Essa relação pode ser obtida através de um modelo beta-binomial [15], que permite predizer de forma aproximada qual é a produtividade percebida (através do percentual de necessidades identificadas) dado o número de pessoas entrevistadas. A relação não é linear devido ao número de requisitos repetidos elicitados pelos *stakeholders*, que aumenta na medida em que o número de *stakeholders* também aumenta. O modelo beta-binomial foi utilizado para calcular o quanto de produtividade é perdido com requisitos repetidos dado o número de *stakeholders* alocados. A parte do modelo que descreve essa relação é apresentada na Fig. 3

---

<sup>3</sup> Elicitadores são os profissionais que efetivamente realizam a elicitação, os entrevistadores.

<sup>4</sup> *Overhead* de comunicação aqui considerado como o tempo adicional gasto para os profissionais se comunicarem, que se torna maior à medida que o número de profissionais aumenta.

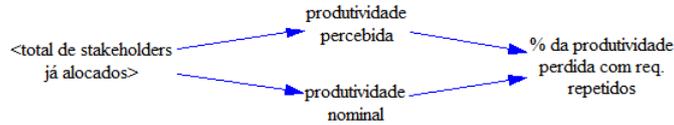


Fig.3. Parte do modelo referente aos requisitos repetidos

A porcentagem da produtividade perdida com requisitos repetidos é calculada a partir da diferença entre a produtividade nominal de um *stakeholder* e a produtividade média percebida dos *stakeholders* alocados no projeto. A produtividade nominal é a produtividade de um único *stakeholder* alocado no projeto, e a percebida é a média das produtividades dos *stakeholders* alocados, que variam de acordo com o modelo beta-binomial [15].

A quarta parte do modelo a ser analisada (Fig. 4) mostra os custos envolvidos na elicitação. Para o propósito deste trabalho, é necessário apenas que exista um parâmetro de comparação de custos entre os diversos cenários a serem simulados a partir do modelo proposto. Portanto, não é necessário que esses custos sejam representados por valores exatos.

Nessa parte do modelo, são calculados os custos médios dos profissionais na equipe dado o número de pessoas de cada perfil multiplicado pelo valor da remuneração desse perfil. Além disso, é calculado o custo médio por requisito, dividindo o custo total pelo número total de requisitos elicitados até o momento.

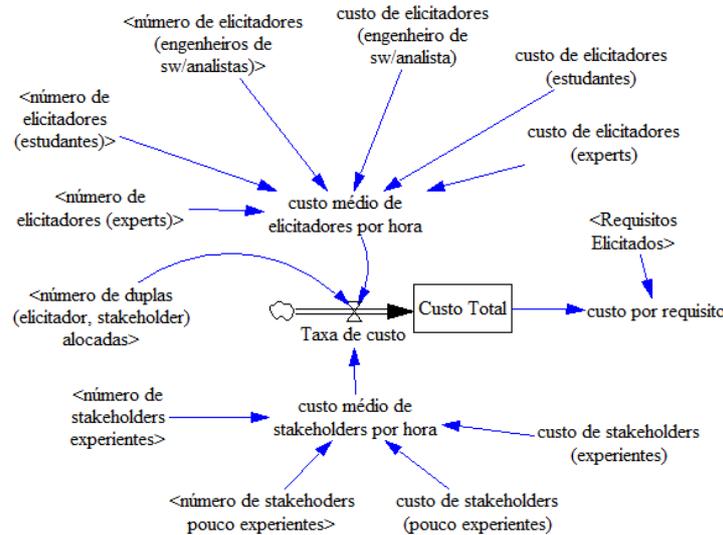


Fig. 4. Parte do modelo referente aos custos

## 4 Simulações e Análises dos Resultados

Para executar simulações no modelo apresentado na seção anterior, é necessário fornecer o número de horas a serem simuladas, o número de profissionais alocados de

cada perfil e suas respectivas remunerações. Os demais parâmetros do modelo já estão baseados em dados disponíveis na literatura, conforme mostrado na Seção 3.

Nesta seção, foram criados cinco cenários que abrangem diferentes configurações de fases iniciais de projetos de software. Esses cenários facilitam as análises e comparações entre as simulações.

Em todos os cenários simulados os custos por hora dos profissionais foram fixos em: 20,00UM (unidades monetárias) por elicitador engenheiro/analista, 5,00UM por elicitador estudante, 40,00UM por *stakeholder* experiente e 15,00UM por *stakeholder* pouco experiente. Além disso, foi considerada a simulação de 30 horas em sessões de elicitação, que serão divididas de acordo com o número de profissionais alocados.

O cenário 1 (C1) é composto por apenas um elicitador engenheiro de software/analista e um *stakeholder* experiente. Os resultados da simulação para esse cenário são exibidos na Fig. 5a.

O cenário 2 (C2) é formado por 2 elicitadores engenheiros de software/analistas e 2 *stakeholders* experientes, ou seja, o dobro de profissionais com relação a C1. Os resultados da simulação para C2 são apresentados na Fig. 5b.

No cenário 3 (C3) são considerados profissionais com menos experiência, ou seja, mão de obra mais barata. Nesse cenário, são alocados 2 elicitadores estudantes e 2 *stakeholders* pouco experientes. Os resultados da simulação são mostrados na Fig. 5c.

No cenário 4 (C4) são alocados 1 elicitador engenheiro de software/analista e 2 *stakeholders* pouco experientes. Neste caso, foram alocadas 15 horas de sessões de elicitação para cada *stakeholder*, totalizando as mesmas 30 horas em sessões de elicitação dos outros cenários. A Fig. 5d mostra os resultados da simulação.

No quinto cenário (C5) são considerados 1 elicitador estudante e 2 *stakeholders* experientes. Assim como em C4, foram alocadas 15 horas de sessões de elicitação para cada *stakeholder*, totalizando 30 horas em sessões de elicitação. Os resultados da simulação são exibidos na Fig. 5e.

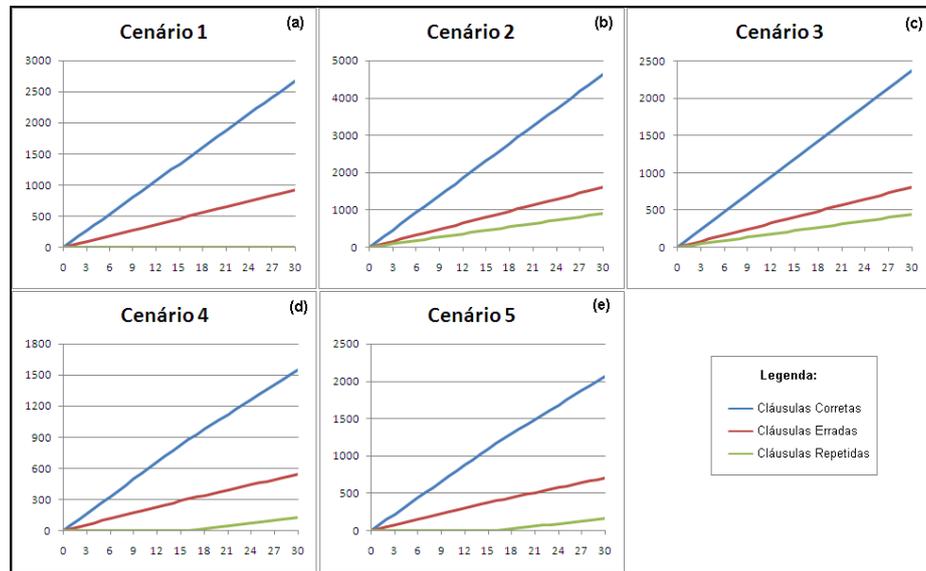
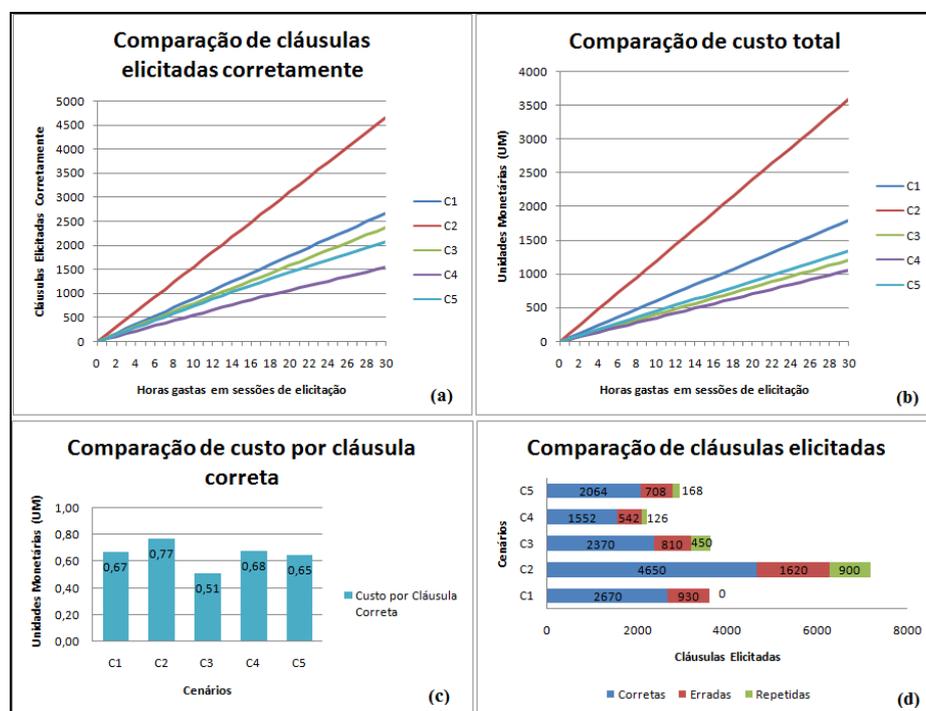


Fig. 5. Resultados das simulações dos cenários

#### 4.1 Comparação dos Cenários e Análises dos Resultados

A simulação foi executada para 30 horas de sessões de elicitação, sem considerar o número de cláusulas que o sistema necessitaria para estar completo, ou seja, foi verificado o número máximo de cláusulas obtidas por cada um dos cenários em 30 horas de elicitação.

As comparações entre os cenários estão representadas na Fig. 6. A Fig. 6a exibe uma comparação com base no número de cláusulas elicidadas corretamente. A Fig. 6b apresenta uma comparação do custo total da elicitação. A Fig. 6c mostra uma comparação quanto ao número total de cláusulas, divididas em cláusulas corretas, erradas e repetidas. Por fim, a Fig. 6d apresenta uma comparação de custo por cláusula correta, considerando o custo total da elicitação ao final das 30 horas dividido pelo número de cláusulas elicidadas corretamente no mesmo período.



Comparando os cenários C1 e C2, é possível observar que dobrando o número de profissionais na equipe (C2), o número total de cláusulas elicidadas aumentou em aproximadamente 99% (Fig. 6d), mas o número de cláusulas corretas não dobrou (aumentou em 74%). Isso ocorre devido ao aumento do número de requisitos repetidos elicitados e aumento no *overhead* de comunicação. Como consequência, o custo por cláusula em C2 teve um aumento de 15% com relação a C1 (Fig. 6c).

Entre C2 e C3, observou-se que utilizando profissionais menos experientes (C3) foi possível reduzir o custo total da elicitação em 77% (Fig. 6b), porém reduziu-se em 49% o número de cláusulas elicidadas corretamente (Fig. 6d). Comparado à C1, apesar

de C3 obter um maior número total de cláusulas, nota-se que o número de cláusulas elicitadas corretamente foi maior em C1 (Fig. 6d), mesmo utilizando metade do número de profissionais de C3. Essa superioridade de C1 é decorrente da maior produtividade dos profissionais mais experientes alocados nesse cenário.

A função do cenário C4 foi verificar se é válido substituir um *stakeholder* experiente por dois menos experientes, considerando que o custo dos menos experientes é menor. O custo total final de C4 foi 42% inferior à C1 (Fig. 6b), porém essa também foi a proporção em que se reduziu o número de cláusulas elicitadas corretamente. Essa proporção pôde ser reforçada pela proximidade entre o custo por cláusula correta entre esses dois cenários, observada através da Fig. 6c.

O cenário C5 serviu como base de comparação com C4. Em C4 foram alocados um elicitador engenheiro e dois *stakeholders* pouco experientes. Em C5 inverteu-se as experiências dos profissionais, sendo considerados um elicitador estudante e dois *stakeholders* experientes. Em C5 obteve-se um aumento de 33% no número de cláusulas elicitadas corretamente em relação a C4, o custo total teve um aumento de 29% (Fig. 6b).

A decisão de qual cenário aplicar depende de fatores que influenciam no projeto e na empresa em questão, principalmente da disponibilidade de profissionais, prazo do projeto e recursos disponíveis. O modelo em si não mostra qual a decisão gerencial que deve ser tomada, mas auxilia os gerentes a enxergarem o contexto dos cenários [23 apud 4].

Considerando por exemplo um projeto de 1000 cláusulas a serem elicitadas, se uma empresa está com o prazo curto e dispõe de recursos poderia optar por C2, concluindo a elicitación em torno de 7 horas, mas assumindo um custo total de 840,00UM. Caso a empresa tenha tempo disponível, mas opte por economizar ao máximo, C3 seria o cenário escolhido, tendo um custo total de 520,00UM (39% de redução em relação a C2), mas terminando a tarefa em 13 horas (85% a mais de tempo em relação a C2).

Caso a empresa tenha um prazo limitado para elicitación e desconheça o domínio do problema a ser abordado no projeto, deve-se evitar alocar profissionais com pouca experiência, principalmente *stakeholders*, pois, apesar da redução considerável no custo do projeto, há um risco maior, visto que algumas cláusulas deixadas de fora pelos profissionais podem representar um conhecimento essencial para o projeto.

É importante ressaltar que o modelo foi construído a partir de resultados de pesquisas e dados experimentais, logo não deve-se esperar que as simulações produzam resultados exatamente iguais aos resultados reais. O objetivo principal das simulações é auxiliar os gerentes em suas tomadas de decisões, permitindo-os comparar cenários simulados antes de tomar as decisões no “mundo real”.

## 5 Conclusões e Trabalhos Futuros

O uso de um modelo de dinâmica de sistemas que represente a atividade de elicitación de requisitos em processos de software pode auxiliar na identificação de riscos envolvidos nas tomadas de decisão em projetos de software, favorecendo a adoção de medidas preventivas.

A partir do modelo apresentado, é possível verificar, por exemplo, como a experiência dos *stakeholders* envolvidos no projeto afeta a produtividade da elicitação, o quanto influencia o número de pessoas (elicitadores e *stakeholders*) no projeto, qual a relação de custo e produtividade, etc. Conhecendo-se essas relações, é possível evitar a tomada de decisões reativas, sem levar em consideração as variáveis dinâmicas envolvidas.

As principais contribuições deste trabalho foram: propor um modelo de dinâmica de sistemas para a atividade de elicitação de requisitos utilizando a técnica de entrevista (dentre vários trabalhos investigados, não foi encontrado um modelo de dinâmica de sistemas que abordasse a atividade de elicitação de requisitos); identificar e selecionar as principais variáveis dinâmicas da atividade de elicitação e descrever os seus relacionamentos; compilar resultados de diversas pesquisas sobre o uso da técnica de entrevista na elicitação de requisitos.

Com o intuito de desenvolver um modelo que produza resultados cada vez mais próximos da realidade, os próximos passos deste trabalho são: construir um modelo de dinâmica de sistemas para a atividade de análise de requisitos; calibrar e validar os modelos com dados disponíveis na literatura e com dados obtidos pelas empresas parceiras do Laboratório de Engenharia de Software (LES) da Universidade Federal de Viçosa; e integrar o modelo do presente trabalho com o modelo produzido por Ambrósio [1, 8], tendo assim um modelo mais completo e robusto, abordando todas as atividades da fase de requisitos em processos de desenvolvimento de software.

Além disso, para facilitar o acesso das empresas ao modelo, estão sendo construídas ferramentas de apoio, que permitam que simulações e análises sejam efetuadas remotamente, via Web. Essas ferramentas de apoio estarão disponíveis através do sítio [www.les.dpi.ufv.br](http://www.les.dpi.ufv.br).

## Referências Bibliográficas

1. Ambrósio, B. G. (2008) "Modelagem da fase de requisitos em processos de desenvolvimento de software: uma abordagem utilizando dinâmica de sistemas." Dissertação (Mestrado em Ciência da Computação), Universidade Federal de Viçosa, 2008.
2. Sayão, M. e Leite, J. C. S. P. (2006) "Rastreabilidade de Requisitos". In: Revista de Informática Teórica e Aplicada. Volume 13, Número 1. p. 57-86.
3. Ramesh, B. e Jarke, M. (2001) "Towards reference Models for Requirements Traceability". In: IEEE Transactions on Software Engineering. Volume 27, Issue 1. p. 58-93.
4. Madachy, R. (2007) "Software Process Dynamics". Wiley/IEEE Computer Society Press, 2007. 591 p.
5. Braga, J. L.; Silva, C. A. B.; Wiazowski, B. A. e Avellar, S. O. C. (2004) "Modelagem com dinâmica de sistemas". In: Santos M. L. e Vieira W. Métodos Quantitativos em Economia. Viçosa, MG: UFV, 2004. P. 411-434.
6. Sterman, J. D. (2000) "Business dynamics: systems thinking and modeling for a complex world". Boston, MA: Irwin McGraw-Hill, 2000. 982 p.
7. Abdel-Hamid, T. K. e Madnick, S. E. (1991). Software Project Dynamics: an Integrated Approach. Englewood Cliffs: Prentice Hall. 264 p.
8. Ambrósio, B. G., Braga, J. L., Oliveira, A. P. (2008) "Um modelo dinâmico para análise dos impactos da rotatividade de pessoal durante a fase de requisitos". In: XXII SBES - Simpósio Brasileiro de Engenharia de Software, 2008, Campinas, SP.

9. SWEBOK (2004). Guide to the Software Engineering Body of Knowledge. 2004 Version. A project of the IEEE Computer Society Professional Practices Committee. Disponível em: <<http://www.swebok.org/>>. Acesso em: 30 de março de 2010.
10. Dieste, O. e Juristo, N. (2010) "Systematic Review and Aggregation of Empirical Studies on Elicitation Techniques". IEEE Transactions on Software Engineering, 11 Fev. 2010. IEEE computer Society Digital Library. IEEE Computer Society.
11. Dieste, O.; Lopez, M. e Ramos, F. (2008) "Updating Systematic Review about Selection of Software Requirements Elicitation Techniques". In Workshop on Engenharia de Requisitos (WER 2008), Barcelona, Espanha.
12. Davis, A., Dieste, O., Hickey, A., Juristo, N., e Moreno, A. M. (2006) "Effectiveness of Requirements Elicitation Techniques: Empirical Results derived from a Systematic Review". Proceedings of the 14th IEEE International Conference on Requirements Engineering, Minneapolis, USA. Setembro, 2006, pp. 179-188.
13. Goguen, J. A. and Linde, C. (1993) " Techniques for requirements elicitation," Proceedings of IEEE International Symposium on Requirements Engineering, pp. 152-164, 1993.
14. Dieste, O., Juristo, N. Shull, F. (2008) "Understanding the Customer: What Do We Know about Requirements Elicitation?". IEEE Software, vol. 25, p. 11-13, 2008. -> 15
15. Griffin, A and Hauser, J. R. (1993) "The voice of the customer", Marketing Science vol. 12, pp. 1-27, 1993.
16. Fowlkes, J. E., Salas, E., Baker, D.P. (2000) "The utility of event-based knowledge elicitation". Human Factors, vol. 45, pp. 24-35, 2000.
17. Vensim (2010). Vensim from Ventana Systems, Inc. Disponível em: <<http://www.vensim.com>>. Acesso em: 19 de agosto de 2010.
18. Corbridge, B., Rugg, G., major, N.P., Shadbolt, N.R., Burton, A.M. (1994) "Laddering – technique and tool use in knowledge acquisition", Knowledge Acquisition, vol. 6, pp. 315-341, 1994.
19. Burton, A.M., Shadbolt, N.R., Rugg, G., Hedgecock, A.P. (1990) "The efficacy of knowledge acquisition techniques: A comparison across domains and levels of expertise", Knowledge Acquisition, vol. 2, pp. 167-178, 1990.
20. Burton, A.M., Shadbolt, N.R., Hedgecock, A.P., Rugg, G. (1987) "A formal evaluation of knowledge elicitation techniques for expert systems: Domain 1". In D.S. Moralee, Ed., Research and Development in Expert Systems IV, pp. 136-145. Cambridge: Cambridge University Press.
21. Burton, A.M., Shadbolt, N.R., Rugg, G., Hedgecock, A.P. (1988) "Knowledge elicitation techniques in classification domains". In Y. Kodratoff, Ed., ECAI-88: Proceedings of the 8th European Conference in Artificial Intelligence, pp. 85-90. London: Pitman.
22. Simmons, D. B. Communications: a software group productivity dominator. Software Engineering Journal, v. 6, n. 6, p. 454-462, 1991.
23. Boehm, Barry W. (1981) Software Engineering Economics, Englewood Cliffs, NJ: Prentice-Hall, 1981.