

# Auxiliando nas Decisões Gerenciais de Projetos de Software com Agentes Inteligentes

Leandro L. C. De Souza, Gustavo A. L. de Campos, Mariela I. Cortés, Anderson C. P. Queiroz

Centro de Ciências e Tecnologia – Universidade Estadual do Ceará (UECE) Av. Paranjana, 1700 – Campus do Itaperi – Fortaleza – CE - Brasil.

{leocadiodesouza, andersoncpdq}@gmail.com, {gustavo, mariela}@larces.uece.br

***Abstract.** Monitoring and Control the progress and performance of project activities is critical to ensure that the project meets the target planning. This paper presents an approach for monitoring and control of software process that combines the concepts of intelligent agents and the Earned Value technique. In this sense, information about real cost and time associated with the project scope are analyzed continuously and automated in order to measure and evaluate the performance and progress of the project activities. The approach is intended to assist the project manager for the detection and correction of possible deviations from what was planned.*

***Resumo.** Monitorar e controlar o progresso e desempenho das atividades de projeto é crítico para garantir que o projeto atenda às metas estabelecidas. Este artigo apresenta uma abordagem para o monitoramento e controle de processos de software que combina a noção de agentes inteligentes e a técnica do Valor Agregado. Para isso, informações relativas a custo e tempo associadas ao escopo do projeto, são analisadas de forma contínua e automatizada, visando medir e avaliar o desempenho e progresso do projeto. A abordagem pretende auxiliar o gerente do projeto na detecção e correção de possíveis desvios em relação ao que foi planejado.*

## 1. Introdução

Apesar da evolução nas técnicas de construção e gerência de sistemas baseados em computador, um dos problemas mais comuns ainda nos dias de hoje é entregar software de qualidade no prazo e dentro do orçamento [Pressman 2009]. Neste cenário, os processos responsáveis pelo monitoramento e controle da execução do projeto [PMI 2008.a] se tornam fundamentais para o atendimento das restrições do projeto.

O Gerenciamento do Valor Agregado (GVA) [PMI 2005.b] surge como um método amplamente utilizado para monitorar e controlar o desempenho das atividades de projeto, integrando na sua formulação, as medidas de escopo, custo e tempo. A partir dos dados obtidos, a opinião de especialistas auxilia a equipe de gerenciamento do projeto na interpretação dos dados e na proposta de ações necessárias para assegurar que o desempenho do projeto atinja as expectativas.

Os processos de monitoramento e controle incorporam certo grau de complexidade uma vez que envolvem a análise integrada de diversos aspectos do projeto, em um ambiente dinâmico e em constante mudança. Assim, a utilização de métodos e técnicas de apoio à medição e monitoramento contínuo e em tempo real de

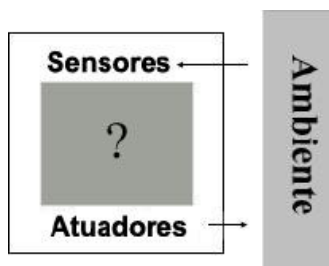
informações gerenciais é crucial para a detecção e reparo precoce de eventuais desvios. Nestas circunstâncias, características como autonomia e proatividade, fazem com que a tecnologia de agentes de software se torne uma solução adequada para o desenvolvimento deste tipo de sistemas [Jennings 2001]. Um agente inteligente é um sistema autônomo capaz de perceber as alterações do ambiente e agir sobre ele ao longo de um período de tempo [Russel e Norvig 2004]. Sua percepção pode ocorrer através da recepção de estímulos do seu meio ambiente, razão da combinação de informações recém-adquiridas com suas metas e conhecimentos já existentes, e agem através da seleção e execução de uma ação apropriada [Franklin e Graesser 1996].

O presente trabalho apresenta uma abordagem baseada na tecnologia de agentes para a automatização da aplicação do GVA cujo objetivo é observar e mensurar de forma periódica e proativa o desempenho do projeto, incorporando um grau de automação ao processo e reduzindo eventuais problemas decorrentes da dependência de interações humanas [Sethuraman et al. 2008]. Este artigo é estruturado como segue. A Seção 2 apresenta um referencial teórico sobre as técnicas e modelos utilizados na proposta. A Seção 3 aborda os trabalhos relacionados. A Seção 4 apresenta uma descrição da abordagem. Na Seção 5, o agente de monitoramento é apresentado em um nível mais técnico, onde a formulação da proposta é melhor discutida. A Seção 6 apresenta uma aplicação do agente proposto com alguns resultados gerados através de simulações. Finalmente, as considerações finais e trabalhos futuros são apresentados na Seção 7.

## 2. Fundamentação Teórica

### 2.1. Agentes Inteligentes

Um agente inteligente é uma entidade autônoma que percebe seu ambiente através de sensores e age sobre o mesmo utilizando atuadores [Russel e Norvig 2004]. O comportamento autônomo do agente implica na habilidade de executar sua tarefa por iniciativa própria, sem supervisão ou controle externo, detendo o controle sobre suas ações e estado interno [Gonçalves 2009]. A Figura 1 representa esquematicamente a interação entre o agente e seu ambiente.



**Figura 1. Agente com seus sensores e atuadores interagindo com o ambiente [Gonçalves 2009].**

A partir da percepção do seu ambiente, os agentes respondem em tempo hábil às mudanças que nele acontecem. Adicionalmente, os agentes alcançam seus objetivos de projeto através do monitoramento proativo e periódico do seu ambiente. Outros atributos em agentes incluem: comportamento colaborativo, capacidade de inferência, continuidade temporal, adaptabilidade, mobilidade, entre outros. Estas propriedades nos agentes possibilitam a geração de soluções para a crescente complexidade dos sistemas

de computação que geralmente devem operar em ambientes não predizíveis, abertos e que mudam rapidamente [Jennings 2001].

Os agentes podem ser classificados, segundo [Russel e Norvig 2004], de acordo com a maneira através da qual coletam informações e agem no ambiente. Mais especificamente, essa classificação representa a arquitetura dos agentes e são conhecidos como: (i) agentes reativos simples, (ii) agentes reativos baseados em conhecimento, (iii) agentes baseados em objetivo e (iv) agentes baseados em utilidade. Cada uma dessas arquiteturas contém internamente, uma função específica, que determina as ações dos agentes.

## **2.2. Monitoramento e Controle de Projetos**

Os processos de monitoramento e controle são responsáveis por acompanhar, revisar e regular o progresso e o desempenho do projeto, identificar todas as áreas nas quais serão necessárias mudanças no plano de execução do projeto e recomendar a aplicação das mudanças [PMI 2008.a]. Objetivam observar e mensurar o desempenho do projeto de forma periódica e uniforme para identificar variações em relação aos objetivos traçados no plano de gerenciamento. O trabalho do projeto é monitorado e controlado a partir do acompanhamento das áreas-chave: escopo, custo e tempo.

O projeto é mantido sob controle a partir da determinação de ações corretivas ou preventivas, ou o replanejamento com objetivo de resolver questões de desempenho em relação aos eventuais desvios detectados no monitoramento. Somente solicitações de mudanças aprovadas são incorporadas à linha de base revisada, o que pode requerer a elaboração de novas ou revisadas estimativas de custos e cronograma.

O monitoramento é definido como uma análise contínua da aderência do projeto aos seus planos, realizada em intervalos predeterminados. Os custos, o esforço empreendido, o seguimento do cronograma, os produtos do trabalho e os recursos utilizados até o momento são examinados continuamente em comparação ao que foi planejado [Swebok 2004].

## **2.3. Gerenciamento do Valor Agregado (GVA)**

O Gerenciamento do Valor Agregado (GVA), inserido na indústria por engenheiros de fábricas norte-americanas no início do Século XX, consiste de um método conhecido mundialmente para o monitoramento de custos e cronograma na gestão de projetos. Atualmente, o GVA é o principal método de desempenho e análise da execução do cronograma e despesas geradas no gerenciamento da construção de projetos [Chang et al. 2006].

O GVA é uma técnica muito utilizada, que compara as informações planejadas e realizadas de escopo, tempo e custo para medir o desempenho das atividades de projeto e, a partir dos indicadores obtidos, estabelecer uma tendência até seu término [PMI 2005.b]. Essa técnica requer das informações de uma linha de base integrada (planejamento atualizado e aprovado para todos os itens no projeto) a partir da qual o desempenho possa ser comparado durante o desenvolvimento das atividades do projeto.

Como gerentes de projetos se concentram mais em orçamento e cronograma, o GVA fornece uma maneira simples de resolver a questão do controle do tempo e custo separadamente. Dessa forma, é possível medir a quantidade de trabalho realizado, a

previsão de custo e a data de conclusão de um projeto, o quão bem o projeto está sendo realizado em relação ao seu plano inicial [Fleming e Koppelman 2006].

O valor agregado (VA) é frequentemente utilizado para descrever a porcentagem completa de um projeto, em termos de orçamento aprovado atribuído ao trabalho terminado e entregue. O VA é utilizado para refletir a situação corrente e determinar as tendências de desempenho em longo prazo [PMI 2005.b].

### **3. Trabalhos Relacionados**

Vários trabalhos propõem a utilização de agentes inteligentes para o gerenciamento dos diferentes aspectos no desenvolvimento de projetos. Nesta seção são apresentados trabalhos de pesquisa que utilizam agentes como abordagem para gerenciar o monitoramento e controle do andamento de projetos.

Em [Wu et al. 2009] é apresentada uma ferramenta baseada em agentes, o *Software Project Associate (SPPA)*, que consiste em um sistema multi-agente baseado em métricas do projeto, tais como produtividade e esforço. O objetivo da ferramenta é acompanhar o andamento do projeto de software para garantir a conformidade com o planejamento de metas para a realização das atividades. O SPPA alerta os gerentes quando as metas não são atingidas, desta forma o gerente somente poderá adotar ações corretivas. Nenhuma solução para a correção dos desvios é sugerida pela ferramenta. Adicionalmente, as métricas são elaboradas com base nas dimensões de escopo e tempo, sem considerar o custo nas formulações.

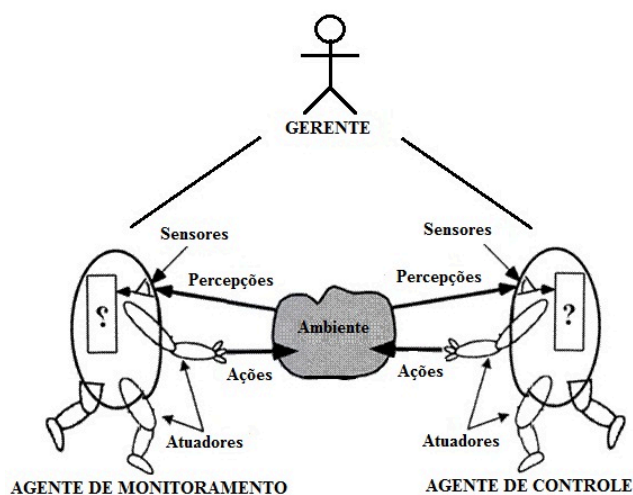
O modelo *Software Project Management supported by Software Agents (SPMSA)* [Nienaber 2008] consiste em um framework genérico, baseado em agentes de software, projetado para suportar vários aspectos do gerenciamento do projeto de software em um ambiente distribuído. A pesquisa propõe uma abordagem *black box*, onde cada componente envolve agentes cooperativos, cuja implementação a princípio, não é fornecida. O SPMSA prevê na sua estrutura um agente de monitoramento, porém se limita a acompanhamento de tarefas e fases do projeto, sem levar em consideração o aspecto de custo. É previsto a notificação dos *stakeholders*, no entanto nenhuma proposta para correção de desvios em relação à linha de base do projeto é apresentada.

Por último, o *Parametric Project Monitoring and Control (PPMC)* consiste de um modelo de estimativa proprietário fundamentado na técnica do GVA [Galorath 2006], cujo objetivo é estender o escopo da estimativa de projetos de desenvolvimento de software estabelecendo uma iteração na condução das atividades de gerenciamento. Esse modelo trabalha apenas com ações preventivas e com isso, não garantem que todos os desvios possam ser inibidos. O PPMC não é baseado em agentes, e, portanto não funciona de forma proativa, mas somente quando requerido pelo gerente ou membro da equipe. A ferramenta prevê a elaboração de relatórios e gráficos, porém não alerta sobre desvios e não propõe ações corretivas.

### **4. Abordagem Proposta**

A abordagem proposta considera o desenvolvimento de dois agentes inteligentes, os quais colaboram entre si, e com o projetista para atingir os objetivos de monitorar e controlar o trabalho do projeto, como ilustrado na Figura 2. A utilização da tecnologia de agentes é justificada pela sua capacidade de detectar mudanças no ambiente no qual

estão inseridos, raciocinar sobre essas mudanças e agir de forma proativa, selecionando uma ação e executando tarefas de forma a reduzir os efeitos negativos de possíveis desvios detectados. Para esta abordagem em específico, os agentes têm, como um objetivo particular, perceber o ambiente no qual estão inseridos e agir nesse ambiente através de mensagens de alerta, em relação a desvios (custo e tempo) que podem ocorrer durante o processo de desenvolvimento.



**Figura 2. Relação entre os agentes e o projetista. Adaptado de [Russel e Norvig 2004].**

O compartilhamento e troca de informações entre os agentes e o projetista consiste de uma atividade dinâmica e constante que envolve os (i) dados relativos ao planejamento inicial e trabalho real, relatado pelo projetista, (ii) análise da situação corrente em relação à desejada, diagnosticada pelo agente monitor, e (iii) o tratamento de desvios, realizado pelo agente de controle.

O agente monitor incorpora um conjunto de regras condição-ação baseadas na teoria do valor agregado, implementadas para determinar, em tempo real, o quão à frente ou atrás do cronograma e/ou orçamento o projeto se encontra. A partir dos indicadores obtidos, o agente alerta o gerente sobre eventuais desvios detectados. A partir das informações advindas do monitor, o agente de controle reage de acordo com o grau da variação entre o planejamento e o desempenho, sugerindo ações corretivas e preventivas com vistas a minimizar o efeito negativo dos desvios detectados. Tais ações podem envolver a compressão do cronograma e/ou replanejamento das atividades do projeto.

Os agentes de software monitoram o desempenho das principais variáveis do projeto (tempo e custo) e controlam os desvios utilizando-se de ações corretivas. O monitoramento e controle é realizado a partir do plano consolidado do projeto (linha de base). Durante a execução das atividades, o projetista intervém no processo de monitoramento e controle de modo a abastecer o sistema com as informações de desempenho do projeto. No presente artigo é apresentado o agente reativo baseado em regras condição-ação responsável pelo monitoramento das atividades de projeto, detalhado na Seção 5.

## 5. Agente de Monitoramento (AMon)

### 5.1. Monitoramento do Projeto

O funcionamento do agente AMon pressupõe a existência de um plano a partir do qual a data de início do projeto é estabelecida, e o tempo total do projeto (*timeProject*) e seu custo total (*budgetProject*) são estimados. Além destas duas variáveis e do momento atual em que o projeto se encontra (*K*), outras variáveis consideradas na teoria do valor agregado são utilizadas no processo de monitoramento. Os valores destas variáveis de estado do projeto são calculados e atualizados periodicamente dependendo do andamento do projeto, são elas: porcentagem do trabalho desejado até o momento (*planejadaCompleta*), porcentagem do trabalho efetivamente realizado até o momento (*realCompleta*) e a despesa gerada até o momento (*custoReal*). A estrutura do agente AMon é apresentada na Figura 3.

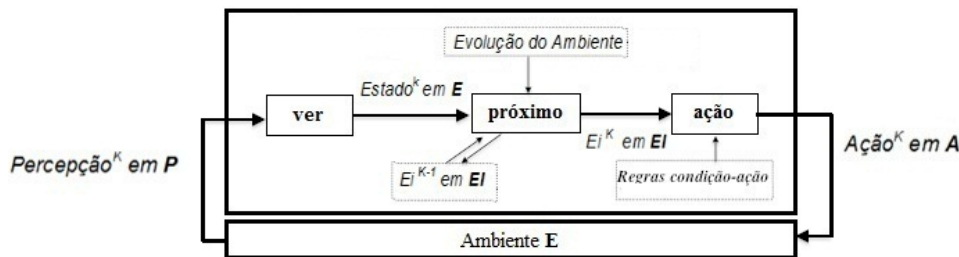


Figura 3. Arquitetura do agente AMon. Adaptado de [Russel e Norvig 2004].

A função *ver* representa os sensores do agente. Em qualquer interação do agente com o ambiente *K*, em qualquer momento do projeto *T*, esta função percebe as informações a respeito dos valores das variáveis de estado do projeto (*Percepção<sup>K</sup>*) e mapeia estas informações em uma representação computacional adequada (*Estado<sup>K</sup>*), que, no caso do AMon, é uma sêxtupla da forma: (*T*, *timeProject*, *budgetProject*, *planejadaCompleta*, *realCompleta*, *custoReal*). Considerando esta informação estruturada e outras informações que o agente mantém internamente, a respeito de como o ambiente de desenvolvimento do projeto evolui, a função *próximo* atualiza o estado interno que o agente mantém a respeito do projeto (*Ei<sup>K</sup>*). Mais especificamente, esta atualização ocorre através dos cálculos dos valores correntes das seguintes variáveis contempladas no plano inicial fundamentado em GVA onde VP representa o valor planejado, VA o valor agregado, IDP o índice de desempenho de prazo, VDP a variação de prazo, IDC o índice de desempenho de custo e VC a variação de custo:

$$VP = budgetProject \times planejadaCompleta \quad eq.01$$

$$VA = budgetProject \times realCompleta \quad eq.02$$

$$IDP = VA / VP \quad eq.03$$

$$VDP = VA - VP \quad eq.04$$

$$IDC = VA / custoReal \quad eq.05$$

$$VC = VA - custoReal \quad eq.06$$

Os valores das variáveis (01)–(06), componentes do estado interno do agente representam uma sêxtupla da forma: (VP, VA, IDP, VDP, IDC, VC), indicando se existem diferenças entre o estado corrente do projeto no momento *K* e seu estado desejado. Mais especificamente, estes valores permitem que o agente determine a

situação real dos trabalhos de projeto quanto a orçamento e cronograma. Assim, em qualquer momento  $K$  do projeto, a função *ação* do agente considera estes valores e um conjunto de regras condição-ação, e seleciona como ações ( $A^k$ ) uma ou mais mensagens indicando ao gerente a situação atual do projeto quanto ao cronograma e orçamento. A Tabela 1 apresenta um subconjunto de doze regras para o monitoramento envolvendo as variáveis (03)-(06).

As regras (01)-(03) permitem alertar o gerente sobre a situação atual do projeto em relação ao cronograma. As três situações de alertas possíveis são informadas através do cálculo da variável (IDP) julgada na eq. 03.

**Tabela 1. Subconjunto de 12 regras condição-ação embutidas no AMon.**

Id	Regras Condição-Ação
...	...
1 2 3	se $IDP = 1.0$ então faça(mensagem('Projeto está dentro do cronograma')) senão se $IDP > 1$ então faça(mensagem('Projeto está adiantado do cronograma')) senão faça(mensagem('Projeto está atrasado no cronograma'))
4 5 6	se $VDP = 0.0$ então faça(mensagem('Projeto está dentro do cronograma')) senão se $VDP > 0$ então faça(mensagem('Projeto está VDP reais a frente do cronograma')) senão faça(mensagem('Projeto está VDP reais atrás do cronograma'))
7 8 9	se $IDC = 1.0$ então faça(mensagem('Projeto está dentro do orçamento')) senão se $IDC > 1$ então faça(mensagem('Projeto está abaixo do orçamento')) senão faça(mensagem('Projeto está acima do orçamento'))
10 11 12	se $VC = 0.0$ então faça(mensagem('Projeto está dentro do orçamento')) senão se $VC > 0$ então faça(mensagem('Projeto está VC reais abaixo do orçamento')) senão faça(mensagem('Projeto está VC reais acima do orçamento'))
...	...

As regras (04)-(06), por sua vez, permitem o agente informar o quanto em dinheiro, o projeto está à frente ou atrás do cronograma. Essas regras têm como base a eq. 04. As regras (07)-(09) compreendem informações a respeito do custo do projeto, informando a situação real quanto ao orçamento. Essas informações são processadas através do cálculo de IDC, julgado na eq. 05. Por fim, as regras (10)-(12) permitem informar ao gerente a variação, em dinheiro, positiva ou negativa do orçamento. Para isso, o VC é calculado na eq. 06. A Figura 4 sintetiza o esqueleto do programa agente AMon.

```

função agente-AMon(PercepçãoK) retorna uma ação
entradas: Percepção em P, percepção dos valores, uma descrição dos valores percepção
variáveis estáticas: EstadoK em E, uma sêxtupla da forma (T, timeProject, budgetProject,
planejadaCompleta, realCompleta, custoReal)
INFOevolução, informação nas Expressões (01)-(06)
EiK em EI, uma sêxtupla da forma (VP, VA, IDP, VDP, IDC, VC)
REGRAS, conjunto de doze regras na Tabela 1
AçãoK em A, quatro mensagens nos consequentes de quatro regras

EstadoK ← ver(PercepçãoK)
EiK ← próximo(EstadoK, EiK-1, INFOevolução)
AçãoK ← ação(EiK, REGRAS)
retornar AçãoK

```

**Figura 4. Esqueleto do programa agente AMon.**

O mecanismo de seleção de ação do AMon foi concebido para indicar como solução do problema de monitoramento em uma interação  $K$ , todas as mensagens que aparecem nos consequentes das regras cujos antecedentes descrevem condições avaliadas como verdadeiras no momento  $K$ . Com doze regras, quatro mensagens serão ativadas em cada interação que o agente mantiver com o ambiente de projeto.

## 6. Simulação do Funcionamento do Agente AMon

Conforme a Seção 5 ilustra, a concepção do agente AMon considera que o agente interage com o ambiente de projeto periodicamente, buscando perceber os valores de seis variáveis de estado do projeto e, sempre que necessário, enviar para o projetista mensagens de alerta indicando diferenças positivas ou negativas entre os estados corrente e desejado do projeto. Esta seção apresenta os resultados das interações do agente AMon com um ambiente de projeto simulado, mas que pode ser configurado para gerar diferentes cenários de satisfação orçamento/escopo do projeto ao final de seu tempo estimado.

No modelo do ambiente concebido, o tempo estimado para o término do projeto e seu custo total, do ponto de vista do modelo, representados por TP e BP, são inicializados e mantêm-se constantes até o final das simulações. A porcentagem desejada do trabalho planejado, PC, a porcentagem realizada do trabalho planejado, RC, e os gastos realizados, CR, também devem ser inicializados, mas variam com o tempo, T. Mais especificamente, o modelo incorpora os valores iniciais na definição dos comportamentos das variáveis PC, RC e CR em função do tempo:

$$PC(T) = ((1 - PC1) * (T/TP) + PC1) \quad \text{eq.07}$$

$$RC(T) = (1 - \delta RC - RC1) * (T/TP) + RC1 \quad \text{eq.08}$$

$$CR(T) = (BP - \delta CR - CR1) * (T/TP) + CR1 \quad \text{eq.09}$$

onde T representa o tempo atual do projeto, PC1 e RC1 representam respectivamente as porcentagens desejada e realizada do trabalho planejado no início das simulações; CR1 representa o custo real do projeto no início;  $\delta RC$  representa a porcentagem não realizada do trabalho planejado e  $\delta CR$  o valor acima/abaixo do orçamento gasto/economizado ao final do tempo do projeto, tal que as duas condições abaixo devem ser satisfeitas:

$$0 \leq \delta RC < 1 - RC1 \quad \text{eq.10}$$

$$CR1 - BP < \delta CR \leq \%BP \quad \text{eq.11}$$

Considerando as condições descritas em eq.10 e eq.11 para atribuição de valores aos parâmetros do modelo,  $\delta RC$  e  $\delta CR$ , foram configurados seis cenários diferentes para simulação das interações, K, do AMon com o ambiente do projeto. Cada cenário é descrito em termos do nível de satisfação alcançado no trabalho realizado e em seu custo ao final do tempo estimado para duração do projeto (TP). A Tabela 2 descreve os cenários simulados, considerando três alternativas possíveis de níveis de satisfação escopo/custo ao final das simulações, ou seja: insatisfação (-), satisfação (•) e mais que satisfação (+). A Tabela 2 também apresenta as regras que foram ativadas pelo agente nas interações que manteve com o ambiente de projeto devidamente parametrizado. Nos três primeiros cenários as atividades do projeto realizaram 100% do trabalho planejado ao final do tempo estimado. No Cenário A o trabalho realizado consumiu exatamente o que tinha sido previsto. No Cenário B o projeto conseguiu economizar 30% do



orçamento. No Cenário C, ultrapassou o orçamento em 30%. Nos outros três cenários, entretanto o trabalho planejado não foi concretizado. No Cenário D exatamente 20% deixou de ser realizado. No Cenário E, 20% não foi finalizado. No Cenário F, ao todo, 30% deixou de ser entregue no tempo previsto.

**Tabela 2. Cenários de Interação AMon/Ambiente.**

Cenário	$\delta_{RC}$	$\delta_{CR}$	Nível de Satisfação		Regras Ativadas
			Escopo	Custo	
A	0	0	•	•	r1, r2, r4, r5, r7, r8, r10, r11
B	0	-0.3*BP	•	+	r1, r2, r4, r5, r7, r8, r10, r11
C	0	0.3*BP	•	-	r1, r2, r4, r5, r7, r9, r10, r12
D	0.1	0	-	-	r1, r2, r3, r4, r5, r6, r7, r9, r10, r12
E	0.2	-0.3*BP	-	+	r2, r3, r5, r6, r7, r8, r10, r11
F	0.3	0.3*BP	-	-	r2, r3, r5, r6, r7, r9, r10, r12

No que diz respeito à parametrização necessária à simulação das interações do agente com o ambiente, para a avaliação das regras ativadas por AMon na Tabela 2, foi suficiente realizar seis interações ( $K = 1, \dots, 6$ ) considerando o tempo estimado para a finalização do projeto igual a trinta unidades de tempo,  $TP = 30$  e  $T = 0..30$ , e os seguintes valores iniciais expressos em (7)-(9):  $BP = R\$ 20.000,00$ ;  $PC1 = 30\%$ ,  $RC1 = 45\%$ ,  $CR1 = R\$ 9.000,00$ . A Tabela 3 apresenta o *log* das principais informações processadas pelo agente AMon na primeira ( $K = 1$ ) e na última interação ( $K = 6$ ) do agente com o ambiente configurado no Cenário B.

**Tabela 3. Saídas de *ver*, *próximo* e *ação* do AMon em seis interações no Cenário B.**

K	Percepção <sup>K</sup>				Ei <sup>K</sup>				Re <sup>K</sup>	Ação <sup>K</sup>
	T	PC	RC	CR	IDP	VDP	IDC	VC		
1	0	0.3	0.45	9000	1.5	3000	1.0	0	2 5 7 10	Cronograma Adiantado. R\$ 3000 Frente Cronograma. Dentro Orçamento. Dentro Orçamento.
...	...	...	...	...	...	...	...	...	...	...
6	30	1.0	1.0	14000	1.0	0.0	1.42	6000	1 4 8 11	Dentro Cronograma. Dentro Cronograma. Abaixo Orçamento. R\$ 6000 Abaixo Orçamento.

No Cenário B, onde ocorre satisfação quanto a escopo e custo ao final do projeto, foram ativadas oito das doze regras mantidas internamente por AMon. Na primeira interação com o ambiente ( $K = 1$ ), o agente percebeu os parâmetros que foram

inicializados no modelo do ambiente, atualizou seu estado interno e ativou o subconjunto de regras correspondentes, ou seja, as regras 2, 5, 7 e 10, cujos consequentes contém as mensagens de mais que satisfação escopo/custo, que foram enviadas para o ambiente. Na última interação ( $K = 6$ ) o agente agiu corretamente enviando as mensagens de satisfação nos consequentes das regras 1, 4, 8 e 11. A Figura 5 apresenta um gráfico com as curvas de variação entre o planejado e o desempenho real. Como pode ser observado, o custo do projeto teve uma redução ao longo do tempo.

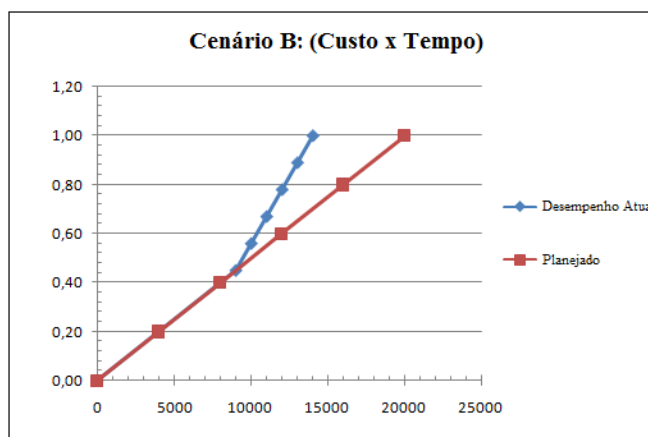


Figura 5. Gráfico do Cenário B (Custo x Tempo).

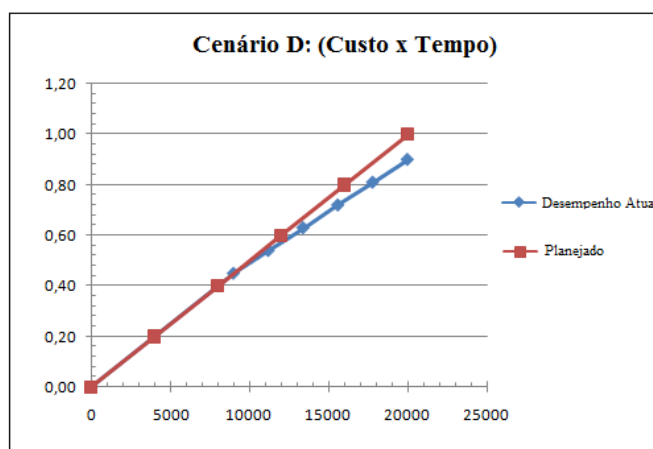
A Tabela 4 apresenta o *log* das principais informações processadas pelo agente em três interações que manteve com o ambiente configurado no Cenário D. Para este cenário, considera insatisfação em 10% no escopo e insatisfação no custo ao final do projeto. Como era esperado nos seis cenários, a primeira interação com o ambiente no Cenário D ocorreu da mesma maneira que no Cenário B. Entretanto, na quinta ( $K = 5$ ) e na última interação, o agente se comportou de maneira diferente do ocorrido no Cenário B. Na quinta interação o agente ativou as regras 3, 6, 9 e 12, enviando mensagens indicando insatisfação quanto a escopo e insatisfação quanto a custo. Na sexta, considerando as regras ativadas 3, 6, 9 e 12, que foram as únicas regras não ativadas no Cenário B, o agente continuou enviando mensagens de insatisfação quanto a escopo e insatisfação quanto a custo.

Tabela 4. Saídas de *ver*, *próximo* e *ação* do AMon em seis interações no Cenário D.

$K$	$Percepção^K$				$Et^K$				$Re^K$	$Ação^K$
	T	PC	RC	CR	IDP	VDP	IDC	VC		
1	0	0.3	0.45	9000	1.5	3000	1.0	0	2 5 7 10	Cronograma Adiantado. R\$ 3000 Frente Cronograma. Dentro Orçamento. Dentro Orçamento.
...	...	...	...	...	...	...	...	...	...	...
5	24	0.86	0.81	17800	0.94	-999.9	0.91	-1599	3 6 9	Cronograma Atrasado. R\$ 999.9 Atrás Cronograma. Acima Orçamento.

									12	R\$ 1599 Acima Orçamento.
6	30	1.0	0.9	20000	0.9	-2000	0.9	-2000	3	Cronograma Atrasado.
									6	R\$ 2000 Atrás Cronograma.
									9	Acima Orçamento.
									12	R\$ 2000 Acima Orçamento.

Assim como no Cenário B, a Figura 6 apresenta as curvas de variação entre o planejado e o desempenho real do Cenário D. Agora, o projeto encontra-se atrasado no cronograma e acima do orçamento planejado. De acordo com as curvas, o desempenho está abaixo do ideal.



**Figura 6. Gráfico do Cenário D (Custo x Tempo).**

Os Cenários B e D ilustram duas situações em que cada uma das doze regras do AMon foi ativada em uma das seis interações registradas. Conforme pode ser observado nas Tabelas 3 e 4, o agente respondeu adequadamente às percepções capturadas do ambiente nas diversas interações que realizou. As mensagens enviadas ao seu ambiente condizem integralmente com a realidade do trabalho do projeto. Da mesma maneira, realizando-se uma análise das regras que foram ativadas nos outros seis cenários descritos na Tabela 2, é possível observar que o agente também respondeu adequadamente. De acordo com os resultados apresentados, o agente AMon apresenta um comportamento racional, enviando mensagens coerentes de acordo com os estados que percebe.

## 7. Considerações Finais e Trabalhos Futuros

Este artigo apresenta uma proposta para a assistência inteligente no gerenciamento de projetos de software. A abordagem integra agentes inteligentes cujo comportamento é baseado na teoria do valor agregado, para auxiliar os gerentes e suas equipes no processo de monitoramento e controle de custo e tempo do trabalho do projeto. Considerando a complexidade dos processos e suas interações nem sempre aparentes, assim como também o volume de informação envolvida, a abordagem proativa para o monitoramento contínuo se apresenta promissora.

O agente de monitoramento apresentado foi desenvolvido utilizando a plataforma JADE (*Java Agent DEvelopment Framework*), e uma simulação foi

realizada abordando diferentes cenários para demonstrar diversas situações possíveis de um projeto de software. Baseado nos resultados apresentados observou-se que os agentes inteligentes e a técnica do valor agregado podem fornecer informações relevantes dentro do contexto de controle de processo de desenvolvimento de software.

O mecanismo de monitoramento do AMon está sendo refinado através de um conjunto de regras capaz de perceber aspectos menos aparentes do projeto. Considerando que a base de regras será estendida, está sendo concebido um modelo de detecção de conflitos de maneira a evitar a emissão de mensagens contraditórias.

Como trabalho futuro é previsto o desenvolvimento de um protótipo JADE do agente de controle e, a seguir, a comunicação de ambos os agentes com vistas à sua integração com uma ferramenta de gerência de projetos real.

## Referências

- Chang, Q., Ji, G e Li, C. (2006) “The research to second-level earned value method for improve performance measurement of engineering project”, Chinese Journal of Management Science, vol. 14, no. 2, pp. 65-70.
- Fleming, Q. W. e Koppelman, J. M. “Earned value project management”, 3rd ed., Project Management Institute, Atlanta, 2006.
- Franklin, S. e Graesser, A. (1996) “Is it an Agent, or just a program? A taxonomy for Autonomous Agents”, Em: Proceedings of the Third International workshop on Agent Theories, Architectures and Languages. Berlin: Springer- Verlag. 2135.
- Galorath, D. D. e Galorath, J. (2006) “Achieving software development success - using best practice planning, estimation, tracking and control”, Em: Software Measurement European Forum, Roma. Proceedings. Roma: SMEF, p. 293-304.
- Gonçalves, E. J. T. (2009). Modelagem de Arquiteturas Internas de Agentes de Software Utilizando A Linguagem MAS-ML 2.0. Dissertação de Mestrado em Ciência da Computação da UECE, Centro de Ciência e Tecnologia. Fortaleza.
- Jennings, N. R. (2001) “An Agent-Based approach for building Complex Software Systems”, Em: Communications of the ACM, 44, pp. 35-39.
- Nienaber, R. C. (2008) “A model for enhancing software project management using software agent technology”, Em: Universidade da África do Sul. Trabalho apresentado para o título de Doutor do Departamento de Ciência da Computação.
- Pressman, R. (2009). Engenharia de Software. 6 Ed. São Paulo: McGraw-Hill Brasil.
- Project Management Institute (PMI) (a) (2008) Um Guia do Conhecimento em Gerenciamento de Projetos (Guia PMBoK). Em: 14 Campus Boulevard Newtown Square, Pennsylvania 19073-3299, Estados Unidos, 4º Edição.
- Project Management Institute (PMI) (b) (2005) “Practice Standart for Earned Value Management”, 1º Edição. USA: Project Management Institute, Inc..
- Russel, S. e Norvig, P. (2004) “Artificial Intelligent: A modern approach”, Nova Jersey, Estados Unidos, 2º Edição.

Sethuraman, A., Yalla, K. K., Sarin, A. e Gorthi, R. P. (2008) “Agents assisted software project management”, Em: Proceedings of the 1<sup>st</sup> Bangalore Annual Compute Conference, ACM New York, USA.

Swebok, Guide to (2004). Disponível em: <http://www.swebok.org/>.

Wu, C., Chang, W. e Sethi, I. (2009) “A Metric-Based Multi-Agent System for Software Project Management,” Eight IEEE/ACIS International Conference on Computer and Information Science.