VANDERSON BOTELHO DA SILVA

UM MODELO DE CONFIANÇA CERTIFICADO BASEADO EM ASSINATURA DIGITAL APLICADO A SISTEMAS MULTIAGENTE

Dissertação apresentada ao Programa de Pós-Graduação em Informática da Pontifícia Universidade Católica do Paraná como requisito parcial para obtenção do título de Mestre em Informática.

CURITIBA
Setembro/2009

VANDERSON BOTELHO DA SILVA

UM MODELO DE CONFIANÇA CERTIFICADO BASEADO EM ASSINATURA DIGITAL APLICADO A SISTEMAS MULTIAGENTE

Dissertação apresentada ao Programa de Pós-Graduação em Informática da Pontifícia Universidade Católica do Paraná como requisito parcial para obtenção do título de Mestre em Informática.

Área de Concentração: *Agentes de Software*

Orientador: Prof. Dr. Edson Emílio Scalabrin

CURITIBA

Setembro/2009

Silva, Vanderson Botelho da
S586m Um modelo de confiança certificado baseado em assinatura digital aplicado
2009 a sistemas multiagente / Vanderson Botelho da Silva ; orientador, Edson Emílio Scalabrin. – 2009.
117 p. : il. ; 30 cm

Dissertação (mestrado) – Pontifícia Universidade Católica do Paraná, Curitiba, 2009
Bibliografia: p. 80-93

1. Assinatura digital. 2. Criptografia. 3. Agentes inteligentes (Software). 4. Informática. I. Scalabrin, Edson Emilio. II. Pontifícia Universidade Católica do Paraná. Programa de Pós-Graduação em Informática. III. Título.

CDD 20. ed. – 005.82

Pontifícia Universidade Católica do Paraná
Centro de Ciências Exatas e de Tecnologia
Programa de Pós-Graduação em Informática

ATA DE DEFESA DE DISSERTAÇÃO DE MESTRADO
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

DEFESA DE DISSERTAÇÃO Nº 20/2009

Aos 14 dias do mês de setembro de 2009 realizou-se a sessão pública de Defesa da Dissertação **"Um Modelo de Confiança Certificado Baseado em Assinatura Digital Aplicado a Sistemas Multiagente"**, apresentada pelo aluno **Vanderson Botelho da Silva** como requisito parcial para a obtenção do título de Mestre em Informática, perante uma Banca Examinadora composta pelos seguintes membros:

Prof. Dr. Edson Emílio Scalabrin PUCPR (Orientador)	(*assinatura*)	Aprovado (*aprov/reprov.*)
Prof. Dr. Emerson Cabrera Paraiso PUCPR		Aprovado
Prof. Dr. Alessandro Zimmer UFPR		APPROVADO

Conforme as normas regimentais do PPGIa e da PUCPR, o trabalho apresentado foi considerado aprovado (*aprovado/reprovado*), segundo avaliação da maioria dos membros desta Banca Examinadora. Este resultado está condicionado ao cumprimento integral das solicitações da Banca Examinadora registradas no Livro de Defesas do programa.

Prof. Dr. Mauro Sérgio Pereira Fonseca
Diretor do Programa de Pós-Graduação em Informática

Programa de Pós-Graduação em Informática
PPGIa
PUCPR

Rua Imaculada Conceição, 1155 Prado Velho CEP 80215 901 Curitiba Paraná Brasil
Tel.: (41) 3271 1669 Fax: (41) 3271 2121 www.ppgia.pucpr.br

Aos meus pais,
minha amada esposa Mirinha,
meus queridos filhos Heitor e Felipe.

Agradecimentos

Primeiramente a Deus por ter me criado e salvado. Seus planos em minha vida foram maravilhosos e a conclusão de mais esta etapa é uma prova de sua presença em mim.

Aos meus pais, por me darem o amor e a educação necessária para ser um cidadão de bem, em especial à minha mãe por seu caráter e exemplo que até hoje influenciam a minha vida.

À minha amada esposa Mirinha, por seu apoio nos momentos mais difíceis desta jornada, sempre me encorajando, mesmo sofrendo com minha ausência principalmente nos últimos dias deste projeto.

Ao professor Dr. Edson Emílio Scalabrin pelos diversos momentos entre aulas, reuniões e conversas que auxiliaram, desde a minha escolha do grupo de agentes de software até a conclusão desta dissertação. Suas orientações foram valiosas para o meu crescimento profissional.

À CAPES pelo apoio financeiro.

Aos meus atuais e antigos chefes, Elizier Santos, Mauro Simião e Carlos Murasse que em nome da direção do SERPRO (Serviço Federal de Processamento de Dados) me deram totais condições para desempenhar minhas atividades neste projeto ao longo destes dois anos e meio.

Aos meus amigos, que não são poucos e, portanto não poderei nomear todos aqui, por sempre me animarem e diretamente contribuírem para o meu sucesso.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Símbolos

AES	*Advanced Encryption Standard*
ART	*Agent Reputation and Trust*
BDI	*Belief Desire Intention*
CAST	*Carlisle Adams and Stafford Tavares*
DEA	*Data Encryption Standard*
DSA	*Digital Signature Algorithm*
FIPA	*Foundation for Intelligent Physical Agents*
IDEA	*International Data Encryption Algorithm*
JADE	*Java Agent Development Framework*
PGP	*Pretty Good Privacy*
RSA	*Rivest, Shamir, Adleman*
TDES	*Triple Data Encryption Standard*

Resumo

Este trabalho apresenta um modelo de confiança que visa garantir a legitimidade da informação veiculada entre agentes que coabitam em um ambiente aberto. Em termos gerais, no ambiente proposto, um agente provedor *b* realiza um serviço para um agente cliente *a*. O agente *a* devolve ao agente *b* uma avaliação criptografada *r* sobre o serviço fornecido *c*. O agente *b* usará *r* como testemunha quando ele for inquirido a realizar novamente o serviço *c* para qualquer outro agente cliente. As nossas hipóteses são: (i) o controle sobre os testemunhos pode ser distribuído à medida que os mesmos são armazenados localmente pelos agentes avaliados, i.e., cada agente avaliado é o proprietário de sua avaliação; e (ii) os testemunhos, fornecidos pelos agentes servidores sobre os seus serviços, podem ser considerados legítimos à medida que os pareceres são criptografados por mecanismos de chaves assimétricas. Esta abordagem visa reduzir as atuais limitações dos modelos de confiança baseados, respectivamente, na experiência direta entre os agentes (*confiança direta*) e na experiência indireta obtida por meio de testemunhos (*confiança indireta*). A confiança direta possui baixo desempenho vis-à-vis à dificuldade de um agente cliente *a* realizar um número de interações com um agente provedor *b* para produzir base de experiência significativa. A confiança propagada depende do altruísmo das testemunhas para compartilhar suas experiências. Finalmente, o modelo empírico foi testado em um sistema multiagente, denominado *CRONOS*, aplicado ao mercado de ações, no qual agentes provedores fornecem recomendações de compra ou venda de um ativo e agentes clientes escolhem os provedores em função de suas reputações. Os resultados mostraram que o modelo de confiança proposto torna os agentes mais eficientes na escolha dos seus parceiros.

Palavras-Chave: modelo de confiança, reputação, sistema multiagente, assinatura digital.

Abstract

This work presents a certified confidence model which aims to ensure credibility for information exchanged among agents which inhabit an open environment. Generally speaking, the proposed environment shows a supplier agent *b* which delivers service for a customer agent *a*. The agent *a* returns to *b* a cryptographed evaluation *r* the service delivered. The agent *b* will employ *R* as testimonial when requested to perform the same task for a distinct customer agent. Our hypotheses are: (i) control over testimonials can be distributed as they are locally stored by the assessed agents, i.e., each assessed agent is the owner of its testimonials; and (ii) testimonials, provided by supplier agents on their services, can be considered reliable since they are encapsulated with public key cryptography. This approach reduces the limitations of confidence models based, respectively, on the experience resulted from direct interaction between agents (*direct trust*) and on the indirect experience obtained from reports of witnesses (*indirect trust*). Direct trust is a poor-quality measure for a customer agent *a* hardly has enough opportunities to interact with a supplier agent b so as to grow a useful knowledge base. Indirect trust depends on the willingness of witnesses to share their experiences. The empiric model was tested in a multiagent system, called *CRONOS*, applied to the stock market, where supplier agents provide recommendations for buying or selling assets and customer agents then choose suppliers based on their reputations. Results demonstrate that the confidence model proposed enables the agents to more efficiently choose partners.

Keywords: trust model, reputation, multiagente system, digital signature.

Capítulo 1

Introdução

1.1 Contextualização

Atualmente, a complexidade das aplicações em termos de volume de dados e número de funcionalidades requer uma abordagem distribuída e flexível, na medida em que os sistemas devem ser capazes de adaptarem-se dinamicamente as modificações de estrutura e de ambiente. Neste contexto, os sistemas multiagente são fortes candidatos à construção de arquiteturas abertas distribuídas e flexíveis. Capazes de oferecer grandes quantidades de serviços por meio de trabalho coletivo, sem impor uma estrutura *a priori*. Entretanto, se a distribuição do controle e dos dados torna o sistema mais robusto *vis-à-vis* a disponibilidade de um serviço, a ausência de um controle central enfraquece as relações de confiança entre as partes envolvidas de um sistema multiagente. Segundo Huynh *et al.* (2006) a questão central é, em um sistema aberto[1], como um agente pode confiar em um estranho?

Diversos estudos mostram como os *modelos de confiança* podem se tornar uma alternativa viável à redução dos riscos envolvidos em comunidades virtuais abertas (Castelfranci & Falcone, 1998; Griffiths, 2005; Fullan, *et al*., 2005; Mui, *et al*., 2002; Sabater & Sierra, 2001; Huynh, *et al.,* 2006). Por meio de duas abordagens clássicas é possível identificar comportamentos inadequados de agentes em uma sociedade complexa, a saber: *confiança direta, onde os agentes* utilizam suas experiências diretas e *confiança indireta, onde os agentes utilizam-se* de diferentes testemunhos.

Apesar de ser uma área de pesquisa relativamente nova, os modelos de confiança vêm ganhando destaque como uma subárea de pesquisa dos sistemas

[1] Sistema Multiagente Aberto são compostos por comunidades de agentes heterogêneos, nos quais possuem a liberdade de entrar e sair de comunidades de agente (Huynh, *et al.*, 2006).

multiagente, porém diversos problemas, normalmente relacionados à arquitetura distribuída, ainda não apresentam uma solução estável. **Um deles é a manutenção descentralizada das informações de confiança.** Garantir a autenticidade e a integridade das informações em um ambiente aberto, competitivo, onde os agentes podem mentir ou comportar-se maliciosamente ainda é um grande desafio. O gerenciamento eficiente das informações a cerca da confiança trocadas entre agentes virtuais heterogêneos é o foco deste trabalho. Em outras palavras, a nossa proposta é: como manter a gestão distribuída da confiança, garantindo a eficiência e a segurança para as entidades virtuais envolvidas.

1.2 Problema

O propósito de um *modelo de confiança* é prover aos agentes a capacidade de identificar, no meio de uma sociedade virtual, quais indivíduos são parceiros confiáveis para determinado tipo de interação. Em um ambiente multiagente aberto, no qual os agentes normalmente são desconhecidos, esta habilidade torna-se essencial, pois as relações entre os agentes dependem fortemente da crença de que eles farão aquilo que dizem fazer (Castelfranchi & Falcone, 1998).

Apesar dos diversos modelos de confiança propostos nos últimos anos, praticamente todos eles são adaptações ou derivações de duas grandes classes de modelos: *diretos* e *indiretos*. A primeira estratégia produz a confiança a partir da interação direta entre dois agentes. A segunda utiliza o compartilhamento de informações (experiências de outros agentes chamadas de *testemunhos*) para obter a *reputação* de um agente. Os modelos diretos possuem a vantagem da *precisão* das informações, pois os próprios agentes calculam a utilidade de suas interações. Os modelos indiretos possuem a vantagem do *desempenho*, pois os agentes não precisam interagir diretamente entre si para obter o valor da reputação (Fritschi & Dorer, 2002). Portanto, a combinação dos dois modelos, que respectivamente contribuem com a precisão e desempenho de um modelo de confiança, é uma das estratégias mais promissoras.

No entanto, a abordagem indireta possui diversos desafios do tipo (Jurca & Faltings, 2006): como encontrar boas testemunhas? Quando utilizá-las? Como calcular sua confiança? Como prover um mecanismo de recompensa que incentive os agentes a testemunhar?. Em ambientes competitivos, o compartilhamento de experiências pode ser utilizado contra o próprio agente que a compartilha, portanto nem sempre eles

estarão dispostos a contribuir colaborativamente. Gerenciar agentes que atuam como testemunhas em um sistema multiagente de larga escala pode ser uma atividade tão complexa quando o próprio modelo de confiança na seleção de bons agentes.

Uma proposta para solucionar o tratamento das testemunhas é descrita no trabalho de Huynh *et al.* (2006), que apresenta uma nova abordagem, denominada de *reputação certificada*, na qual os próprios agentes fornecem avaliações sobre suas atuações anteriores. Estas avaliações são fornecidas apenas aos agentes que desejam interagir, portanto o tratamento dos testemunhos é realizado localmente pelo agente avaliado. Apesar de resolver os principais problemas dos modelos indiretos, a proposta possui problemas conceituais que impedem, atualmente, seu largo uso.

A principal crítica à reputação certificada é o fato dos agentes avaliados poderem selecionar as avaliações que testemunharão sobre eles mesmos. É natural que estes agentes selecionem apenas as avaliações mais favoráveis a eles. A omissão de avaliações negativas ou a seleção exclusiva de boas avaliações resulta em um cálculo impreciso da confiança e distorce o real comportamento do agente. Outro problema do modelo é a ausência de um mecanismo que garanta a segurança das informações trafegadas. Como o agente avaliado possui suas próprias avaliações é necessário garantir que ele não as viole.

Esta dissertação trata a problemática de ter-se um modelo de confiança distribuído baseado na confiança direta e na confiança indireta por meio de testemunhas. A proposta inicial para a reputação certificada foi utilizada e modificada de forma a garantir a precisão das informações. Nesta proposta, a eficiência e a segurança no processo foram asseguradas, respectivamente, pela abordagem de manter os testemunhos armazenados localmente no próprio agente avaliado e prover um canal de comunicação seguro para o tráfego das informações de confiança.

1.3 Objetivos

Primeiramente, cabe destacar que este trabalho não visa à construção de um sistema multiagente, mas a concepção, análise e construção de uma de suas partes, que neste estudo é o modelo de confiança. Apesar dos experimentos apresentados, no Capítulo 4, tratarem especificamente de um sistema multiagente para o mercado financeiro, este trabalho sugere um modelo de confiança genérico.

1.3.1 Objetivo Geral

Propor um modelo de confiança que auxilie os agentes de uma comunidade virtual a selecionar bons parceiros para interação, diminuí-se o risco das relações entre agentes heterogêneos em um sistema multiagente aberto.

Desta forma, este trabalho consiste em examinar os problemas dos atuais modelos de confiança, propor uma solução que permita reduzir tais problemas e empiricamente mostrar as melhorias alcançadas.

1.3.2 Objetivos Específicos

1. Realizar um amplo estudo sobre os modelos de confiança;
2. Conceber um modelo de confiança que permita eliminar ou reduzir as principais limitações encontradas nas atuais abordagens de confiança;
3. Construir um simulador que permita avaliar o modelo de confiança proposto.
4. Avaliar o desempenho do modelo proposto em relação aos demais modelos estudados.

1.4 Motivação

Confiança é um elemento fundamental em todos os momentos das nossas vidas. Niklas Luhman, sociólogo, afirma: "*A total ausência de confiança impediria o nosso levantar da manhã*" (Luhmann, 1979). Ela faz parte da "liga" que uni nossas sociedades. Sem ela os governos perderiam suas regras e as pessoas não poderiam trabalhar juntas de forma cooperativa. A confiança ajuda a reduzir a complexidade de decisões que devem ser tomadas sob circunstâncias de grande risco. Da mesma maneira, a *reputação* é um conceito universal que está presente nas sociedades humanas há muito tempo. Para os antigos gregos, o conceito de reputação possuía um papel essencial na organização das sociedades humanas, pois ela era um dos mais importantes elementos utilizados para construir a confiança recíproca.

Até pouco tempo, ambos os conceitos eram aplicados apenas em sociedades humanas, estudadas nos campos da sociologia, psicologia e filosofia. Com o surgimento da Internet e por conseqüência das sociedades virtuais, não necessariamente humanas, é agregada uma nova dimensão para estes antigos, mas relevantes conceitos. A pesquisa científica que trata os mecanismos de confiança e reputação para sociedades virtuais é

uma disciplina recente que visa prover maior credibilidade e desempenho às comunidades virtuais.

A Ciência da Computação tem, ao longo dos anos, direcionado suas atenções para as abordagens baseadas em uma única máquina para soluções estruturadas em redes de computadores e computação distribuída. Analogamente, a inteligência artificial também direcionou rapidamente seus esforços da IA clássica, baseada na inteligência isolada, ao paradigma distribuído onde a resolução de problemas é construída a partir da inteligência social e coletiva de indivíduos. Este novo paradigma chamado de *Sistemas Multiagente* (SMA) juntamente com o surgimento de tecnologias voltadas às sociedades virtuais, como o caso dos sistemas de comércio eletrônico, são responsáveis pelo crescente interesse nos mecanismos de confiança e reputação aplicados as sociedades virtuais.

Um agente é uma entidade capaz de agir em um ambiente, comunicar-se diretamente com outros agentes, mover-se por um conjunto de tendências, possui seus próprios recursos, consegue perceber alterações no seu ambiente, possui habilidades e seus comportamentos tendem a atingir seus níveis de satisfação ou objetivos (Ferber, 1999). Quando inseridos em ambientes abertos, na presença de agentes heterogêneos, a capacidade de interação deve prever eventuais comportamentos inadequados em relação aos demais agentes desta comunidade. Analogamente, quando estamos em uma grande metrópole, é natural nos preocuparmos com comportamentos inadequados de estranhos como roubos, assaltos, violência, entre outros. Os conceitos de confiança e reputação possuem similar relevância nos dois tipos de sociedade.

1.5 Publicações

Parte do trabalho apresentado nesta dissertação produziu as seguintes publicações:

- BOTELHO, V. S.; *et al.* Certified Trust Model; in 5th IFIP Conference on Artificial Intelligence Applications & Innovations, April 23-25, 2009, Thessaloniki, Grécia.
- BOTELHO, V. S.; *et al.* Encrypted Certified Trust in Multi-Agent System; in 13TH International Conference on CSCW in Design, April 22-24, 2009, Santiago, Chile.

1.6 Organização da Dissertação

O restante deste trabalho está organizado em cinco capítulos e dois apêndices que são distribuídos em três blocos: suporte teórico ao tema abordado (Capítulos 2 e 3); idéias propostas e conclusões aferidas (Capítulos 4 e 5) e suporte aos experimentos (Apêndice A e B). O Capítulo 2 analisa e apresenta um conjunto de modelos de confiança e reputação, bem como alguns *frameworks* utilizados para avaliar estes modelos. O Capítulo 3 trata das definições de segurança em sistemas de informação baseadas em técnicas de criptografia e assinatura digital. O Capítulo 4 apresenta o modelo de confiança proposto neste trabalho e descreve o sistema multiagente utilizado para avaliar seu desempenho, apresenta os experimentos realizados e seus resultados. O Capítulo 5 apresenta as conclusões deste trabalho e direciona trabalhos futuros sobre o tema. O Apêndice A apresenta a fundamentação teórica sobre o mercado de ações, utilizada para a construção do sistema multiagente do Capítulo 5. O Apêndice B apresenta a modelagem utilizada para a construção da ferramenta CRONOS.

Capítulo 2

Modelos de Confiança para SMA

2.1 Introdução

É indiscutível a importância da confiança e reputação para as sociedades humanas. Portanto, este tema não apresenta grande surpresa para as áreas de pesquisas que já tratavam, por distintas perspectivas, ambos os conceitos. Economia (Andreoni & Miller, 1993; Selten, 1978), psicologia (Bromley, 1993), biologia (Pollock, 1992), filosofia (Katz, 1953) são algumas das áreas que têm investidos esforços no estudo da confiança e reputação. Entretanto, há outra disciplina onde as pesquisas de reputação e confiança tem ganhado crescente evidência nestes últimos anos. Estamos falando da ciência da computação, mais especificamente da *Inteligência Artificial Distribuída* (IAD). Duas grandes classes de sistemas computacionais têm contribuído significativamente para o estudo da confiança e reputação: o paradigma dos sistemas multiagente e o crescimento dos sistemas de e-commerce.

Este Capítulo oferece uma visão geral a respeito dos atuais modelos de confiança aplicados aos sistemas multiagente. Ele também apresenta os principais conceitos relacionados ao tema e tenta estabelecer um conjunto de critérios para a classificação destes modelos. A segunda parte do Capítulo é dedicada à apresentação dos modelos de confiança mais relevantes propostos ao longo dos últimos anos e alguns projetos voltados especificamente para a avaliação de desempenho destes modelos.

2.2 Confiança

Os agentes de software são entidades, que por natureza, possuem limitações computacionais que restringem sua capacidade de interação, principalmente quando inseridos em ambientes de grande escala. Além disso, a limitação do próprio ambiente como largura de banda e velocidade dos canais de comunicação restringe a capacidade sensorial dos agentes. Portanto, em situações reais é tecnicamente impossível para um agente chegar a um estado de informação perfeita a cerca do ambiente e dos demais agentes que compõem este ambiente (Axelrod, 1984; Binmore, 1992; Russell & Norvig, 1995). Apesar do significativo grau de incerteza, natural a este ambiente, os agentes devem continuar interagindo entre si e tomando decisões importantes, porém mantendo sempre um nível aceitável de segurança. Então, surge naturalmente entre os indivíduos, a necessidade de *confiar* uns nos outros, de forma a minimizar riscos inerentes às interações entre elementos desconhecidos de um ambiente aberto.

O termo *confiança* possui definições distintas para cada área de estudo. Para os sistemas multiagente, Dasgupta, (1998) afirma que confiança é a convicção que um agente possui de que outro agente fará o que diz que irá fazer, determinando uma oportunidade para atrair para si uma recompensa mais alta. A confiança representa as convicções de um indivíduo em relação à probidade do outro (Ramchurn, *et al.*, 2004). Este conceito é fundamental para definir as regras de interação de uma sociedade, seja ela composta por pessoas, animais ou por sistemas virtuais como no caso dos sistemas multiagente (Resnick, 2000). Segundo Castelfranchi (1998), confiança é um modelo mental que compõe um agente BDI (*Belief, Desire, Intention*).

A relação entre o conceito de confiança e o paradigma dos sistemas multiagente é explicada por Castelfranchi, (1998). Para ele a delegação de tarefas, essencial ao trabalho colaborativo dos agentes, está fortemente vinculada à confiança. Portanto, quando um agente delega uma tarefa a outro, ele na verdade gera uma crença de que o outro agente poderá executar a tarefa delegada, ou seja, ele *confia* no agente delegado. Um grande problema na delegação de tarefas pode ser a ausência de garantia ao cumprimento das tarefas atribuídas para agentes desconhecidos. Desta forma, surge a necessidade de mecanismos que diminuam os riscos de se ter uma tarefa não cumprida. Diversas pesquisas como as de Griffiths, (2005) e Fullam, (2005) mostram que a confiança pode ser útil para reduzir os riscos associados às interações dos agentes. Como a delegação de tarefas é uma peça fundamental aos protocolos de cooperação, as

pesquisas relacionadas à confiança também ganham papel de destaque, tornando-se uma área de grande relevância no estudo dos sistemas multiagente.

2.3 Reputação

Reputação não é um assunto trivial por tratar-se de um tema multidisciplinar (Mui, *et al.*, 2002). A biologia tenta explicar como a reputação atua na competição entre as espécies (Nowak, 1998). Economistas tentam decifrar o comportamento "irracional" dos investidores nos mercado financeiro (Kreps, 1982). Cientistas da computação tentam utilizar a reputação para modelar confiança entre indivíduos ou organizações no âmbito do comércio eletrônico (Mui, 2002).

Quando o sentimento de confiança deixa de ser apenas uma crença pessoal, de um ser em relação a outro, e ganha abrangência coletiva, i.e., a confiança de uma sociedade sobre algo ou alguém, denomina-se esse fenômeno de *Reputação* (Pujol, 2002). Ramchurn, (2004) complementa dizendo que ela pode ser "*derivada da agregação de opiniões de uma comunidade*". Da mesma forma que a confiança resulta em reputação a recíproca também é verdadeira; a reputação tem o poder de propagar a confiança. Quando ouvimos bons comentários a cerca de um filme, mesmo sem conhecê-lo, sentimos naturalmente o desejo de assisti-lo. Este exemplo ilustra como a reputação pode influenciar diretamente a construção de confiança.

A reputação é fundamental aos sistemas multiagente de larga escala, na medida em que um *sistema de reputação* pode evitar que os agentes interajam desnecessariamente. Por meio da reputação, por exemplo, compradores podem selecionar bons vendedores, além disso, os vendedores podem comporta-se melhor a fim de evitar a perda de futuros negócios por conta de uma má reputação. Surge então a necessidade de um *sistema de reputação* que tem como principais objetivos coletar, distribuir e agregar *avaliações* sobre o comportamento passado dos seus participantes Resnick, (2000). Desta forma, um modelo de reputação visa: promover formas para incentivar a produção de avaliações; promover estruturas para coletar e armazenar as avaliações produzidas e promover mecanismos para recompensar avaliações conforme seu nível de acerto.

Para tratar a organização, recuperação e agregação das avaliações de outros agentes, alguns sistemas de reputação utilizam o conceito de *cadeia social* (Burt, 1982; Buskens, 1998). Semelhante as sociedades humanas, esta cadeia assume que os agentes estão interconectados através de vários mecanismos de comunicação e relacionados

entre si por meio de papéis. Transversalmente as redes de relacionamento social, os agentes atuam como *testemunhas* de suas próprias interações e podem compartilhar suas experiências com outros agentes (Panzarasa, *et al*., 2001). A construção de uma base de avaliações, por meio de inferências, resulta em novas descobertas; o vendedor demora a fazer suas entregas ou o comprador é um péssimo pagador para certo tipo de produto. Estas avaliações são compartilhas pelos diversos nós da rede provendo o conceito de reputação.

2.4 Classificação

Confiança e reputação são assuntos que podem ser analisados por diferentes perspectivas e aplicados as mais diversas áreas. Portanto, a tentativa de classificá-los não é uma tarefa trivial. Esta seção propõe uma classificação aos modelos de confiança, levando em consideração um conjunto de características de forma a agrupá-los em classes distintas.

2.4.1 Tipo de Paradigma

Os modelos de confiança podem ser classificados em função de dois paradigmas: cognitivo e matemático. Conforme defendido por (Castelfranchi & Falcone, 1998), nos modelos baseados sob a visão cognitiva, a confiança é tratada como uma convicção derivada de um conjunto de crenças[2], portanto confiança é uma função que mensura o grau destas crenças. Em oposição ao pensamento cognitivo, tem-se o paradigma matemático (Teacy, *et al.*, 2005; Zheng, *et al.*, 2006), que não utiliza crenças ou intenções da abordagem BDI para a modelagem de confiança. A confiança e a reputação não são frutos de um processo de estados mentais de um agente cognitivo, mas é o resultado de um modelo matemático que utiliza funções sobre o conjunto de interações passadas. Apesar do modelo de confiança cognitivo tentar reproduzir mais fielmente o mecanismo humano de raciocínio, (Sabater, 2003) acredita que o processo utilizado pelos modelos matemáticos possui vantagens como a simplicidade para a construção, teste e comparação em relação a outros modelos matemáticos, além de garantir alto desempenho, essencial em sistemas de larga escala.

[2] Do inglês *Belief*, representa um dos três estados mentais descritos na literatura dos agentes BDI *Belief, Desire and Intention* (Rao & George, 1995).

2.4.2 Origem da Informação

É possível classificar os modelos de confiança quanto à origem da informação, i.e., o local onde as informações de confiança são geradas. Estas informações representam a base para se calcular o valor da confiança. A partir da origem da informação consideram-se duas principais classes: as experiências diretas dos agentes em suas interações, dando origem os denominados *modelos diretos* e as informações referentes a testemunhos, relatos de terceiros, que dão origem aos denominados *modelos indiretos*.

Cada classe de modelo depende profundamente das capacidades sensoriais dos seus agentes. Idealmente, espera-se que os agentes possam manipular informações de várias origens; isto garante maior precisão no cálculo da confiança. Entretanto, a inclusão de novas capacidades ao tratamento das informações resulta em maior complexidade à construção do modelo de confiança e, consequentemente, aos seus agentes.

Modelos Diretos

Os modelos ditos *diretos* assumem que a interação e a observação dos agentes são as principais fontes de informação para calcular o valor da confiança. Neles, os agentes alimentam sua base de conhecimento a partir de experiências passadas. Cada experiência pode ser adquirida a partir de dois fatores: a *interação direta* com outros parceiros, na qual os agentes utilizam o grau de satisfação de suas interações passadas para projetar o atual nível de confiança dos seus parceiros (Mass & Shehory, 2001); ou a *observação* de interações, neste caso o agente calculador da confiança não participa ativamente da interação, mas atua como um expectador (Mui, *et al.*, 2002). Neste segundo caso, os agentes diminuem seus riscos; eles aprendem com os acertos e erros dos outros. Entretanto, para que a observação seja possível, a interação deve ser vista pelo modelo de confiança como um evento público, no qual qualquer agente pode ter acesso. Em debates políticos, por exemplo, os eleitores observam as ações e reações de seus candidatos diante do confronto de idéias, e a partir da observação é possível decidir em qual candidato eleger.

Os modelos diretos são considerados os modelos de confiança mais precisos, visto que a informação vem diretamente do próprio agente que calcula a confiança (Sabater, 2003). Entretanto, o modelo direto depende da constante interação dos agentes para que se mantenha atualizado, o que é, em sistemas de larga escala, praticamente

inviável manter tal nível de interação com todos os agentes da comunidade, além de que os agentes, durante esta fase exploratória, podem realizar péssimas interações com agentes desconhecidos. Por estes motivos, os modelos diretos não são utilizados para a descoberta de bons parceiros, mas associados a outros modelos podem incrementar a base de conhecimentos dos agentes e contribuir para melhorar a precisão dos cálculos da confiança.

Modelos Indiretos

Em grandes comunidades não é possível que todos os indivíduos se conheçam diretamente. Isto demandaria muito tempo para construir as interações diretas entre todos os membros da comunidade. No entanto, continua sendo fundamental que cada membro possa representar um valor de confiança aos demais membros de sua sociedade. Assim, sem evidências diretas a cerca de um indivíduo, a confiança deve ser inferida a partir de informações *indiretas*.

A primeira forma de inferência indireta é a chamada de *reputação pré-derivada* onde os agentes utilizam suas convicções passadas sobre outros agentes desconhecidos (Zacharia, 1999). Em uma sociedade humana, cada um de nós tem convicções adquiridas sobre alguém ou algo que não conhecemos muito bem e chamados isso de *pré-conceito* (Nowak & Sigmund, 1998). A discriminação sexual ou racial pode ser vista como uma consequência destas convicções.

Os modelos de confiança também podem utilizar como informação indireta a chamada *reputação derivada por grupo*. Trata-se de um mecanismo de confiança complementar que atribui inicialmente um valor de confiança a cada grupo de agente. Portanto, cada agente possui um valor, conforme seu grupo, e a medida que outras fontes de informações são utilizadas, o cálculo da confiança vai ser tornando mais preciso (Halberstadt & Mui, 2001).

Um dos tipos indiretos de informação mais conhecidos e pesquisados da atualidade são as *testemunhas*. A testemunha representa um indivíduo que utiliza suas experiências passadas para reportar a probidade de alguém. Portanto, o modelo indireto impõe a necessidade de compartilhamento das bases de conhecimentos dos agentes. Quando nos referimos a modelos de confiança aplicados aos sistemas multiagente, os testemunhos podem ser chamados de *avaliações*. O compartilhamento das avaliações evita que os agentes corram o risco de interagir com agentes desconhecidos. Diferente

do modelo direto baseado em observações, as testemunhas não exigem que as interações sejam eventos públicos, portanto, podem ser aplicados a mais tipos de situação.

2.4.3 Visibilidade

A confiança ou reputação de um indivíduo pode ser vista como uma propriedade *global* acessível por todos os membros de uma comunidade ou como uma propriedade *pessoal*, também chamada de *local*, na qual cada indivíduo da comunidade tem a sua representação de confiança em relação ao outro. No primeiro caso, o valor de confiança é calculado a partir das avaliações de cada indivíduo que no passado interagiu com o indivíduo avaliado. Seu valor é tornado público e acessível por todos os membros da comunidade. A atualização deste valor é feita sempre que um indivíduo lança uma nova avaliação a cerca do indivíduo avaliado. No segundo caso, quando um indivíduo *a* obtém uma avaliação de confiança em relação a um indivíduo *b,* seja por suas experiências diretas ou conforme testemunhos de outros indivíduos, a confiança de *b* não é generalizada aos demais integrantes da comunidade, pois a confiança local representa apenas uma percepção de *a* sobre *b.* Portanto, *b* pode ser confiável para uns e não confiável para outros.

A confiança *global* é comumente utilizada em sistemas de comércio eletrônico como o E-bay (Ebay, 2009) e Mercado Livre (Mercado, 2009) que tratam grandes comunidades virtuais compostas por milhões de usuários. O principal problema dos sistemas que consideram a confiança como uma propriedade global é a falta de personalização para o valor da confiança. Algo que é ruim para alguém, pode ser aceitável para outro e vice-versa. A abordagem da confiança global presume que todas as características boas e más de um indivíduo possuem o mesmo peso para todos os membros da comunidade. Isto nem sempre é verdade em interações complexas. Quando os indivíduos necessitam tratar eventos com múltiplos atributos, comum em um processo de negociação, todos os diferentes atributos que influenciam em uma decisão possuem pesos distintos para cada agente, portanto necessitam de uma representação específica de confiança.

Os sistemas que consideram a confiança como uma propriedade personalizada assumem que cada agente utiliza suas experiências diretas ou observações, além de associar os relatos ou testemunhos de outros agentes a cerca do agente avaliado para construir localmente uma representação de confiança. Esta abordagem aproxima-se do comportamento humano, onde os indivíduos definem seus critérios para avaliar o nível

de confiança de alguém. Entretanto, seu uso é mais indicado para comunidades de pequeno a médio porte, nas quais os agentes podem se encontrar com frequência para a construção de fortes vínculos entre si.

2.4.4 Dimensões

Quando confiamos em um experiente engenheiro para a construção de um edifício isso não significa que devemos confiar, neste mesmo indivíduo, para pilotar um avião. A reputação de bons cozinheiros pouco ajuda quando procuramos selecionar garçons para um evento. Parece claro que a confiança é uma propriedade com forte dependência de um contexto. Griffiths, (2005) denomina estes contextos de *dimensões* e descreve a necessidade de se ter modelos que manipulem informações de confiança e reputação sob múltiplas dimensões. A pesar da evidente importância, a inclusão desta capacidade sob os modelos de confiança pode produzir custos em termos de complexidade e resultar em efeitos colaterais nem sempre desejáveis.

Um modelo de confiança *monodimensional* é projetado para associar um único valor de confiança a um indivíduo, sem levar em consideração outros contextos. Do contrário, um modelo *multidimensional* é composto por um mecanismo que trata simultaneamente diversos contextos de forma que cada contexto possa receber diferentes valores de confiança. Um indivíduo pode ser um competente médico, ótimo pai e péssimo marido. Nem sempre a utilização de múltiplas dimensões é essencial a um modelo de confiança, principalmente quando o modelo está focado em cenários específicos.

2.4.5 Comportamento

O mecanismo utilizado pelos modelos de confiança para tratar agentes com diferentes níveis de comportamento é também uma forma de classificação. São considerados três níveis para categorizar os modelos de confiança do ponto de vista do comportamento dos agentes:

- Nível 0: Comportamento malicioso desconsiderado. O modelo confia em um grande número de agentes que provem avaliações honestas e que, portanto, pode anular o efeito de algumas falsas avaliações fornecidas por um agente malicioso.
- Nível 1: O modelo assume que os agentes podem omitir ou influenciar alguma informações, mas jamais mentir.

- Nível 2: O modelo assume que os agentes podem mentir e especifica mecanismos para tratar mentiras.

2.4.6 Tipo de Informação

O tipo de informação trocada pelas testemunhas é o último critério para a classificação dos modelos de confiança. Aqui são definidos dois grandes grupos: os modelos que representam a informação por meio de um valor *booleano* e os modelos que tratam as informações por meio de valores contínuos. A escolha de uma das abordagens tem grande influência no projeto do modelo de confiança. Normalmente, os modelos baseados nos métodos probabilísticos operam com informações *booleanas,* enquanto que os modelos matemáticos usam informações contínuas.

2.5 Modelagem

A construção de um modelo de confiança aplicado a sistemas multiagente exige observar um conjunto de requisitos mínimos quanto ao tipo de aplicação que será desenvolvida. Antes de tudo, deve-se fundamentalmente compreender os principais problemas que se deve enfrentar na construção de um modelo de confiança. Ramchurn, (2004) elenca estas questões por meio de três perguntas:

- Quais protocolos e mecanismos serão utilizados para os agentes encontrarem seus parceiros?
- Como os agentes decidem com quem interagir?
- Como os agentes decidem quando interagir?

A partir destas questões chaves é possível projetar uma série de requisitos necessários à concepção de um modelo de confiança. Alguns destes requisitos são descritos no trabalho de Fullam (2005a), que propõe um sistema para avaliação de modelos de confiança e reputação, denominado *Agent Reputation and Trust (ART) Testbed*. O sistema visa comparar diversas tecnologias e fornecer um conjunto de ferramentas que permita facilmente realizar experimentos para avaliação dos sistemas de confiança. Para comparar dois ou mais modelos de confiança, o ART define uma lista de requisitos essenciais para a maioria destes modelos:

- **Precisão**: refere-se à previsão correta do comportamento futuro de outro agente. A precisão pode ser atribuída em função do grau de semelhança entre o valor de confiança calculado pelo modelo de confiança e o valor real da confiança do individuo avaliado.

- **Adaptabilidade**: refere-se à capacidade em acomodar características dinâmicas de confiança. A confiança pode ser composta por diversos atributos, como por exemplo, tempo de resposta, qualidade da resposta, entre outros. Estes atributos devem ser de fácil absorção pelo modelo de confiança.
- **Rápida convergência**: refere-se à capacidade em determinar rapidamente a confiança de agentes desconhecidos que entram no sistema. A rápida convergência é necessária para identificar agentes que tentam camuflar sua reputação ao entrar e sair repetidamente do sistema.
- **Multidimensional**: refere-se à capacidade de distinguir a confiança dos agentes a partir de várias dimensões. Por exemplo, na compra de um produto o cliente deve avaliar o vendedor em relação ao preço, tempo de entrega, atendimento entre outros atributos.
- **Eficiência**: refere-se à eficiência dos algoritmos utilizados para calcular a confiança dos agentes. Visa destacar os modelos que possuem menor custo computacional para processar as informações de confiança. A eficiência pode ser medida, por exemplo, por meio do tempo utilizado para completar uma atualização no modelo de confiança.

Além dos requisitos utilizados para avaliar a qualidade de um modelo de confiança, Fullam (2005) descreve três características importantes para avaliação individual dos agentes:

- **Identificação de agentes maliciosos:** um agente deve possuir a capacidade de identificar e isolar os agentes indignos de confiança, recusando-se a interagir com eles.
- **Cálculo de interação:** um agente deve calcular a utilidade de uma interação a o qual o acordo será cumprido. Isto é útil para negociar melhores condições do acordo, como por exemplo, o pagamento apropriado para um serviço ou produto.
- **Decisão com quem interagir**: um agente deve saber escolher seus parceiros de interação corretamente e prever se o acordo entre eles será cumprido.

2.6 Modelos de Confiança

Ao longo dos últimos anos diversos modelos de confiança têm sido propostos, cada um utilizando características, paradigmas e tecnologias diferentes. Nesta seção é apresentada uma vasta seleção destes modelos a fim de prover uma visão ampla desta área de pesquisa.

2.6.1 E-commerce

Um dos grandes fenômenos da Internet são os sistemas de comércio eletrônico como o eBay, (2009), Amazon, (2009), OnSale (2009) que permitem conectar consumidores e fornecedores de diversos produtos e serviços garantindo, por meio dos seus modelos de confiança, um canal seguro para a realização de negócios pela *Web*. O eBay é um dos maiores sistemas de comércio eletrônico do mundo. Ele é composto por uma comunidade de milhões de usuários. Seu modelo de confiança baseia-se em uma arquitetura centralizada, onde os compradores e vendedores, ao final de uma negociação, enviam um *feedback* a cerca do respectivo parceiro de negociação. Os usuários podem atribuir três tipos de valores aos *feedbacks*: positivo, negativo ou neutro. Todos os *feedbacks* são armazenados num repositório global e os cálculos são realizados de maneira centralizada. A reputação dos participantes pode ser mensurada pela quantidade de *feedbacks* positivos e negativos ou pela porcentagem dos *feedbacks* positivos em relação ao número total de *feedbacks*. Os sistemas Amazon e OnSale utilizam a mesma abordagem para calcular o valor da confiança dos seus usuários.

Todos os sistemas citados consideram a confiança como uma propriedade global e a representam por meio de um único valor. O valor da confiança para estes sistemas não é dependente do contexto, portanto são sistemas *monodimensionais*. As avaliações são enviadas pelos próprios indivíduos da comunidade, que interagiram com os indivíduos avaliados. Tem-se então que a obtenção da informação é *indireta* e baseada em *testemunhos*. Estes sistemas não possuem um mecanismo para tratar usuários que dão falsos testemunhos (*Nível 0*); um grande número de opiniões verdadeiras diluem as avaliações falsas.

2.6.2 ReGreT

O ReGreT é um modelo de confiança orientado para ambientes complexos de comércio eletrônico, no qual as relações sociais possuem um relevante papel (Sabater,

2003). Diferente da maioria dos modelos de confiança que utilizam apenas as duas principais abordagens de informação (a experiência direta e o testemunho), o ReGreT resgata uma terceira fonte de informação que pode ser útil, as chamadas *relações sociais*. Como consequência direta das interações é possível identificar diferentes tipos de relações sociais entre os membros de uma sociedade. Sociólogos e psicólogos têm estudado as redes sociais na sociedade humana ao longo do tempo e como estas redes podem ser utilizadas para avaliar a confiança e a reputação dos seus indivíduos (Pujol, *et al.*, 2003). O modelo de confiança de ReGreT mostra ser possível criar afirmações, mesmo que pequenas, sobre o comportamento de indivíduos utilizando informações obtidas a partir da análise de suas redes sociais.

O modelo de confiança do ReGreT foi projetado para ser utilizado em grandes comunidades de agentes, por meio de sistemas multiagente abertos. Os agentes destas comunidades podem utilizar o conhecimento sobre a estrutura social dos seus parceiros como artifício para superar a ausência de experiências diretas. A combinação das experiências diretas, dos testemunhos e dos conhecimentos sociais permite ao sistema calcular os valores de confiança e reputação dos seus agentes. Além disso, é possível calcular o nível de credibilidade das informações enviadas pelas testemunhas. A confiança para o ReGreT é uma propriedade local e pode ser representa por diferentes valores, um para cada contexto; o sistema considera a confiança dependente do contexto. Estes sistemas possuem um mecanismo para tratar usuários que dão falsos testemunhos, portanto classificam-se no *Nível 2* das categorias mencionadas na seção anterior.

2.6.3 FIRE

O modelo FIRE (Huynh, *et al.*, 2004) tem como principal característica a capacidade de estimar a confiança e a reputação dos seus indivíduos a partir de diferentes fontes de informação. Ele incorpora quatro tipos de informação:

- *interaction trust*: modela a confiança a partir das interações diretas entre os indivíduos — utiliza-se aqui o mesmo mecanismo do ReGret;
- *role-base trust*: atribui os valores iniciais de confiança aos agentes baseado nos papéis que eles desempenham — o valor inicial é calculado conforme o valor médio da confiança dos agentes de um determinado papel;
- *witness reputation*, calcula-se a reputação com base nas avaliações reportadas por outros agentes — as testemunhas.

- *certified reputation*, calcula-se a reputação com base nas referências fornecidas pelo próprio agente avaliado.

O FIRE possui uma arquitetura distribuída aplicada ao tratamento da confiança em sistemas multiagente abertos, no qual cada agente é responsável pelo armazenamento e processamento das informações de confiança.

2.6.4 S. Marsh

Um dos modelos de confiança mais antigos, proposto por (Marsh, 1994), propõe um modelo baseado exclusivamente nas experiências diretas entre agentes. Ele categoriza a confiança em três níveis:

- confiança básica: modela a confiança de um agente independente dos seus parceiros. Ela é calculada a partir de todas as experiências acumuladas por um agente. Boas experiências resultam em um alto nível de disposição.
- confiança geral: é a confiança que um agente possui em relação a um outro, levando em consideração uma situação específica. Ele representa simplesmente a confiança geral de um agente. Ela é denotada por $T_x(y)^t$ que representa a confiança do agente *x* em *y* para um tempo *t*.
- confiança por situação: é a confiança que um agente possui em relação a um outro, levando em consideração uma determinada situação. Ela é denotada por $T_x(y, s)^t$ onde *s* representa a situação considerada na avaliação.

O modelo S. Marsh destaca-se pelo uso dos conceitos de *memória* e *reciprocidade* para calcular a confiança dos agentes. A memória dos agentes é formalizada pelo modelo que trata os principais problemas relacionados à limitação dos agentes quanto ao armazenamento de informações. O segundo conceito que trata a reciprocidade utiliza-se a idéia de que se um agente *a* ajudou um agente *b* no passado e *b* mentiu ou traiu *a*, a confiança de *a* em *b* é reduzida.

2.6.5 Spora

O SPORA é um dos mais clássicos modelos de confiança, seus princípios são utilizados até hoje para a concepção de novos modelos. Segundo Zacharia, (1999), idealizador do modelo, o SPORA propõe um serviço de reputação baseado nos seguintes princípios:

1. Cada novo usuário do sistema recebe o valor mínimo de reputação, à medida que interage com o sistema ou com os membros da comunidade seu valor é atualizado;
2. O valor da reputação de um usuário nunca deve ser abaixo da reputação de um novo usuário. Independente de quão baixo seja seu desempenho;
3. Ao final de cada interação a reputação do usuário avaliado é atualizada conforme a *realimentação* do respectivo parceiro da interação;
4. Dois usuários podem avaliar-se mutuamente uma única vez. Caso dois usuários interajam mais de uma vez, o sistema manterá a mais recente avaliação;
5. Usuários com valores muito elevados de reputação possuem sua reputação reduzida a cada atualização.

Novos usuários possuem o valor mínimo de reputação igual a zero e podem chegar ao valor máximo 3000. A reputação varia conforme o comportamento dos indivíduos: 0,1 para o pior desempenho e 1,0 para o melhor. A reputação dos usuários jamais varia sob um valor negativo, desta forma nenhum indivíduo possui valor menor que a reputação de um novo usuário.

2.6.6 Abdul-Rahman e Stephen Hailes

Abdul-Rahman & Hailes (2000), propõem um modelo de confiança para estruturar comunidades virtuais de agentes de software. Neste estudo, a confiança é categorizada com base em quatro níveis de crenças, conforme a Tabela 2.1.

Nível de Crença	Significado
vt	Muito confiável (*very trustworthy*)
t	Confiável (*trustworthy*)
u	Pouco confiável (*untrustworthy*)
vu	Pouquíssimo confiável (*very untrustworthy*)

Tabela 2.1 Níveis de crenças.

Considerando tal categorização, cada agente deve classificar, para cada contexto, seus parceiros nos quatro níveis de crença. Cada nível dever receber um valor numérico. Desta forma, a confiança é representa por uma *tupla* de quatro posições: $S = (s_{vt}, s_t, s_u, s_{vu})$. Por exemplo, se para certo contexto um agente é classificado como S = (1, 5, 2, 3), significa que ele é confiável, visto que a crença $s_t=5$ possui o maior valor da *tupla*.

Quando há empates entre os quatro valores da *tupla* como: (2, 4, 4, 3) ou (4, 4, 4, 3), o modelo de confiança descreve a situação como uma *incerteza*.

A Tabela 2.2 mostra três possibilidades de incertezas quando há mais valores confiáveis, quando há mais valores não confiáveis ou quando ambos os valores são iguais.

Situação	Símbolo	Significado
$vt + t > u + vu$	u^+	Mais confiável que não confiável
$vt + t < u + vu$	u^-	Mais não confiável que confiável
$vt + t = u + vu$	u^0	Igualmente confiável e não confiável

Tabela 2.2 Níveis de incerteza.

O modelo de confiança de Abdul e Hailes não prevê o armazenamento local das experiências diretas, as *tuplas* são internamente atualizadas pelo agente por meio dos valores obtidos exclusivamente pelas testemunhas. As experiências diretas são utilizadas para comparar o ponto de vista das testemunhas contra a percepção direta do agente. O problema desta abordagem é a falta de personalização da confiança, sem o armazenamento local, os agentes não podem construir confianças customizadas às suas necessidades.

2.6.7 TRAVOS

O modelo de confiança TRAVOS (*Trust and Reputation model for Agentbased Virtual OrganisationS*) proposto por Teacy *et al.*, (2006), também aplica o conceito de confiança à sistemas multiagente. Neste estudo, a confiança é calculada por meio da teoria de probabilidade a partir das interações passadas entre os membros da comunidade. O modelo TRAVOS propõe duas maneiras para representar a confiança dos agentes: por meio da experiência direta ou por meio da reputação obtida pelas informações de testemunhas. Cada agente *gerencia o nível* de confiança dos demais agentes do sistema.

Para TRAVOS, a confiança é representa pela probabilidade de um agente cumprir aquilo que ele afirma fazer. As interações diretas de um agente são consideradas como eventos que podem ser retirados do universo total de eventos para se calcular o valor da *variável aleatória*[3]. Diferentes amostras podem ser selecionadas sob uma mesma população, e amostras diferentes podem resultar em estimativas diferentes.

[3] Variável aleatória é uma variável cujo valor é o resultado numérico de um experimento aleatório. É uma função formada por valores numéricos definidos sobre o espaço amostral de um experimento. (Morettin & Bussab, 2003)

Desta forma, a confiança é uma variável aleatória, podendo assumir valores diferentes para cada amostra. Para garantir níveis de confiança a esta variável aleatória, sua estimativa é realizada por um intervalo de valores que considera a variação da diferentes amostras e determina um *intervalo de confiança*.

Quando não é possível determinar a confiança do agente por meio das interações diretas, por conta do intervalo de confiança não atingir um valor mínimo necessário, os agentes devem utilizar as interações indiretas. Neste caso, a reputação é obtida pelo uso de testemunhas. O modelo evita, pelo menos em um primeiro momento, utilizar os testemunhos, considerados mais inseguros, visto que não há completa garantia a cerca da honestidade das testemunhas. TRAVOS considera o tratamento da *credibilidade* das testemunhas como um mecanismo essencial à qualidade das informações indiretas. Seu cálculo, também considerado como uma probabilidade é obtido pelos *feedbacks* dos próprios agentes que avaliam a precisão das testemunhas.

Uma limitação encontrada no modelo de confiança do TRAVOS é a ausência de mecanismos para tratar o peso das avaliações, como por exemplo, considerar avaliações recentes como mais relevantes em relação a avaliações antigas. Apesar disso, o TRAVOS é um dos melhores exemplos de modelos de confiança que utilizam o paradigma matemático para representar a confiança em comunidades virtuais.

2.6.8 Esfandiary e Chandrasekharan

O modelo de confiança de Esfandiary e Chandrasekharan, (2001) propõe a utilização de dois mecanismos para o tratamento da confiança. O primeiro é baseado na observação, onde é proposto o uso de *redes bayesianas*. Nos casos mais simples, para uma estrutura totalmente observável por uma rede bayesiana, as atividades de aprendizagem dos agentes resumem-se a meras considerações estatísticas.

O segundo mecanismo é o baseado nas interações entre os agentes. Nesta abordagem são utilizados dois protocolos de interação, o protocolo explorativo (*exploratory protocol*), onde o resultado das tarefas dos agentes é publicado de maneira que sua confiança possa ser calculada pela observação dos resultados, e o protocolo de consulta (*query protocol*), onde os agentes consultam as avaliações de outros agentes, semelhante ao mecanismo de testemunha. Uma forma simples de calcular a confiança baseada na interação direta (T_{inter}) de um agente A em relação a um agente B durante a fase exploratória é:

$$T_{inter}(A,B) = \frac{número de respostas certas}{número total de respostas} \quad (1)$$

Para tratar as informações recebidas das testemunhas, cada agente representa um vértice do grafo e suas arestas representam a confiança entre os agentes. Por exemplo, a aresta (*a*,*b*) representa a confiança que agente *a* tem sob o agente *b*. A Figura 2.1 ilustra a representa da confiança por meio de um grafo dirigido:

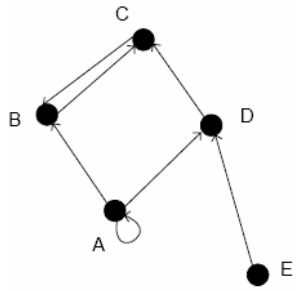

Figura 2.1 Grafo de confiança [Esfandiary & Chandrasekharan, 2001].

A ausência de aresta significa ausência de confiança. Neste grafo há possibilidade de ciclos que podem erroneamente aumentar o valor da confiança com diferentes caminhos contraditórios. Para solucionar este problema, ao invés de selecionar um único caminho para calcular o valor da confiança, são analisados todos os caminhos de forma a obter o menor e maior caminho não cíclico entre dois agentes.

Segundo Esfandiary e Chandrasekharan, (2001), a análise gráfica da confiança equivale à probabilidade de um roteamento em uma rede de computadores, assim os algoritmos de rede utilizados para resolução do roteamento podem ser aplicados ao problema do cálculo da confiança. Para o grafo adaptar-se à múltiplos contextos, os autores sugerem arestas coloridas, cada cor para cada tipo de contexto. Entretanto, nada foi dito sobre como combinar os diferentes mecanismos para aquisição da confiança.

2.6.9 Yu e Singh

Yu e Singh, (2000), propõe um mecanismo denominado de *social* para gerenciar a reputação de comunidades virtuais com o objetivo de evitar interações com indivíduos indesejáveis. Este mecanismo utiliza técnicas de segurança para garantir a *autenticação*, i.e., garantir que o indivíduo é quem ele diz ser e garantir que ele fará apenas o que lhe for permitido. Neste modelo, as informações armazenadas pelos agentes, a cerca de suas experiências diretas, possui uma semântica diferenciada em relação aos modelos de confiança apresentados até o momento. Confiança para Yu e Singh, (2000) significa um

nível de qualidade, i.e., a qualidade dos serviços prestados entre agentes consumidores e agentes fornecedores. A formalização desta nova semântica ao conceito confiança é dado pelo termo *qualidade de serviço* (*QoS*).

O cálculo da reputação e confiança ao modelo de Yu e Singh possui alguns princípios, que são: apenas a interação mais recente é considerada para o cálculo; cada agente define os valores limites (mínimo e máximo) que serão considerados para atribuir a qualidade de um serviço *QoS* aos seus parceiros; o cálculo da confiança utiliza a Teoria de Evidência de Dempster-Shafer[4] aplicada às informações histórias dos agentes para calcular a probabilidade/confiança de um agente sobre um determinado serviço; uma testemunha pode retornar dois tipos de informação, se ela conhecer diretamente o agente requisitado, retornará sua reputação, do contrário retornará um conjunto de referências a outras testemunhas que também poderão retornar os dois tipos de informação mencionados. O tratamento proposto as testemunhas resulta na denominada *TrustNet* ou rede de confiança.

O modelo de confiança de Yu e Signh não trata a combinação entre as informações diretas (interação direta) e as informações indiretas (testemunhas). A utilização das testemunhas é considerada apenas quando as informações diretas não conseguem avaliar com precisão a confiança. Outra observação ao modelo é a degradação de desempenho no tratamento das testemunhas em função do crescimento da comunidade. Sempre que alguém requisita a reputação de um agente desconhecido, todas as testemunhas são invocadas desnecessariamente, gerando um número excessivo de mensagens e desmotivando os agentes a compartilharem suas informações.

2.6.10 AFRAS

O modelo de confiança de Carbo, *et al.* (2002) é caracterizado pelo uso da lógica Fuzzy[5] para representar os valores de confiança. O *fuzzy set* do AFRAS, mostra o nível de satisfação das últimas interações para um dado parceiro, é calculado agregando avaliações antigas com os valores de avaliações recentes. A agregação destes valores com diferentes características é ajustada por meio de pesos. Os pesos utilizados para a agregação são calculados sobre um único fator chamado de *memory*. Por meio deste

[4] Dempster-Shafer também conhecida como a Teoria da Evidência é uma das técnicas ao tratamento de incertezas. Muito utilizada em sistemas de Inteligência Artificial que requerem raciocínio sob ambientes com informações incertas ou incompletas como os sistemas baseados em conhecimento. (Chen, 1986)

[5] Lógica difusa ou lógica fuzzy é uma derivação da lógica booleana que representa múltiplos valores lógicos entre os clássicos valores booleanos: Falso (0) e Verdadeira (1) é possível admitir infinitos valores entre estes intervalos. (Russell & Norvig, 1995)

fator é possível, por exemplo, atribuir maior relevância as avaliações mais recentes e menor relevância as mais antigas.

O fator *memory* é modelado como uma função que receber valores entre: o valor atual da reputação; o valor da última interação e o valor anterior do fator *memory*. Portanto, se o valor das experiências passadas for igual ao valor da última experiência, então a relevância da experiência passada é elevada. Se o valor da ultima experiência for diferente das experiências passadas, então a relevância da ultima experiência é elevada. A credibilidade para o valor de reputação também é representada por um *fuzzy set*, que indica o nível de incerteza da reputação.

As avaliações enviadas por outros agentes, como testemunhas, são agregadas diretamente com as experiências diretas. O peso dado às avaliações depende do nível de confiança da testemunha, assim quanto maior for a confiança da testemunha maior será a relevância de suas informações. Para calcular a confiança das testemunhas, os agentes comparam as recomendações com o real comportamento do agente avaliado.

2.6.11 Sen e Sajja

O trabalho de Sen a Sajja, (2002), apresenta um modelo de reputação que considera duas fontes de informação: a experiência direta e a experiência indireta ou testemunhos. A experiência direta é comprovadamente a fonte de informação mais precisa para o cálculo da confiança, diferente das fontes baseadas em observação ou testemunho que freqüentemente contém *ruídos*, i.e., informações que diferem do seu real valor. Portanto, a regra utilizada para atualizar o valor da reputação de um agente, quando há uma nova interação direta, possui maior relevância em relação a regra utilizada para atualizar a reputação quando há uma nova observação. Aqui, a *Aprendizagem por Reforço*[6] é considera uma alternativa ao mecanismo de atualização da reputação. O valor da reputação varia entre 0 e 1, onde o valor 0,5 é o marco para diferenciar uma boa reputação (> 0,5) de uma má reputação (< 0,5).

Outra característica deste modelo é a sua abordagem utilizada para compartilhar informações. Os agentes podem ser consultados sobre o desempenho de outros agentes. Semelhantes às testemunhas, eles devem retornar um valor booleano que indica se o agente consultado é “bom” ou não. O modelo também aborda o tratamento de agentes mentirosos na comunidade. Um agente é considerado mentiroso quando informa que

[6] Aprendizagem por Reforço é uma subárea da aprendizagem de máquina preocupada em como um agente pode agir sobre um ambiente de forma a melhorar suas percepções ao longo do tempo, por meio de um mecanismo de recompensa. (Claus & Boutilier, 1998)

outro agente possui bom desempenho quando ele não o possui ou vice-versa. Para detectar os agentes mentirosos, o modelo utiliza a quantidade de avaliações positivas e negativas recebidas das testemunhas. Por convenção é considerado mentiroso aquele que não responder conforme a maioria. Cada erro dos agentes é pontuado negativamente, caso ele possua certa quantidade de erros, será considerado malicioso ou mentiroso pelo modelo.

O modelo de Sen e Sajja é considerado um exclusivo modelo reputação. Os agentes utilizam apenas às informações de testemunhas para a seleção dos seus parceiros. As experiências diretas são utilizadas apenas como informações para serem compartilhadas entre os agentes da comunidade. Não há nenhuma mensão, neste modelo, no tocante a combinar os dois tipos de informação para calcular o valor final da reputação. O interesse é o estudo de como os agentes podem usar o popular "boca-a-boca" para selecionar os seus parceiros.

2.6.12 Castelfranchi e Falcone

O modelo de confiança proposto por Castelfranchi e Falcone, (1998) representa um dos principais exemplos de modelos de confiança cognitivos. A confiança neste modelo ganha semântica de *estado mental*, na qual afirma que "*Agentes BDI requerem princípios de confiança*". A principal tese defendida neste trabalho é a forte relação entre os conceitos de confiança e delegação. Em outras palavras, quando o agente *a* delega uma tarefa ao agente *b* ele possui um conjunto de "crenças" que o faz "confiar" que *b* é capaz de executar a tarefa e que ele estará disposto a fazê-la. Do ponto de vista de Castelfranchi e Falcone "apenas um agente cognitivo pode *confiar* em outro agente, portanto apenas um agente dotado de metas e crenças".

As relações entre confiança, delegação, meta e crença são representadas em um cenário multiagente: quando um agente *x* possui uma *meta m* a ser atingida e não é capaz de atingi-la sozinho, ele pode *delegar* uma ou mais tarefas ao agente *y*. Ao delegar suas tarefas, *x* utiliza o estado mental de confiança em *y* baseado em um conjunto de *crenças* básicas:

1. Crença em competência: *x* deve acreditar que *y* é capaz de auxiliá-lo em alguma tarefa, que pode fornecer os resultados esperados, que pode desempenhar um papel em seus planos ou que *y* tem alguma utilidade capaz de apoiar *x*.

2. Crença em disposição: *x* deve acreditar que *y* irá fazer a tarefa delegada, que está disposto a ajudá-lo. É uma crença em relação à vontade de *y*.
3. Crença em dependência: *x* acredita que *y*, dentre outras opções, é de fato necessário à execução da tarefa, ou que ele é a melhor opção para isso.
4. Crença em execução: *x* deve acreditar que a meta *m* será alcançada. Quando *x* confia em *y* para atingir *m* ele também confia que *m* será atingida.

O cálculo da confiança é representado por uma função que, a partir de um conjunto de crenças (competência, disposição, dependência e execução), representa uma *certeza* a cerca da confiança. O nível de confiança é utilizado para formalizar uma linha de raciocínio que decide se *y* será utilizado. A Figura 2.2 ilustra um cenário simplificado para decisão da confiança de *x* em *y*:

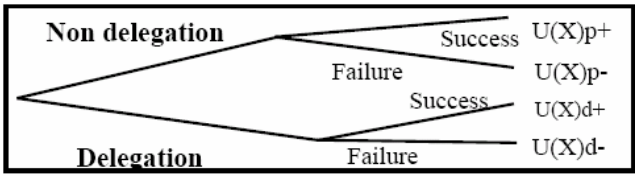

Figura 2.2 Cenário de escolha [Castelfranchi & Falcone, 1998].

Conforme ilustração há quatro situações para uma decisão de confiar ou não: *não delegação positiva*, quando *x* é auto-suficiente para atingir seus objetivos resultando em uma *função de utilidade positiva* ($U(X)p^+$); *não delegação negativa*, quando *x* não consegue sozinho atingir sua meta (utilidade negativa: $U(X)p^-$); *delegação positiva*, quando *x* depende de *y* e essa dependência prover uma função de utilidade positiva ($U(X)d^+$) e finalmente a *delegação negativa* quando a dependência de *y* resulta em uma função de utilidade negativa ($U(X)d^-$).

Para *x* maximizar suas chances de sucesso (utilidade positiva) o modelo de confiança considera a fórmula (1) abaixo.

$$DoT_{XY} \otimes U(X)d^+ + (1 - Fr_{XY_T}) \cdot U(X)d^+ > U(X)p^+ \qquad (1)$$

A fórmula (1) condiciona a confiança de *x* em *y* (DoT_{XY}) quando a função de utilidade positiva de em relação ao fato de risco (Fr_{XY}) é maior que a função de utilidade positiva quando *y* é delegado. Para definir a fator de risco foi usada a fórmula (2).

$$Fr_{XY_T} = (U(X)p^+ - U(X)d^-)/(U(X)d^+ - U(X)d^-) \qquad (2)$$

O fator de risco representa a razão entre a variação da utilidade quando *y* não é delegado e a variação de utilidade quando *y* é delegado.

Apesar da rica formalização do modelo de confiança cognitiva, Castelfranchi e Falcone não especificam como os agentes obtêm as informações para alimentar suas crenças.

2.7 Resumo Comparativo

A Tabela 2.3 apresenta um resumo comparativo dos modelos de confiança descritos neste capítulo em função da classificação apresentada na seção 2.4. As abreviaturas utilizadas são apresentadas na Tabela 2.4:

	Paradigma	Origem da Informação	Visibilidade	Dimensões	Comportamento	Tipo de Informação
E-commerce	M	T	G	UC	0	B
ReGret	M	ED +T	L	MC	2	B
FIRE	M	ED + T	L	MC	1	B
S. Marsh	M	ED	L	MC	NA[7]	N
Spora	M	T	G	UC	0	C
Abdul-Rahman e Stephen Hailes	M	ED + T	L	MC	2	B
TRAVOS	M	ED + T	L	MC	1	B
Esfandiary e Chandrasekharan	M	ED + OD + T	L	MC	0	B
Yu e Singh	M	ED + T	L	UC	0	B
AFRAS	M	ED + T	L	UC	2	B
Sen e Sajja	M	ED + OD + T	L	UC	2	B
Castelfranchi e Falcone	C	NA[8]	L	MC	NA[9]	NA

Tabela 2.3 Comparação dos modelos de confiança.

Característica	Abreviatura	Significado
Paradigma	M	Matemático
	C	Cognitivo
Origem de Informação	ED	Experiência Direta
	OD	Observação Direta
	T	Testemunhos
Visibilidade	L	Local
	G	Global
Dimensões	UC	Único contexto
	MC	Múltiplos contextos
Comportamento	0	Conforme seção 2.4.5
	1	
	2	
Tipo de Informação	B	Booleano
	C	Contínuo
	NA	Não utiliza testemunhas

Tabela 2.4 Abreviaturas

2.8 Considerações Finais

Neste capítulo foram tratados os principais conceitos de confiança e reputação, além de apresentar sua aplicabilidade em comunidades virtuais como as que utilizam uma arquitetura multiagente aberta. A fim de promover uma ampla visão sobre o tema,

[7] Não há troca de informações entre os agentes.
[8] Não é mencionado no modelo, como os agentes obtêm informações para construir suas crenças.
[9] O modelo não especifica como os agentes obtêm informações para detecção de agentes maliciosos.

foram também apresentados diversos modelos de confiança que abrangem as principais estratégias para a sua construção tais como: tipos de paradigmas, origens da informação, dependência de contexto, comportamento dos agentes, dentre outras características. Após análise descrita neste capítulo podemos fazer algumas considerações a cerca do tema.

Primeiramente, destacamos a importância dos modelos de confiança como um estruturador fundamental à construção de sistemas que tratam comunidades virtuais abertas, sejam elas compostas por pessoas, máquinas ou componentes de software. A presença de indivíduos desconhecidos nestas comunidades a tornam essencialmente instáveis quanto à segurança. Entretanto, os estudos mostram que a utilização de modelos de confiança pode minimizar os riscos entre as interações dos indivíduos e conseqüentemente garantir níveis de segurança toleráveis à evolução destas comunidades.

Percebe-se, na Tabela 2.3, que o paradigma matemático é predominante na modelagem de sistemas de confiança. Uma possível razão seria a histórica influência matemática sobre as principais áreas envolvidas: sistemas multiagente (computação) e e-commerce (economia). Outro motivador à abordagem matemática seria a alta complexidade na construção de modelos puramente cognitivos.

Quanto à origem da informação, destacamos as duas principais fontes utilizadas pelos modelos analisados: a experiência direta combinada com a experiência indireta por meio de testemunhas. Apesar da informação obtida pelo mecanismo direto ser a mais precisa do ponto de vista do agente, à medida que é o mesmo indivíduo que interage e a obtém diretamente o resultado desta interação. O mecanismo indireto possibilita maior desempenho aos sistemas, à medida que o reuso das informações (por meio do compartilhamento das testemunhas) obtidas pelo mecanismo direto, reduz o número de interações diretas e, conseqüentemente, aumenta a escalabilidade dos sistemas. Finalmente, o mecanismo indireto promove novos desafios como a seleção de testemunhas confiáveis, e a garantia de autenticidade das informações compartilhadas.

Acreditamos ser possível melhorar a eficiência dos atuais modelos de confiança, propondo uma nova estratégia aos mecanismos de interação direta e indireta que pode agregar as duas respectivas vantagens: precisão e desempenho. Outra contribuição essencial às comunidades abertas é a garantia da veracidade das informações trocadas sobre este ambiente por meio de um canal de comunicação seguro para troca de informações.

Capítulo 3

Métodos de Criptografia e Assinatura Digital

3.1 Considerações Iniciais

A utilização das redes de computadores por organizações para dirigir e ampliar seus negócios e o massivo crescimento da Internet, dão origem à necessidade de mecanismos mais sofisticados para prover a segurança das informações que trafegam nestas redes. Segurança é um tema de grande importância para qualquer sistema computacional, pois a possibilidade de um sistema ter suas informações expostas a ataques é uma realidade iminente. Isto é ainda mais agravante quando o sistema de informação está conectado a Internet. Novas vulnerabilidades são descobertas todos os dias e novos ataques surgem utilizando técnicas cada vez mais complexas a fim de violar a privacidade e a segurança dos canais de comunicação. Diante deste cenário, a segurança da informação tem ganhado maior relevância na administração de sistemas.

Uma das mais antigas técnicas para evitar o acesso indevido às informações privadas é o uso da criptografia. Além disso, surge o conceito das *assinaturas digitais* que permitem identificar com segurança a autoria de documentos eletrônicos. Conforme apresentado no Capítulo 2, o uso dos modelos de confiança, sob comunidades virtuais abertas, depende essencialmente da construção de canais de comunicação que garantam serviços básicos de autenticação, privacidade, confidencialidade e integridade das informações vinculadas a estas comunidades. Desta forma, o estudo de técnicas de segurança como criptografia e assinatura digital podem auxiliar na concepção de um modelo de confiança mais seguro.

3.2 Criptografia

A criptografia[10] é estudo de técnicas para transformar o conteúdo de informações legíveis em conteúdo ilegível a fim de evitar seu acesso por pessoas não autorizadas (Meyer, 1989). Por meio de tais técnicas é possível oferecer um ambiente seguro para armazenar, recuperar ou trafegar informações, seja internamente em um computador ou de maneira distribuída por uma rede de computadores. As diversas técnicas de criptografia podem ser agrupadas em duas classes: *criptografia por código* e *criptografia por cifra*. A Figura 3.1 apresenta a classificação completa das técnicas de criptografia.

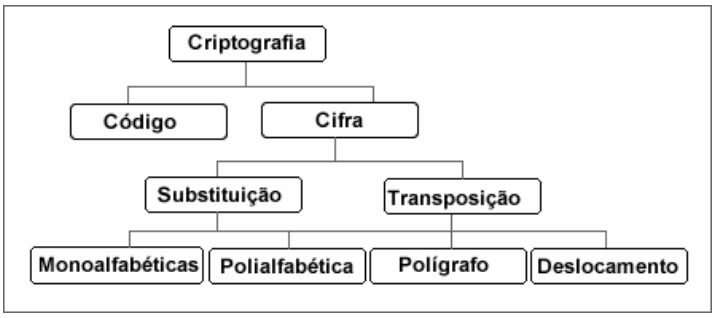

Figura 3.1 Classificação da criptografia.

A *criptografia por código* visa ocultar o conteúdo da informação a partir de códigos conhecidos entre o emissor e o receptor da mensagem. Usemos como exemplo um cenário bélico em que tropas possuem três opções estratégicas: avançar a leste desviando-se de um confronto direto, avançar a norte atacando o inimigo vizinho ou recuar a oeste voltando à base de origem. Foi acordado entre as tropas e o quartel general, que se enviada à mensagem "sol", o exercito tomaria a primeira opção, se enviada à mensagem "lua" seria a segunda opção ou se enviada à mensagem "mar" seria a terceira. Definidos os códigos, mesmo que a mensagem seja capturada por uma tropa inimiga seu significado não será descoberto.

A principal desvantagem da criptografia por código é o número limitado de mensagens que podem ser enviadas. Caso haja uma nova estratégia de guerra será necessário criar um novo código que deverá ser comunicado às partes interessadas.

As limitações da criptografia por código podem ser superadas pela *criptografia por cifra* que visa ocultar o conteúdo das informações por meio de operações de trocas

[10] Criptografia do grego, "kriptos" e "grifo" que significam "oculto" e "grafia", é a arte de escrever de maneira codificada, utilizando um conjunto de técnicas que permitem tornar a informação ilegível a partir de uma cifra ou código de acordo com um determinado algoritmo. (Meyer, 1989)

ou substituições de caracteres da mensagem original. Desta maneira é possível transmitir novas mensagens sem a criação de códigos. *Cifrar* uma mensagem significa aplicar um conjunto de operações sob o conteúdo original e *decifrar* significa aplicar o processo inverso sob a mensagem cifrada. Conforme apresentado na Figura 3.1 a cifra pode ser classificada como de *transposição* ou de *substituição*.

A *cifra de transposição* é o método de deslocamento dos caracteres sem modificá-los, a transposição apenas troca a posição das letras. Por exemplo, a palavra "MEIA" pode ser cifrada transpondo-a para "AMIE". O deslocamento de cada caractere é feito conforme um algoritmo predefinido. Um destes algoritmos é conhecido como *Cerca-de-ferrovia* (Singh, 2001), no qual cada linha da mensagem é dividida em duas linhas. As letras de posição impar ficam na primeira linha e as letras de posição par ficam sob a segunda linha. O exemplo a seguir ilustra o uso do algoritmo:

Mensagem original: VAMOSATACARAMANHAPELAMANHA

Mensagem cifrada: VMSTCRMNAEAAH

AOAAAAAHPLMNA

A *cifra de substituição* é o método baseado em operações de substituição dos caracteres por meio de um algoritmo ou tabela de substituição. A substituição pode ser feita trocando uma letra por outra, neste caso denominamos a técnica de *monoalfabética*. Esta técnica foi utilizada pelo Império Romano no qual substituía as letras de acordo com sua posição no alfabeto, exemplo, à letra "A" trocada por duas posições se torna a letra "C", "F" em "H" e assim por diante. Outra técnica de substituição consiste no uso de várias tabelas de troca, denominada de *polialfabética* onde para cada posição das letras é utilizada uma respectiva tabela de substituição. Outra técnica de substituição é feita por *polígrafos* que substituem, ao invés de uma letra, um grupo de letras, exemplo, o conjunto "BAN" é trocado por "ROC", "CAS" por "HUI" e assim por diante. Por último, temos a cifra de substituição por *deslocamento*, ao contrário da monoalfabética que utiliza um número de fixo para deslocar as letras, esta técnica utiliza critérios variados, exemplo, a palavra "AVIAO" sob o critério "1325" resulta no deslocamento de 1 posição à letra "A", 3 posições à letra "V", 2 posições à letra "I", a assim por diante.

A princípio de Kerckhoffs, (1983) afirma que a segurança de um sistema de criptografia não deve depender do seu algoritmo, mas de manter sua *chave* em segredo. *Chave* é um termo simbólico dado para seqüências de números e letras que associados a

um algoritmo de criptografia produzem mensagens cifradas. A Figura 3.2 ilustra esta relação:

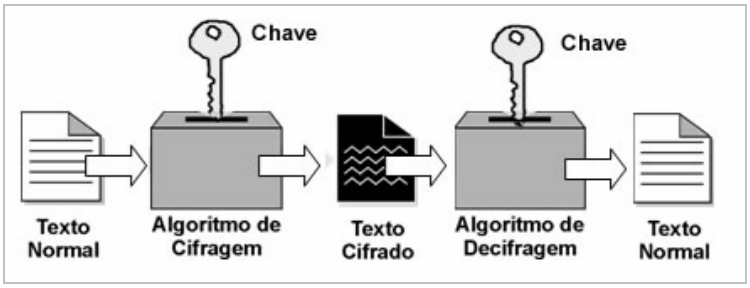

Figura 3.2 Cifragem baseada em chaves.

No caso da cifragem monoalfabética a chave representa o número de deslocamentos do alfabeto ou na cifragem polialfabética a chave são as tabelas de substituição.

A vantagem da criptografia por cifras em relação à criptografia por código é a ilimitada possibilidade de mensagens que podem ser transmitidas. Além disso, é possível aumentar o grau de complexidade do processo de cifragem por meio de algoritmos tornando-o cada vez mais seguro. A utilização de algoritmos associados a chaves é atualmente a abordagem mais recomenda para a construção de canais de comunicação seguros. As duas próximas seções apresentam os dois principais tipos de algoritmos de cifragem baseados em chaves.

3.3 Algoritmos Simétricos

A criptografia de uma mensagem pode ser bastante simples. Por exemplo, na cifragem por substituição se recebermos a informação cifrada: “ZCOQU CVCECT” e soubermos que a chave é “2” e que o algoritmo de criptografia substitui as letras conforme sua posição alfabética será fácil descobrir que a mensagem original é “VAMOS ATACAR”, entretanto os atuais sistemas de criptografia utilizam alto grau de segurança, onde as chaves são compostas por centenas de bits e utilizam diferentes tipos de algoritmos exigindo milhares de operações matemáticas para cifrar e decifrar estas mensagens. A complexidade utilizada nos atuais processos de criptografia torna praticamente impossível decifrar uma mensagem sem conhecer a sua chave.

Os *algoritmos simétricos* ou também chamados de *algoritmos de chave-simétrica* ou *criptografia de chave única* são aqueles que utilizam a mesma chave para

cifrar e decifrar as mensagens (Simmons, 1979). A Figura 3.3 ilustra os processos de cifragem e decifragem por algoritmos simétricos.

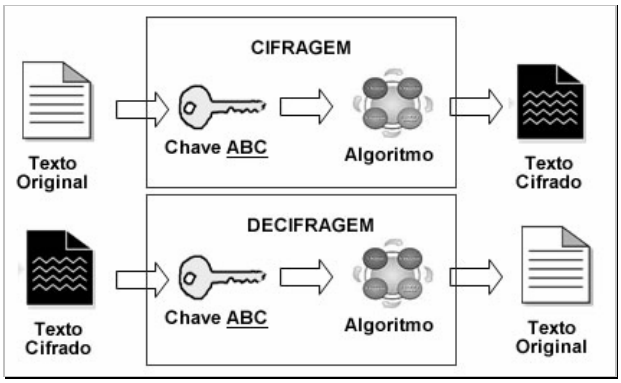

Figura 3.3 Cifragem e decifram por algoritmo simétrico.

A maioria dos algoritmos de chave simétrica utiliza chaves de 64 bits como o Twofish (Schneier, *et al*, 1999), Blowfish (Lin & Lin, 2000), CAST5 (Lee, *et al.*, 1997), RC4 (Schweitzer & Baird, 2006), TDES (Huiping, *et al.,* 2007), e IDEA (Hoffman, 2007). Outros algoritmos utilizam 128 bits ou mais como o AES (Lu & Tseng, 2002) e Serpent (Elbirt & Paar, 2000).

A principal desvantagem da criptografia simétrica é o compartilhamento das chaves secretas. Se um segredo é conhecido por mais de um indivíduo ele deixa de ser segredo. A distribuição das chaves secretas entre os emissores e receptores permite que terceiros possam, devido a um descuido ou propositalmente, capturar a chave secreta destruindo por completo o sigilo das informações.

3.4 Algoritmos Assimétricos

Para evitar o compartilhamento das chaves secretas no processo de criptografia, surgem os algoritmos assimétricos ou também conhecidos como algoritmos de chaves assimétricas (Simmons, 1979). Neste processo é necessário um par de chaves para realizar a cifragem e decifram das mensagens. A primeira chave é denominada de *privada*, ela é de posse exclusiva de seu detentor e ninguém mais a conhece. A segunda chave do par e denominada de *pública* e pode ser enviada a qualquer indivíduo.

O par de chaves é construído por meio de funções matemáticas unidirecionais que tornam seu relacionamento único, as chaves são geradas em função do seu par, portanto uma chave privada está relacionada exclusivamente a uma chave pública e vice-versa. Por conta do mecanismo de criação do par de chaves é possível cifrar um documento com uma chave privada e garantir que apenas a respectiva chave pública

poderá decifrar o documento. O processo inverso também é verdade, se o documento é cifrado por uma chave pública, apenas a respectiva chave privada poderá decifrar o documento. Portanto, na criptografia assimétrica as chaves públicas e privadas são construídas aos pares e seu uso apenas é possível se também for realizado aos pares, pois uma chave depende da outra e nenhuma delas possui utilidade sem a outra.

Apesar da existência de uma chave pública parecer um problema de segurança, na medida em que qualquer pessoa pode decifrar as mensagens com esta chave, a especificação do algoritmo assimétrico define muito bem o tratamento da confidencialidade das informações. Quando alguém cifra um documento com sua chave privada ele não está preocupado na confidencialidade já que qualquer pessoa pode decifrar o documento com sua chave pública. O interesse da cifragem por chave privada é a garantia da *autenticidade* das informações, i.e., ter a certeza de que o emissor de fato é quem diz ser, pois se o documento pode ser decifrado com sua chave pública significa que ele foi cifrado com a respectiva chave privada que apenas é conhecida pelo emissor. A Figura 3.4 ilustra os processos de cifragem e decifragem por chaves assimétricas e exemplifica a garantia de autoria dos documentos criptografados.

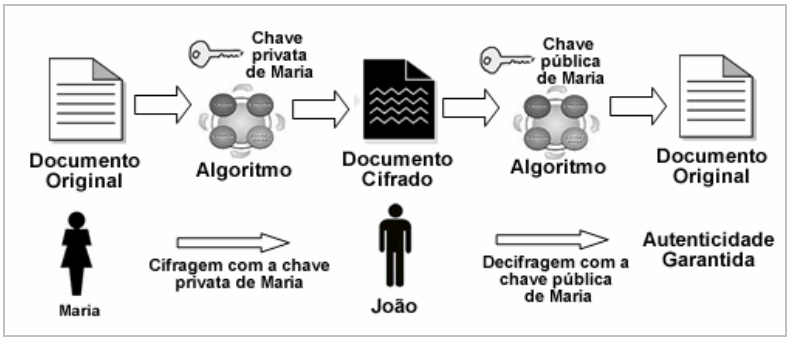

Figura 3.4 Autenticidade da criptografia assimétrica.

Se a autenticidade das mensagens é garantida por meio da criptografia com a chave privada do emissor, a *confidencialidade* é garantida pela criptografia com a chave pública do destinatário. Quando o emissor deseja enviar uma mensagem confidencial, ele deve cifrá-la com a chave pública do destinatário, pois apenas a chave privada é capaz de decifrar uma criptografia de chave pública. Caso a mensagem seja capturada por terceiros estes não poderão decifrá-la, pois a chave privada é de conhecimento exclusivo do destinatário. A Figura 3.5 ilustra a processo descrito de confidencialidade das informações.

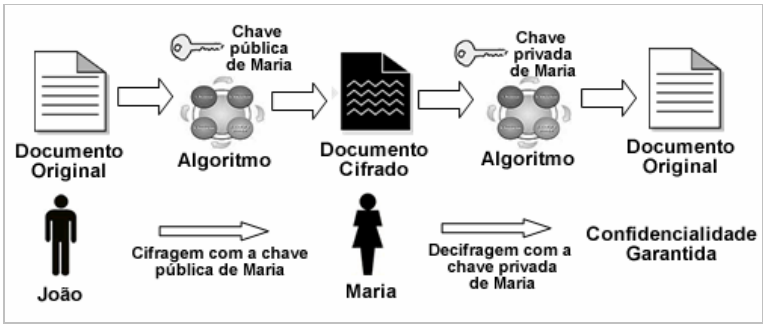

Figura 3.5 Confidencialidade da criptografia assimétrica.

Tanto a autenticidade quanto a confidencialidade dos documentos eletrônicos é garantida por meio do par de chaves assimétricas. Para isto, basta cifrar os documentos com a chave privada do emissor e enviá-los a qualquer destinatário que possua a respectiva chave pública e cifrar o documento com a chave pública do destinatário para que seja mantido o sigilo das informações. Além dos destinatários terem a certeza de que o documento foi emitido pelo emissor, eles também têm a garantia de que o documento não foi alterado durante sua emissão, visto que se um único bit do documento for alterado os emissores não conseguiram decifrar o documento.

Os algoritmos de chave assimétrica são responsáveis por definir como o par de chaves será gerado e como aplicar a criptografia e decifragem dos documentos. Atualmente, há diversos algoritmos utilizados na indústria tais como: *Data Encryption Standard* – DEA (Van Buren, 1990) utilizado pelo governo americano na década de 70, mas atualmente considerado inseguro; o PGP, abreviação de *Pretty Good Privacy* (Branagan, *et al.*, 1996), algoritmo que tem ganho popularidade nos últimos anos se tornando um padrão para cifragem de mensagens de correio eletrônico; o *Digital Signature Algorithm* - DSA (Laih & Yen, 1995) que utiliza os pontos de curvas elípticas para criptografar mensagens e o RSA (Rivest, *et al.*, 1978) pioneiro quanto a possibilitar a criptografia e assinatura digital, considerada uma das melhores implementações de algoritmo de chave assimétrica.

A principal vantagem dos algoritmos assimétricos em relação aos simétricos é o sigilo (pelo não compartilhamento) de suas chaves secretas, promovendo maior segurança aos canais de comunicação, obviamente que esta segurança dependerá da forma como as chaves privadas são protegidas. Como a criptografia assimétrica utiliza algoritmos mais complexos, seu processo é muito mais lento do que os algoritmos simétricos (Silva, *et al.*, 2008). "Dentre os algoritmos criptográficos robustos conhecidos, os assimétricos são mais lentos 10^3 a 10^4 vezes que os simétricos"

(Rezende, 1998). Entretanto, não precisamos criptografar um documento inteiro para garantir sua autenticidade, basta que haja uma forma de comprovar que esse documento foi de fato gerado pelo verdadeiro emissor. A próxima seção discute soluções para garantir a autenticidade dos documentos sem criptografá-los por completo.

3.5 Assinaturas Digitais

A seção anterior apresentou o método de criptografia baseado em chaves assimétricas que utiliza um par de chaves no qual a chave privada garante a autenticidade e a chave pública garante a confidencialidade das informações criptografadas. Além disso, foi afirmado que os algoritmos de chave assimétrica são lentos em relação aos algoritmos simétricos, entretanto quando se deseja apenas a autenticidade das informações não é necessário criptografá-las, mas apenas utilizar um mecanismo que comprove sua autoria. Assim, surge o conceito de *assinatura digital* ou identidade digital que visa, analogamente a uma assinatura em papel, garantir a autoria de documentos eletrônicos.

O processo de assinatura de documento parte da seguinte idéia: se a criptografia de um documento inteiro leva muito tempo, então podemos "resumir" seu conteúdo e criptografar apenas seu resumo. A relação entre o resumo e o documento original deve ser única, portanto se um bit do documento ou um bit do resumo for alterado a relação será invalidada. O mapeamento de uma informação de valor variável para outra informação de tamanho fixo é chamado de função *hash* (Preneel, *et al.*, 1993). Se a função *hash* for conhecida pelos emissores e destinatários, o resumo criptografado com a chave privada do emissor pode ser descriptografado com sua chave pública. O resumo decifrado pode ser comparado com a função *hash* do documento recebido, havendo compatibilidade entre o resumo e o documento, considera-se o documento válido.

Para simplificar o entendimento a cerca do resumo criptográfico, podemos comparar a função *hash* como uma função de *dígito verificador*. Os dígitos verificadores de um CNPJ, por exemplo, são calculados em função dos doze primeiros números, desta forma se algum dos doze números for alterado inclusive se o próprio dígito for modificado a função de dígito verificador acusará inconsistência entre os números e os invalidará. Analogamente, o resumo criptográfico de um documento eletrônico representa os "dígitos verificadores" deste documento. Se houver alteração no documento eletrônico ou em seu resumo a função *hash* identificará esta modificação e invalidará o documento recebido. Portanto, o emissor deve enviar o documento

juntamente com o seu respectivo *hash* que será criptografado com a chave privada do emissor. Desta forma, o destinatário receberá o documento, decifrará o *hash* com a chave pública do emissor, calculará o valor *hash* do documento recebido e comparará o valor *hash* calculado contra o valor *hash* recebido, se os dos valores foram iguais o destinatário estará certo de que o documento foi gerado pelo emissor. A Figura 3.6 representa os mecanismos de envio e recebimento de documentos eletrônico assinados digitalmente.

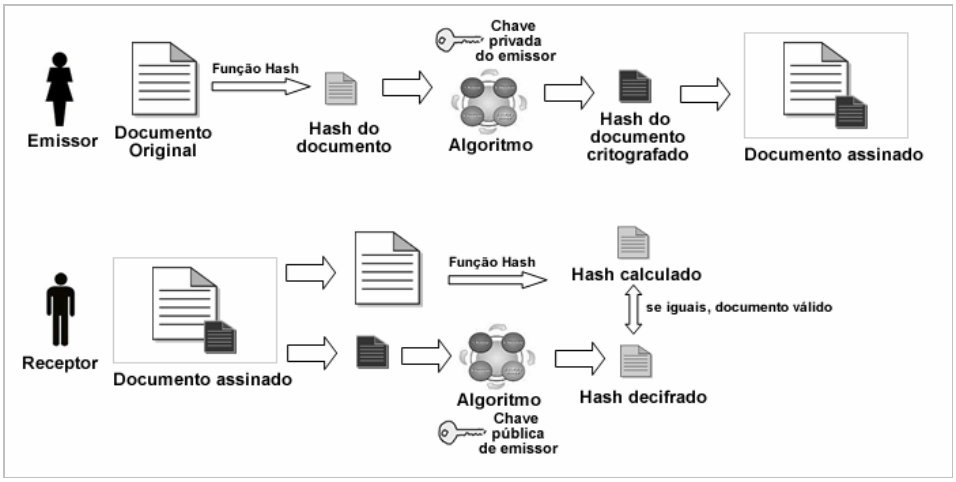

Figura 3.6 Envio e recebimento de documentos assinados.

Como o processo de criptografia é aplicado apenas ao valor *hash* do documento, que é objeto de tamanho consideravelmente pequeno, elimina-se o problema de desempenho dos algoritmos assimétricos. O uso da função *hash* também permite garantir a integridade do documento, além de garantir sua autoria. O resumo criptográfico funciona analogamente a uma assinatura, este conceito é denominado de *assinatura digital.* A Figura 3.7 ilustra a relação dos documentos assinados em papel e os documentos assinados digitalmente.

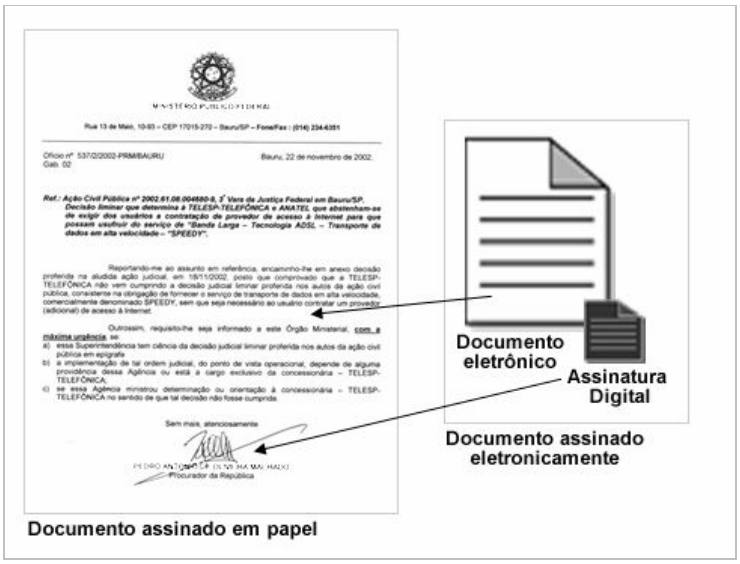

Figura 3.7 Analogia entre a assinatura física e a assinatura eletrônica.

É possível identificar similaridades entre os dois mecanismos de assinatura. Na assinatura eletrônica, o emissor produz um arquivo, seja um texto, uma apresentação, uma planilha eletrônica, um relatório ou qualquer outro documento. A partir deste documento eletrônico é criado um valor *hash* que será criptografado com a chave privada do autor. Na assinatura em papel o autor produz o documento e insere sua assinatura manuscrita que possui características únicas provendo um método seguro de autoria.

A autenticidade de uma assinatura física pode ser verificada por meio da comparação de outra assinatura registrada em um cartório, este processo também é chamado de reconhecimento de *firma*. No caso da garantia de integridade do documento em papel, é possível realizar exames físicos visando encontrar alguma evidência de fraude como falhas de impressão, manchas, rasuras entre outras verificações por meio de laboratórios de análise documental especializados.

Portanto, além da autoria é possível atestar a integridade de um documento físico por meio de exames feitos por organizações confiáveis. Analogamente ao processo físico é possível reconhecer as assinaturas digitais e a integridade dos documentos. Como já explicado anteriormente, a integridade é garantida pelas funções *hash* e a autoria pela criptografia do valor *hash* com a chave privada do autor.

Atualmente, não há restrições técnicas que impeçam a autenticidade e a integridade de documentos eletrônicos. Entretanto, do ponto de vista jurídico é necessária a criação de uma legislação que permita legalmente a assinatura de

documentos eletrônicos. Cada país possui autonomia para definir uma legislação própria ao tratamento das assinaturas digitais. No Brasil foi criada a Medida Provisória 2.200 [MED01] que valida juridicamente documentos eletrônicos assinados em meio digital atestando sua integridade e autenticidade.

Voltando ao ponto de vista técnico, resta apresentar a última questão a cerca das assinaturas digitais para viabilizar de fato seu uso: como validar uma assinatura eletrônica? Até o momento vimos que a assinatura garante a autoria do documento, mas isso não garante a autenticidade da própria assinatura. Tomando como referência os sistemas de autenticação de documentos em papel, além de um documento físico estar assinado pelo autor é necessário reconhecer sua assinatura em um *tabelionato de notas*. O tabelionato por sua fez precisa estar regulamente vinculado a um sistema de cartórios estabelecido por lei. Desta forma, qualquer cidadão pode registrar sua assinatura junto a um cartório e emitir documentos assinados e autenticados.

A próxima seção apresenta a solução técnica dada à construção de "cartórios virtuais" que análogos aos cartórios reais tornam possível a construção de uma infra-estrutura de gestão para as assinaturas digitais.

3.6 Infra-estrutura de Chaves Públicas (ICP)

Além de validar juridicamente os documentos eletrônicos assinados digitalmente, a Media Provisória 2.200 estabelece todo o sistema legal para construção de um conjunto de ferramentas e processos para a operação de um sistema de emissão de certificados denominado de *Infra-estrutura de Chaves Públicas Brasileira – ICP-Brasil* (Rolt *et al.*, 2006).

> *Art. 2o A ICP-Brasil, cuja organização será definida em regulamento, será composta por uma autoridade gestora de políticas e pela cadeia de autoridades certificadoras composta pela Autoridade Certificadora Raiz - AC Raiz, pelas Autoridades Certificadoras - AC e pelas Autoridades de Registro - AR.[MED01]*

A *Autoridade Certificadora Raiz (AC Raiz)* é o órgão fiscalizador das políticas de certificação digital. Ela é responsável por credenciar outras Autoridades Certificadoras (AC) por meio da emissão de certificados próprios às ACs. Todas as ACs devem ter seu certificado expedido por uma Autoridade Certificadora Raiz. Como

exemplos de AC Raiz temos a americana VeriSign[11], a britânica Equifax[12] e instituída pelo governo brasileiro, a AC Raiz da ICP-Brasil[13].

As Autoridades Certificadoras, hierarquicamente abaixo da AC Raiz, são responsáveis por credenciar usuários (pessoas físicas e jurídicas), emitir, revogar, distribuir e gerenciar os certificados. Como os certificados têm validade, a AC também mantém o registro dos certificados revogados e divulga essa lista pela Internet. Portanto a AC executa as principais funções de gestão dos certificados, mas ela não é responsável pelo cadastro dos usuários finais. Compete a *Autoridade de Registros* (AR) atestar a veracidade dos documentos recebidos pelos usuários, como comprovante de endereço, CPF, RG, Título de Eleitor e manter esse cadastro. Portanto a Autoridade de Registros faz a interface direta entre o usuário e a Autoridade Certificadora. A Figura 3.8 representa a estrutura hierárquica da ICP-Brasil.

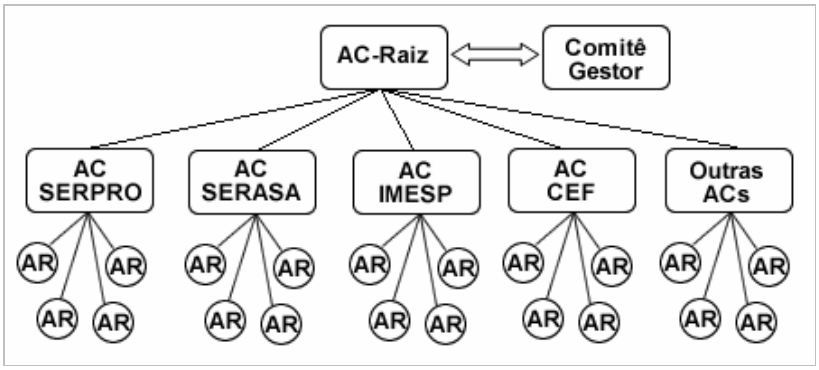

Figura 3.8 Infra-estrutura de chave pública brasileira (ICP-Brasil).

O *Comitê Gestor* define o conjunto de normas que será fiscalizado pela Autoridade Certificadora Raiz. Além disso, a AC Raiz credenciar um conjunto de Autoridades Certificadoras, normalmente pessoas jurídicas como o SERPRO, SERASA, Caixa Econômica Federal, entre outras organizações responsáveis por emitir os certificados digitais e manter sua gestão. O controle cadastral dos usuários é feito pelas *Autoridades de Registro* (AR) credenciadas por uma AC.

O certificado digital é um documento eletrônico que pode ser gravado em um dispositivo de armazenamento. Há dois tipos de certificados: A1 e A3. O certificado A1 é representado por um arquivo armazenado no computador do usuário, não há garantia de segurança quanto ao seu uso, à medida que qualquer indivíduo com acesso físico ao computador poderá utilizá-lo. O certificado A3 é armazenado em dispositivos

[11] VeriSign, site http://www.verisign.com.br/
[12] Equifax, site http://www.equifaxsecure.co.uk/
[13] ICP-Brasil, site http://acraiz.icpbrasil.gov.br/

eletrônicos como *smart card* ou *tokens usb* que criptografam o certificado provendo maior segurança. A estrutura dos certificados é composta pelo seguinte conjunto de informações (Rolt, *et al.*, 2006):

- Identificação do proprietário e endereço;
- Chave pública do proprietário;
- Validade do certificado;
- Número de série;
- Identificação da AC que emitiu o certificado;
- Assinatura digital da AC;
- Extensão.

Os aplicativos, que utilizam as assinaturas digitais para trafegar informações sob uma rede de computadores, devem suportar operações de criptografia, decifração de informação, além de interagir com os dispositivos de armazenamento dos certificados.

3.7 Considerações Finais

Neste capítulo foram apresentados os principais métodos de criptografia utilizados para a construção de canais de comunicação que garantam autenticidade, confidencialidade e integridade das informações trafegadas nestes canais. Atualmente o conceito de assinatura digital é considerado o método de criptografia mais seguro para garantir tais características aos canais de informação.

O crescimento da Internet e a necessidade da legalização de transações virtuais têm provocado discussões no âmbito legislativo em vários países. No caso do Brasil, a Medida Provisória 2.200 tornou juridicamente possível assinar documentos eletrônicos viabilizando a legitimidade de transações virtuais. No âmbito técnico, a criação da ICP-Brasil garantiu a infra-estrutura tecnológica necessário ao gerenciamento dos certificados digitais.

Os diversos aplicativos que tão suporte as chamadas *conexões seguras* como navegadores de internet, clientes de correio-eletrônico e os sistemas de informação com *homebank*, *e-commerce*, entre outros, mostram a viabilidade do mecanismo de assinaturas a todo e qualquer sistema que necessite utilizar um ambiente seguro para troca de informações. Finalmente, podemos concluir que o método de criptografia por assinatura digital pode ser aplicável a comunidades virtuais de agentes. Por conta da falta de garantias, quanto à procedência dos agentes integrantes de uma comunidade

virtuais abertas, usufruir-se de infra-estrutura de chaves pública pode auxiliar na construção de um modelo de confiança seguro.

Capítulo 4

CRONOS – Sistema Avaliador de Modelos de Confiança

Após ter descrito, no Capítulo 2, como os Modelos de Confiança podem reduzir o risco de interações entre agentes e, no Capítulo 3, como a criptografia das informações pode auxiliar a segurança dos sistemas, este penúltimo capítulo apresenta o Modelo de Confiança Certificado (Botelho, et al., 2009b), proposto nesta dissertação como uma alternativa aos clássicos problemas encontrados nos atuais modelos de confiança. Além disso, é apresentado um sistema para avaliação de modelos de confiança denominado CRONOS[14] utilizado para avaliar o modelo de confiança certificado proposto nesta dissertação.

4.1 Considerações Iniciais

Apesar da sua arquitetura aberta, o CRONOS utiliza uma plataforma de acesso segura baseada em assinaturas digitais. A construção deste projeto foi motivada pela necessidade de se ter um ambiente capaz de prover experimentos para avaliação de diferentes modelos de confiança. Para avaliar o modelo de confiança certificado, o CRONOS foi aplicado a um problema do mundo real, mais especificamente do mercado financeiro. Neste contexto foi proposto um conjunto de agentes que se assemelham a um *clube de investimento*[15]. Nele cada agente pode obter recomendações de compra e venda de seus parceiros. O parceiro pode ser qualquer agente que se disponha a

[14] CRONOS – (do grego *Κρόνος*) Deus de tempo da mitologia grega, correspondente ao deus romano Saturno. Por se tratar de um sistema que lida com as incertezas do futuro acionário nada mais sugestivo que utilizar o nome do Deus do tempo.

[15] Clube de Investimento – Associação de pessoas que se juntam para investir na bolsa de valores e compartilhar recursos e conhecimento a fim de obter melhores resultados coletivos. Conceitos referentes ao mercado financeiro são descritos no Apêndice A.

compartilhar informações e auxiliar outros agentes à tomada de decisão. A seleção dos parceiros é tratada pelo modelo de confiança que tem o objetivo indicar os parceiros mais assertivos em suas recomendações. Apesar deste trabalho tratar um domínio de problema do mercado financeiro, sua arquitetura foi projetada para adaptar-se a demais classes de problema.

No trabalho de Botelho, *et al.*(2009a), por exemplo, sua arquitetura foi utilizada para avaliar a qualidade de serviços em *Web Service*. Sua flexibilidade de uso é condicionada a maneira como os agentes são representados no sistema. Esta representação deve prever duas classes de agentes: provedores e consumidores/clientes de serviços. Sistemas distribuídos como *Grid Computing* (Oliveira, *et al*, 2007), comércio eletrônico, computação peer-to-peer, podem ser incorporados ao CRONOS desde que haja interações entre esses dois grupos de agentes.

Sistemas distribuídos que possuem riscos de interoperabilidade entre seus componentes, semelhantes ao CRONOS, podem ser considerados ambientes propícios ao uso de modelos de confiança. Desta maneira, podemos utilizá-los como uma ferramenta de análise, uma vez que o desempenho de um modelo de confiança pode ser medido em função do desempenho global do sistema (Jurca & Faltings, 2006). Portanto, o sistema CRONOS tem como objetivo principal prover um ambiente experimental capaz de analisar, medir e representar o desempenho de um modelo de confiança. Por meio dele é possível comparar a qualidade do modelo de confiança certificado contra as principais abordagens existentes.

4.2 Modelo de Confiança Certificado

Classicamente, um modelo de confiança envolve um agente *a* que quantifica a confiança que ele tem com relação a outro agente *b*, i. e., o agente *a* é o um avaliador (*evaluator)* e o agente *b* é o alvo da avaliação (*target)*. Uma avaliação (*rating*) é calculada a partir de uma experiência passada entre dois agentes. Cada *rating* é representado por uma tupla $r=(a, b, i, v, c)$, onde *a* e *b* são agentes participantes de uma interação *i* e *v* é o valor da avaliação realizada por *a* sobre *b* em um determinado termo *c*. Cada avaliação é armazenada localmente pelo agente avaliado. Este último poderá, quando inquirido por um agente cliente, repassar as suas avaliações ao requerente. O termo *c* garante ao modelo de confiança a expressividade de avaliar cada agente em um contexto diferente, i.e., cada avaliação é atribuída para um termo específico como, por exemplo, tempo de resposta, custo, complexidade entre outros. A notação de confiança

de *a* em *b* para um termo *c* é denota por *T*(*a, b, c*). O cálculo da confiança *T* requer um conjunto de avaliações relevantes. Este conjunto é denotado por *R*(*a, b, c*). Estas definições serão úteis para a representação dos modelos de confiança tratados a seguir neste trabalho.

O *Modelo de Confiança Certificado* que propomos sugere um tratamento diferenciado aos *ratings* produzidos pelos agentes consumidores (ou clientes), i.e., o *rating* de um agente avaliado, classicamente calculado, armazenado e compartilhado pelos agentes clientes (no *modelo indireto*), aqui é tratado localmente pelo próprio agente alvo (*provedor* do serviço). A Figura 4.1 ilustra o tratamento dos *ratings* pelos três principais tipos de modelo de confiança e apresenta o tratamento do modelo de confiança certificado:

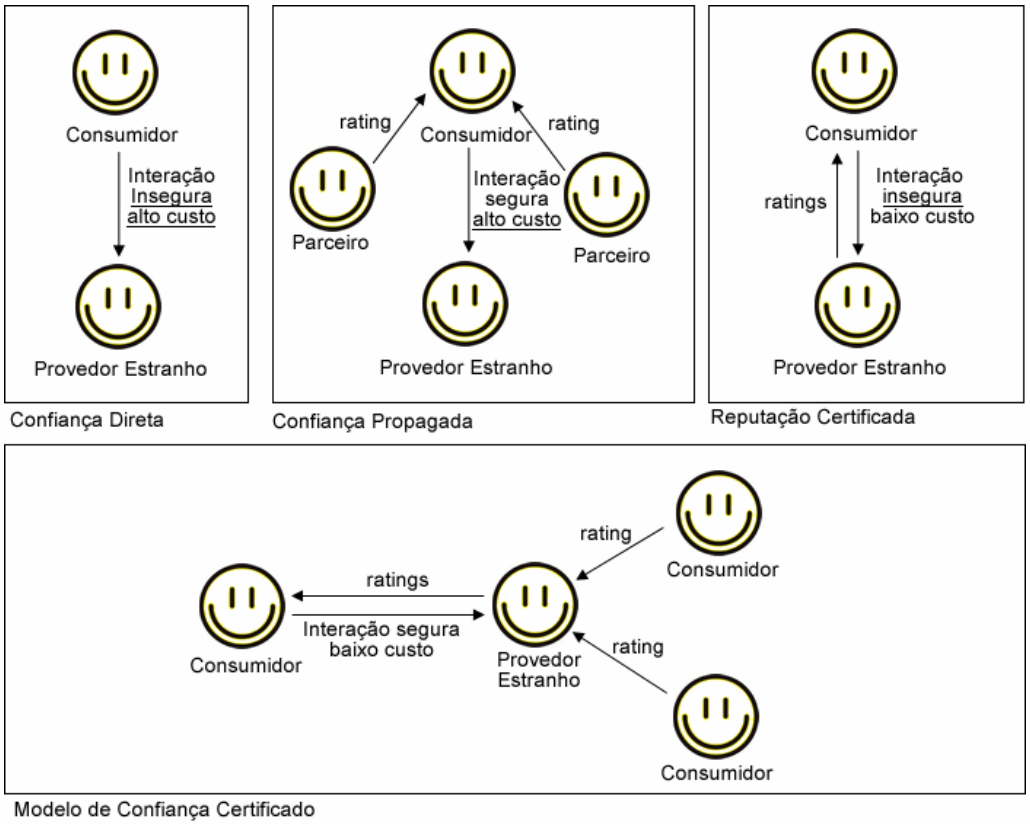

Figura 4.1 Tratamentos dos *ratings* pelos modelos de confiança.

A abordagem da confiança certificada visa reduzir as limitações dos modelos de confiança baseados, respectivamente, na experiência que resulta da interação direta entre os agentes, *confiança direta*; na experiência indireta obtida por meio dos relatos de testemunhas, *confiança indireta,* e na abordagem de *reputação certificada* (Huynh, *et al.*, 2006). A confiança direta tem baixo desempenho vis-à-vis à dificuldade de um

agente cliente *a* realizar um número de interações com um agente alvo *b* para produzir uma base de experiências significativa. A confiança propagada depende do altruísmo das testemunhas para compartilhar suas experiências. A reputação certificada, apesar de tratar os dois problemas anteriores, permite que os agentes mascarem sua reputação fornecendo a outros agentes um valor superestimado, calculado com base apenas nas melhores avaliações recebidas. Conforme a Figura 4.1, o modelo de confiança certificador (nossa proposta) promove um mecanismo de interação seguro, à medida que as avaliações são assinadas (assunto tratado na seção 4.2.2) provendo integridade e autenticidade das informações. E o baixo custo do mecanismo é herdado pela reputação certificada na qual o número de mensagens trocadas para obtenção de testemunhos envolve apenas os dois agentes, um consumidor e um provedor.

4.2.1 Cálculo da Confiança

Ao término de uma interação *i*, o agente alvo *b* solicita ao seu parceiro *a* que avalie seu desempenho *v* para um determinado termo *c*, que resulta em um *rating* *r*=(*a,b,i,v,c*). O agente *b* armazena o *rating* em seu repositório local. Quando um agente cliente *a* informa o seu interesse por um serviço *c* de um agente alvo *b*, *b* responde repassando os seus *ratings* mais relevantes *R*. Esta abordagem reduz o problema do agente avaliador em recusar-se a compartilhar suas experiências, pois o avaliador é inquirido uma única vez a dar sua recomendação sobre *b* e a obtenção da informação necessária para o cálculo da confiança envolve apenas dois agentes.

O cálculo da confiança é dado pela média ponderada de todos os *ratings* retornados pelo agente alvo. Cada *rating* é associado a um peso que varia em função do tempo, quanto mais recente o *rating*, maior o seu peso. O cálculo do peso para um *rating r* em função do tempo é denominado por $\omega(r_i)$, *sendo* $(\omega(r_i) \geq 0)$. Portanto, o cálculo da confiança de um agente *a* em relação a um agente *b* para um termo *c* é definido pela formula a seguir:

$$\mathrm{T}(a,b,c) = \frac{\sum_{r_i \in R(a,b,c)} \omega(r_i) \cdot v_i}{\sum_{r_i \in Rc(a,b,c)} \omega(r_i)} \qquad \textbf{(1)}$$

Para exemplificar o cálculo de *T(a,b,c)*, tomemos a Tabela 4.1 que registra o conjunto de dados utilizados pelo agente *a* para avaliar a confiança do agente *b*. Considere o *tempo* em dias e valor de cada *rating* (Coluna: Valor, v_i), variando entre a faixa de [-1, +1].

Interação	Tempo(Δt)	Valor(v_i)	Peso($\omega(r_i)$)	$\omega(r_i) \cdot v_i$
1	0	0,90	1,00	0,90
2	3	0,70	0,69	0,48
3	7	0,80	0,41	0,33
4	*9*	0,50	0,32	0,16
5	*15*	-0,50	0,15	-0,08
6	*17*	0,94	0,12	0,11
7	*18*	-0,30	0,10	-0,03
8	*20*	0,00	0,08	0,00
9	*25*	0,40	0,04	0,02
10	*30*	0,00	0,02	0,00
Somatório	-	-	2,94	1,89

Tabela 4.1 Interações do agente avaliado *b*.

Cada linha da tabela representa uma interação do agente *b* com os demais agentes do sistema. A confiança é calculada a partir da razão entre o somatório da coluna $\omega(r_i) \cdot v_i$ versus a coluna $\omega(r_i)$, conforme a fórmula (1).

$$\mathrm{T}(a,b,c) = \frac{\sum_{r_i \in R(a,b,c)} \omega(r_i) \cdot v_i}{\sum_{r_i \in Rc(a,b,c)} \omega(r_i)} = \frac{1,89}{2,94} = 0,64$$

A coluna Peso, definida por $\omega(r_i)$, representa o grau de relevância das avaliações. Há vários fatores que podem influenciar o valor do peso como, a procedência do agente avaliador, tempo de vida da avaliação ou outras características especificas das interações. Para simplificar o processo da análise e cálculo da confiança dos agentes, propomos apenas o tempo como fator relevante para mensurar o grau de relevância de um *rating*. As formulas 2 e 3 definem o cálculo do peso de cada *rating*:

$$\omega t(r_i) = e^{-\frac{\Delta t(r_i)}{\lambda}} \qquad (\mathbf{2})$$

$$\Delta t(r_i) = t_{ia} - t_{i0} \qquad (\mathbf{3})$$

Onde $\omega t(r_i)$ representa o peso do *rating* r_i em função do tempo. A variável $\Delta t(r_i)$ representa o intervalo de tempo entre o momento atual t_{ia} e o momento em que o *rating* r_i foi criado t_{i0}. A constante λ representa o coeficiente de velocidade para o decremento da função. Assim, à medida que Δt aumenta, o valor do peso decresce com uma velocidade λ. A definição desta formula é motivada pela premissa de que as últimas avaliações recebidas por um agente representam mais fielmente seu atual

comportamento. Desta forma, os *ratings* mais recentes, i.e., que possuem menores Δt são mais significantes para o cálculo da confiança de um agente.

Para exemplificar o uso da Formula 2, continuaremos com o exemplo dos agentes *a* e *b*. Considere o coeficiente $\lambda = 5{,}5$. A Tabela 4.2 discrimina o valor do peso para cada interação.

Interação	Tempo(Δt)	Valor(v_i)	Peso($\omega t(r_i) = e^{-\frac{\Delta t(r_i)}{\lambda}}$)
1	0	0,90	$e^{-\frac{0}{5,5}} = 1{,}00$
2	3	0,70	$e^{-\frac{3}{5,5}} = 0{,}69$
3	7	0,80	$e^{-\frac{7}{5,5}} = 0{,}41$
4	*9*	0,50	$e^{-\frac{9}{5,5}} = 0{,}32$
5	*15*	-0,50	$e^{-\frac{15}{5,5}} = 0{,}15$
6	*17*	0,94	$e^{-\frac{17}{5,5}} = 0{,}12$
7	*18*	-0,30	$e^{-\frac{18}{5,5}} = 0{,}10$
8	*20*	0,00	$e^{-\frac{20}{5,5}} = 0{,}08$
9	*25*	0,40	$e^{-\frac{25}{5,5}} = 0{,}04$
10	*30*	0,00	$e^{-\frac{30}{5,5}} = 0{,}02$

Tabela 4.2 Cálculo do peso.

Numericamente é possível verificar na Tabela 4.2 que quanto maior Δt, menor o valor do peso. A escolha da constante de decrescimento (λ=5,5) garante que ao final de 30 dias, contando a partir do momento da criação, qualquer *rating* possuirá valor desprezível e poderá ser desconsiderado no cálculo de *T(a,b,c)*. A Figura 4.2 mostra este decrescimento.

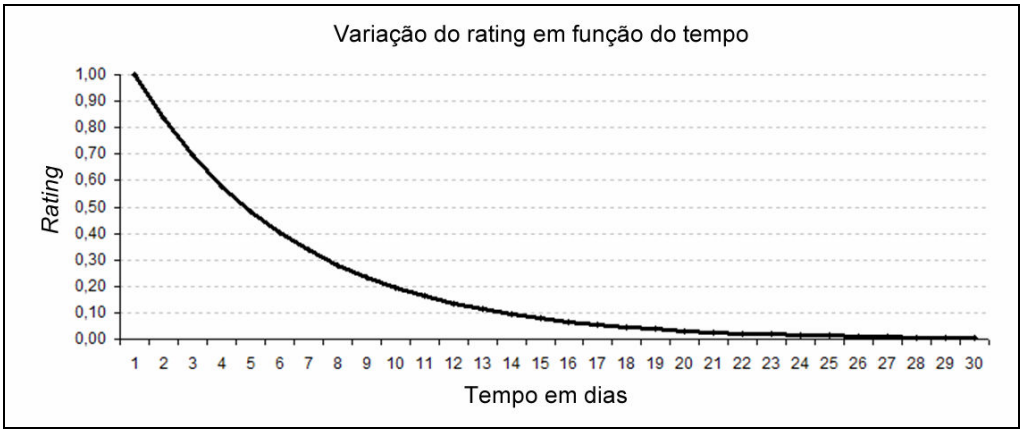

Figura 4.2 Decrescimento do peso durante 30 dias.

O exemplo utilizado neste gráfico apresenta um *rating* com o valor inicial máximo de 1. Ao decorrer 30 dias, o *rating* diminui seu valor rapidamente, chegando próximo a zero. Este comportamento permite aos agentes construir uma janela de tempo para descarte de *ratings* irrelevantes e, conseqüentemente, promovem maior otimização dos recursos de armazenamento localmente.

Deve-se salientar que não há garantia de que os agentes avaliadores sejam honestos em suas avaliações ou que sejam capazes de avaliar corretamente os agentes provedores. O modelo de confiança certificado trata o problema incluindo em seu processo de cálculo um peso que representa a credibilidade do avaliador. A *credibilidade* de um agente avaliador *w* calculada por outro avaliador *a* é denotada por T*Cr(a,w)*∈[-1,+1], onde *Cr* representa a *credibilidade* do avaliador. Desta forma, o peso de uma avaliação envolve a junção de dois pesos: tempo de criação do *rating* $\omega t(r_i)$ e credibilidade do avaliador $\omega c(r_i)$, expresso pela formula abaixo.

$$\omega(r_i) = \omega t(r_i) \cdot \omega c(r_i) \qquad \textbf{(4)}$$

Quando $\omega c(r_i)$ tem valor negativo, assume-se que o avaliador não possui credibilidade alguma, sendo assim, seus *ratings* serão desconsiderados para o cálculo da confiança. A Equação 5 formaliza a situação mencionada, ajustando o valor de $\omega c(r_i)$ para zero nas seguintes condições.

$$\omega c(r_i) = \begin{cases} 0 & se \quad TCr(a,w) \le 0 \\ TCr(a,w) & se \quad TCr(a,w) > 0 \end{cases} \qquad \textbf{(5)}$$

O cenário para a *credibilidade* envolve no mínimo três agentes: *a, b, w*. Considere que *a* avaliou *b e a* registrou localmente seu *rating*, dado por $r_a=(a,b,i_a,c,v_a)$. Quando o agente *a* recebe um *rating* de outro agente avaliador *w*, ele calcula a credibilidade de *w* pela comparação do seu rating r_a contra a avaliação de *w* sobre *b*. O *rating* de *w* em relação *b* é dado por $r_w=(a,b,i_w,c,v_w)$. O modelo de confiança certificado assume que a credibilidade de *w* para *a* é calculada pela diferença entre os dois valores (v_a, v_w), conforme a Equação 6.

$$TCr(a,w) = \begin{cases} 1 - |v_w - v_a| & se \quad |v_w - v_a| < \iota \\ -1 & se \quad |v_w - v_a| > \iota \end{cases} \quad \text{Onde: } (0 \le \iota \le 2) \qquad \textbf{(6)}$$

TCr(a,w) recebe um valor positivo caso a diferença entre os valores de v_w e v_a mantenha-se abaixo de um limite ι, caso contrário a credibilidade é negativa e pela Equação 6 o avaliador é considerado não confiável. Para exemplificar o cálculo da credibilidade considere as seguintes variáveis.

$\iota = 0{,}3 \quad v_a = 0{,}45 \quad v_w = 0{,}40$

O cálculo segue abaixo:

$\omega c(r_i) = TCr(a, w) = 1 - | v_w - v_a | = 1 - 0{,}05 = 0{,}95$

Pode-se afirmar sucintamente que o cálculo do modelo de confiança certificado baseia-se na média ponderada de um conjunto de *ratings* significativos *R* e que para distinguir a relevância destes *ratings* é utilizado um peso que leva em função o tempo de criação do *ratings* e a credibilidade dos agentes que os criou.

4.2.2 Segurança Baseada em Assinatura Digital

No Modelo de Confiança Certificada, a veracidade das informações trocadas entre os agentes do sistema é garantida pelo mecanismo de assinaturas digitais, que utilizam algoritmos de criptografia baseados em chaves assimétricas. Nesta abordagem, cada agente é munido com um par de chaves: uma chave denominada privada *Kpri* e outra denominada pública *Kpub*. A chave privada de cada agente deve ser mantida em segurança pelo agente que a possui, ao contrário à chave pública que pode ser distribuída a qualquer agente que necessite ler suas mensagens assinadas. A autenticidade da mensagem *m*, i.e., a certeza que *m* foi emitida pelo agente *a* é garantida pela assinatura digital *Signature* de *m* por meio da chave privada *Kpri(a)* representada como segue.

$$Autenticidade(a, m) \Leftrightarrow Signature\ (Kpri(a), m)$$

No caso da confidencialidade das informações, i.e., quando o agente *a* deseja certificar-se que apenas o agente destinatário *b* poderá ler *m*, a criptografia de *m* é feita por meio da chave pública de *b*, à medida que apenas *b* poderá descriptografar *m* com sua chave privada, conforme segue.

$$Confidencialidade(b, m) \Leftrightarrow Crypt(Kpub(b), m)$$

Como o agente avaliado armazena todas as suas avaliações recebidas, há o risco dele modificar seu conteúdo ou compartilhar apenas um subconjunto de avaliações que melhor o favoreçam. Para evitar esta situação, a arquitetura considera a existência de agente *Certificador*, ilustrado na Figura 4.3. Todo certificador é agente intrínseco do modelo de confiança certificado, à medida que são considerados plenamente confiáveis. Seu objetivo é garantir que apenas as avaliações cifradas por eles sejam utilizadas para o

cálculo da confiança evitando a seleção arbitrária de avaliações pelos agentes avaliadores.

Antes de entender como um agente certificador pode evitar esta seleção indevida, devemos conhecer o processo de cifragem e decifragem das avaliações feitas entre os agentes avaliadores, consumidores e certificadores. A Figura 4.3 descreve o processo de cifragem para um cenário, onde os três agentes estão representados.

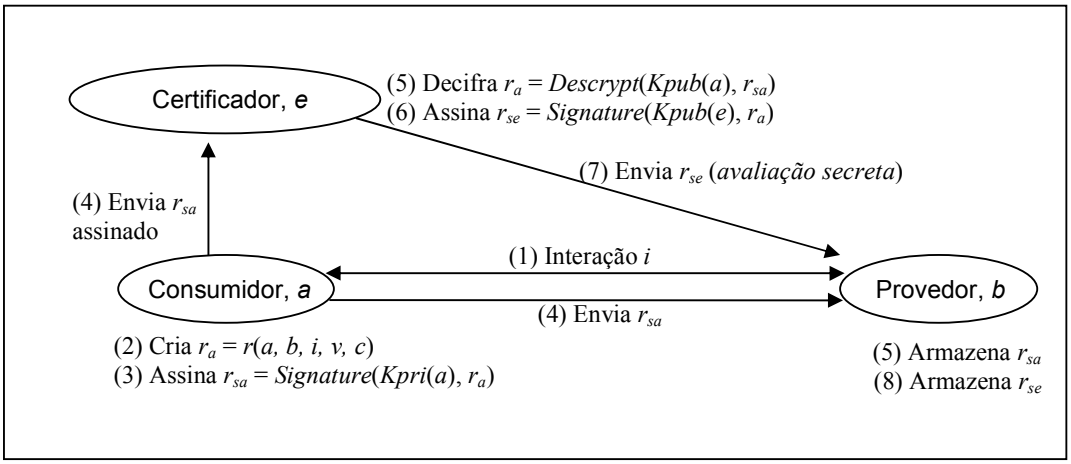

Figura 4.3 Criação e envio de avaliações cifradas.

O processo de criação, assinatura e criptografia das avaliações é composto pelas seguintes etapas: (1) Os agentes *a* e *b* realizam uma interação *i*. (2) A partir de *i,* o agente *a* avalia *b* criando um *rating* denominado r_a. (3) Antes de enviar a avaliação para *b*, *a* assina r_a com sua chave privada, *Kpri(a),* gerando r_{sa}. (4) Com a avaliação assinada, *a* envia simultaneamente r_{sa} aos agentes certificador *e* e ao provedor *b* (5) que armazena localmente r_{sa}. Portanto, *b* pode utilizar esta avaliação para analisar seu desempenho e detectar oportunidades de melhoria. O agente certificador *e* decifra o conteúdo de r_{sa} através da chave pública de *a* e (6) criptografa o *rating* com sua própria chave pública, gerando uma *avaliação secreta* chamada r_{se}. Como este r_e foi cifrado com a chave pública *Kpub*(*e*) apenas *e* poderá decifrá-lo, à medida que é o único agente detentor da chave privada *Kpri(e)*. (7) *e* envia r_{se} ao agente *b* que (8) o armazena localmente. O conjunto de avaliações do tipo r_{se} armazenada por *b* são as únicas consideradas autenticas para servirem de testemunha de *b*, visto que elas foram assinadas por um agente certificador e não podem ser lidas, conseqüentemente, não podem se modificadas ou selecionadas arbitrariamente pelo agente avaliado.

Estabelecido o processo de criação, assinatura e armazenamento das avaliações, o próximo passo é compreender como outros agentes podem obter o conjunto de

avaliações de *b* para calcular sua confiança. Como o agente certificador *e* é a única entidade do sistema capaz de decifrar as avaliações r_e, ele participa ativamente do processo de obtenção destas avaliações. A Figura 4.4 apresenta o fluxo de recuperação das avaliações de um agente provedor no momento que um agente *d* deseja calcular a sua confiança.

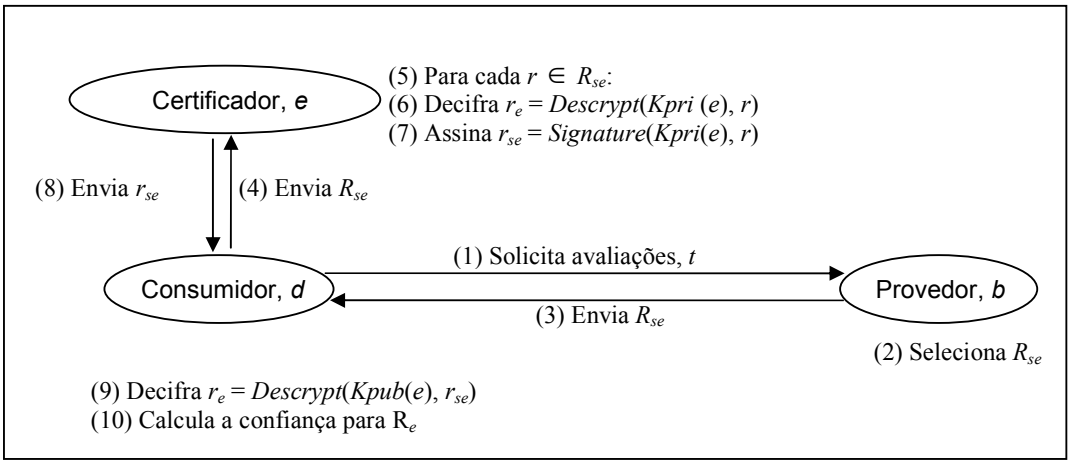

Figura 4.4 Recuperação e decifragem de avaliações.

O processo de recuperação das avaliações de um agente provedor é descrito nas seguintes etapas: (1) O agente *d* candidato a uma interação, requisita ao agente provedor *b* suas avaliações em relação a um determinado termo *t*. (2) *b* deve selecionar um grupo de *ratings* relevantes denominado de *R*, além da relevância, as avaliações devem ter sido geradas pelo agente certificador *e*, portanto esse conjunto é identificado por R_{se}. (3) Feita a seleção, *b* envia o conjunto R_{se} para *d*. (4) Como o conjunto encontra-se confidencialmente cifrado com chave pública do certificador, *d* envia as avaliações para serem decifradas por *e*. (5) Para cada avaliação r_{se} pertencente ao grupo das avaliações R_{se}. (6) *e* decifra r_{se} com sua chave privada. (7) Decifrada à avaliação, *e* assina r_{se} e a envia à *d* (8) que é (9) decifrada com a chave publica de *e*. (10) Com as avaliações decifradas, *d* as pode armazenar localmente ou realizar diretamente o cálculo da confiança. Ao final de tal processo, o agente *d* possui um conjunto de avaliações relevantes para o cálculo da confiança de *b*. Nota-se que, apesar do processo utilizar uma abordagem de confiança indireta baseada em testemunhos não há mais de três agentes envolvidos.

Diferente da Reputação Certificada às avaliações r_e não podem ser selecionadas de maneira a beneficiar indevidamente um provedor, à medida que não há acesso direto a suas avaliações. Os agentes certificadores, responsáveis por esta segurança têm a

política de não decifrar uma avaliação cujo solicitante seja o próprio agente avaliado, conforme ilustrado na Figura 4.5.

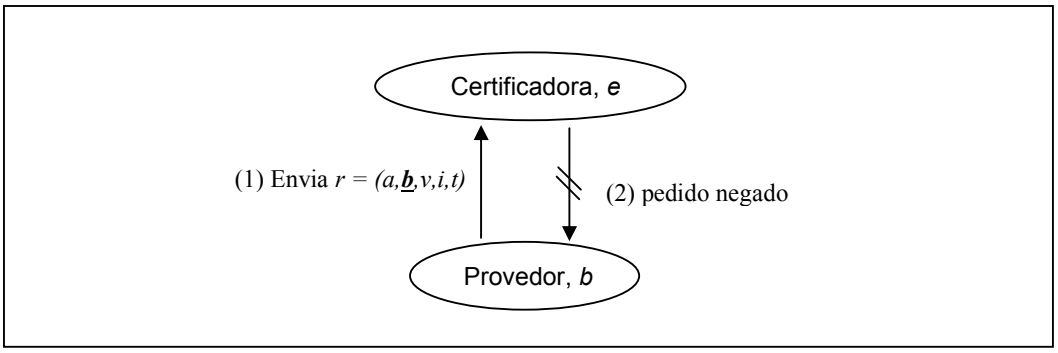

Figura 4.5 Política de acesso as avaliações.

O mecanismo de assinatura do Modelo de Confiança Certificado garante a autenticidade e confidencialidade das avaliações trocadas entre os agentes. Desta forma, é possível realizar um cálculo preciso e com baixo custo computacional para a confiança, à medida que os testemunhos são obtidos diretamente do agente avaliado.

Definidos os conceitos fundamentais do Modelo de Confiança Certificado a próxima seção descreve como o sistema CRONOS aplicando estes conceitos em sua arquitetura.

4.3 Arquitetura Multiagente

A seção anterior definiu conceitualmente o modelo de confiança *certificado*, entretanto as hipóteses levantadas sob ele, como o controle distribuído da confiança e o mecanismo simplificado para obtenção de testemunhos, dependem fortemente de um ambiente empírico que permita comprová-las. O uso de um ambiente conhecido como o *ART Testbed* (Fullan, 2005) poderia simplificar as etapas de execução e análise dos experimentos. Apesar de alguns fatores positivos, o *ART* não foi escolhido para este trabalho pelo fato de não permitir que os modelos de confiança sejam avaliados para diferentes tipos de problemas. Por exemplo, um modelo de confiança *a* pode ter boa eficiência para avaliação de pinturas, problema proposto pelo *ART*, entretanto quando aplicado a um sistema de e-commerce seu desempenho pode não ser o mesmo. A possibilidade de avaliar um modelo de confiança em diferentes óticas torna o CRONOS uma plataforma mais extensível.

A necessidade de construir uma estrutura computacional que automatize um procedimento de avaliação de modelos de confiança e que ainda adapte-se a diferentes

domínios de problema deu origem ao projeto CRONOS. Por conta do tratamento da confiança ser considerado um processo distribuído, sua estrutura é representada por uma arquitetura multiagente. A definição desta arquitetura visa antes de tudo avaliar genericamente diferentes modelos de confiança. Desta forma, espera-se também que esta arquitetura defina um conjunto mínimo de especificações para a construção de novos sistemas multiagente capazes de facilmente acoplarem-se ao modelo de confiança certificado. A descrição desta arquitetura e suas demais especificações são descritas nas próximas seções.

4.3.1 Estudo de Caso

Apesar da arquitetura CRONOS não depender do problema a ser tratado pelo modelo de confiança, neste estudo, foi necessário definir um domínio de problema que pudesse ser utilizado como prova de conceito. A escolha do problema envolveu duas características: o uso de um ambiente aberto e a influência do modelo de confiança no desempenho global do sistema. A primeira característica permite validar o modelo de confiança certificado sobre uma plataforma composta por agentes heterogêneos. A segunda característica facilita calcular o desempenho do modelo de confiança, na medida em que seu valor pode ser baseado no desempenho global do sistema. Outro importante aspecto considerado para a escolha do problema é que seus agentes fossem modelados como provedores ou consumidores de serviços, mantendo assim a compatibilidade de conceitos do modelo de confiança certificado. Apesar destas restrições, diversos sistemas como GRID, comércio eletrônico, computação peer-to-peer entre outros poderiam ser candidatos ao uso desta arquitetura.

Diante do número de possibilidades à escolha do problema, foram analisados os estudos a cerca da eficácia dos sistemas multiagente na resolução de problemas do mercado financeiro (Lee, 2002; Wang, 2002) especificamente ao mercado de ações Davis, (2000a, 2000b, 2002). No geral, os estudos apontam para uma abordagem multiagente, onde as interações entre os agentes visam buscar parceiros confiáveis para obtenção de recomendações de compra e venda de ativos. Considerado como um clube de investimento, estes agentes podem construir parcerias trocando informações e até mesmo vendendo serviços como consultores de mercado.

Além do estudo de caso prover uma abordagem distribuída, por meio de uma arquitetura multiagente, ele atende as características básicas ao uso do ambiente CRONOS: primeiro porque um sistema de mercado de ações pode ser projetado sob

uma plataforma aberta, na qual permite o livre acesso para entrada e saída de agentes que desejam obter recomendações ou prover serviços. Por ser um ambiente aberto, a criação de parcerias demanda confiança entre os agentes que pode ser controlada sob um modelo de confiança robusto; segundo que os agentes podem ser modelados sob a ótica de provedores e consumidores de serviços, na qual os serviços são representados pela venda de recomendações; finalmente, como o desempenho dos agentes é fortemente influenciado pela qualidade das suas parcerias e como essas parcerias são controladas pelo modelo de confiança, podemos assumir que a eficiência do modelo de confiança pode ser medida em função da eficiência local de cada agente.

A bolsa de valores virtual, na qual os agentes interagem, é representada no CRONOS pelo módulo *Simulador*, exibido na Figura 4.6. Suas operações garantem a abertura e fechamento de pregões, atualização das cotações, execução de ordens de compra e venda entre outras. A Figura 4.6 retrata o simulador e seu relacionamento com os demais elementos do sistema.

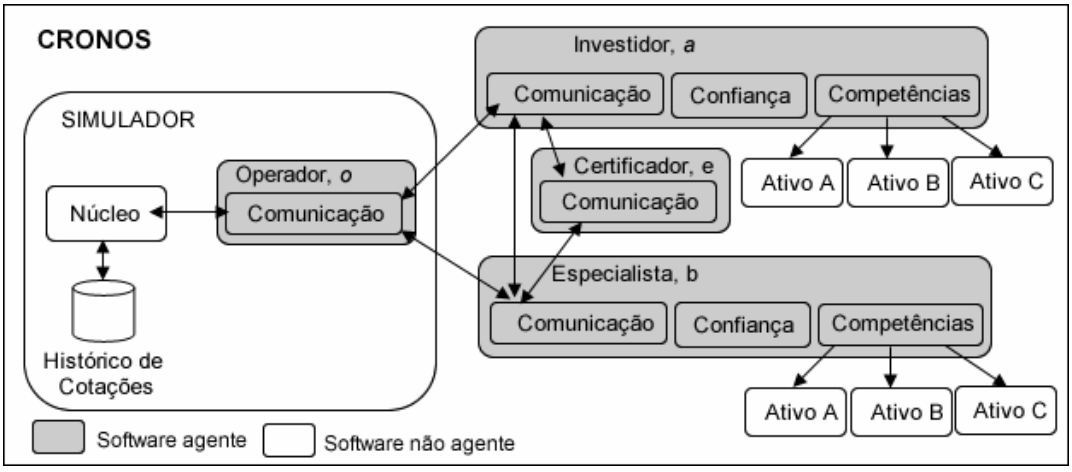

Figura 4.6 Estudo de caso sobre o mercado de ações.

O Simulador é composto por um conjunto de agentes denominados *Operadores*, que provem uma interface de comunicação entre os demais agentes do CRONOS e o *Núcleo* do simulador. O Núcleo controla as operações básicas de uma bolsa de valores: abertura e fechamento de pregões, divulgação de cotações, execução de ordens de compra e venda de papéis, entre outras. Seu principal insumo é a base de cotações históricas dos principais ativos comercializados sob a Bolsa de Valores de São Paulo – Bovespa – (Bovespa, 2008). Além do módulo Simulador, o sistema é composto por um grupo de agentes denominados *investidores* com autonomia para negociar ativos no

simulador. Outro grupo denominado de *especialistas* que fornecem recomendações aos agentes investidores.

A estrutura destes agentes é organizada em três camadas, nos quais temos:

- *Comunicação:* define os protocolos de comunicação entre os agentes e o Simulador. Para o modelo de confiança certificado, esta camada também prover os mecanismos de segurança baseados em assinaturas digitais.
- *Competência*: prover habilidades para analisar o mercado de ações. Utiliza estratégias da análise financeira como a *Análise Técnica*[16] para reconhecer os momentos de compra e venda dos ativos.
- *Confiança*: utilizado para determinar o grau de confiança dos agentes e por conseqüência selecionar bons parceiros para interação, i.e, aqueles que possuem alta probabilidade de acerto em suas recomendações.

4.3.2 Plataforma de Agentes

Uma plataforma padrão de agentes fornece uma infra-estrutura computacional na qual os agentes são inseridos. Esta infra-estrutura é composta por máquinas, sistema operacional, componentes de software, estrutura de rede, entre outros. A plataforma de agentes do CRONOS foi projetada de acordo com a especificação *FIPA* [FIP02a], que trata especificamente da definição de arquitetura para sistemas multiagente. Esta seção apresenta como a arquitetura FIPA foi aplicada à plataforma de agentes do CRONOS. A Figura 4.7 exibe os principais elementos desta plataforma.

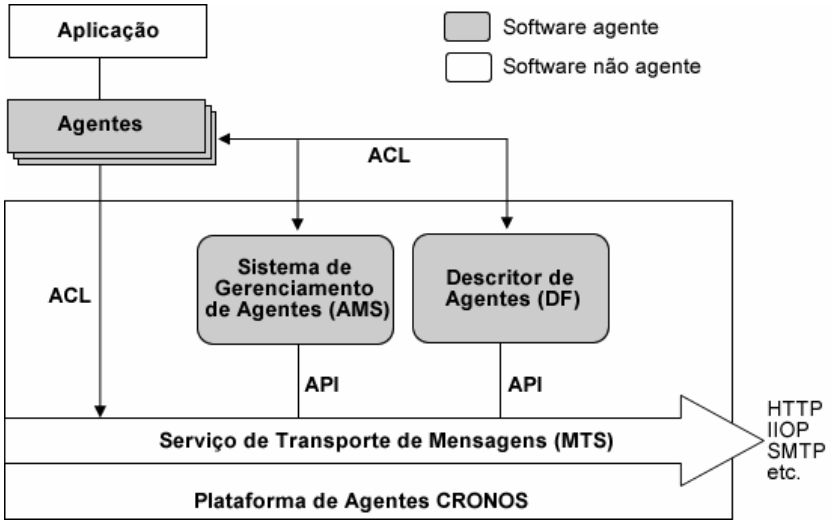

Figura 4.7 Representação da plataforma de agente em múltiplos containeres.

[16] A análise técnica ou análise gráfica é a abordagem que faz uso de gráficos para definir os melhores momentos de compra e venda de um ativo. Vide Apêndice A.

A *Aplicação,* que nesta dissertação, representa o simulador da bolsa de valores utiliza a plataforma para interagir com os dois grupos de agentes pré-definidos: *investidores* e *especialistas*. A camada de comunicação dos agentes é suportada pelo serviço de transporte de mensagens MTS (*Message Transport Service*). Este serviço prover diversos protocolos de comunicação, tais como: HTTP, SMTP, IIOP. Para o CRONOS o protocolo HTTP é considerado o protocolo padrão de comunicação do MTS. Quanto a linguagem para a troca de informações, foi utilizado o padrão ACL (*Agent Communication Language*) [FIP02c]. Da lista de parâmetros disponíveis na especificação ACL para envelopamento de mensagens, foi selecionado um subconjunto de parâmetros apresentado na Tabela 4.3.

Parâmetro	Descrição
To	Endereço do destinatário
From	Endereço do emissor
Acl-representation	Representação da mensagem, aqui definida como String
Security-object	Informações a cerca de criptografia e certificado digital, aqui utilizado para envia das assinaturas digitais
Date	Data de criação da mensagem

Tabela 4.3 Parâmetros de envelopamento de mensagens do CRONOS.

O conteúdo das mensagens também utiliza um subconjunto de parâmetros da especificação ACL. Este subconjunto é apresentado na Tabela 4.4.

Parâmetro	Descrição
Performative	Ato de fala da mensagem.
Sender	Agente emissor da mensagem
Receiver	Agente receptor da mensagem
Reply-to	Endereço para resposta de mensagens
Content	Conteúdo da mensagem
Language	Linguagem utilizada. Aqui definida como Lisp
Protocol	Protocolo de comunicação. Aqui definido como HTTP
Conversation-id	Identificador do tipo de conversa.

Tabela 4.4 Parâmetros de conteúdo de mensagem.

O trecho de código a seguir, ilustra a estrutura e conteúdo de uma mensagem ACL trocada entre dois agentes do CRONOS. Neste exemplo, a mensagem é emitida por um agente *investidor* para um agente *especialista*. A mensagem visa informar ao agente *especialista* sobre o valor de utilidade de uma recomendação criada por ele.

```
(INFORM
:receiver    (set  (  agent-identifier  :name  especialista@NOTEBOOK:1099/JADE
:addresses (sequence http://189.34.103.139:7778/acc )) )
:content     "(setq rating-agent-from         'cliente)
             (setq rating-agent-to            'servidor)
             (setq rating-iteration-id        078)
             (setq rating-iteration-value     0.95)
             (setq rating-iteration-term      'VALE5)"
:language  LISP
:conversation-id  INFORM-RATING )
```

Neste exemplo, a avaliação do *investidor* para o serviço prestado pelo *especialista* obteve valor igual a 0,95, em um intervalo de [-1,+1], para o termo *VALE5*.

Outro elemento importante desta arquitetura é o *Sistema de Gerenciamento de Agentes* AMS (*Agent Management System*), apresentado na Figura 4.7. Ele responde pelo gerenciamento dos agentes inseridos na plataforma. Esta gestão abrange a criação, identificação, destruição, migração e monitoramento dos agentes. Além disso, o AMS do CRONOS realiza o controle das interações entre agentes, por meio do serviço de protocolo de interações. Desta forma, antes de iniciar uma interação, um dos agentes envolvidos solicita ao AMS um número de protocolo que identificará unicamente a interação em questão.

O *Descritor de Agentes* DF (*Directory Facilitator*) é um componente da plataforma que prove o cadastro de serviços dos agentes, semelhante ao conceito de *páginas amarelas*[17]. Neste descritor são cadastrados os serviços que cada agente *especialista* dispõe. Cada ativo é considerado pelo descritor como um tipo de serviços, i.e., se um *especialista* fornece recomendações para três ativos serão criados três registros no DF, um para cada ativo. Desta maneira, os agentes *investidores* podem consultar por ativo quais seus respectivos especialistas. Vale destacar que a busca pelos bons *especialistas* é de responsabilidade do modelo de confiança, à medida que não cabe ao DF este tipo de consulta.

4.3.3 Contêiner

Uma das premissas do sistema CRONOS é seu ambiente aberto e de larga escala. O requisito escalabilidade é suportado pela arquitetura multiplataforma desenvolvida para garantir que agentes construídos por diferentes desenvolvedores e diferentes tecnologias consigam utilizar o sistema e interagir entre si. A

[17] Do inglês *Yellow Pages*, criada em 1883 pelo norte-americano Reuben H. Donnelley é uma lista telefônica para divulgação de empresas e serviços. Em sistemas multiagente o termo *Directory Facilitator* é analogamente comparado ao *Yellow Pages* pela similaridade na divulgação dos serviços fornecidos pelos agentes.

interoperabilidade do CRONOS entre plataformas de agentes que utilizam hardwares, sistemas operacionais e linguagens de programação dos mais variados possíveis só será possível se as plataformas atenderem as especificações de arquitetura FIPA [FIP02a]. A Figura 4.8 apresenta os principais elementos que compõem a arquitetura multiplataforma do sistema CRONOS:

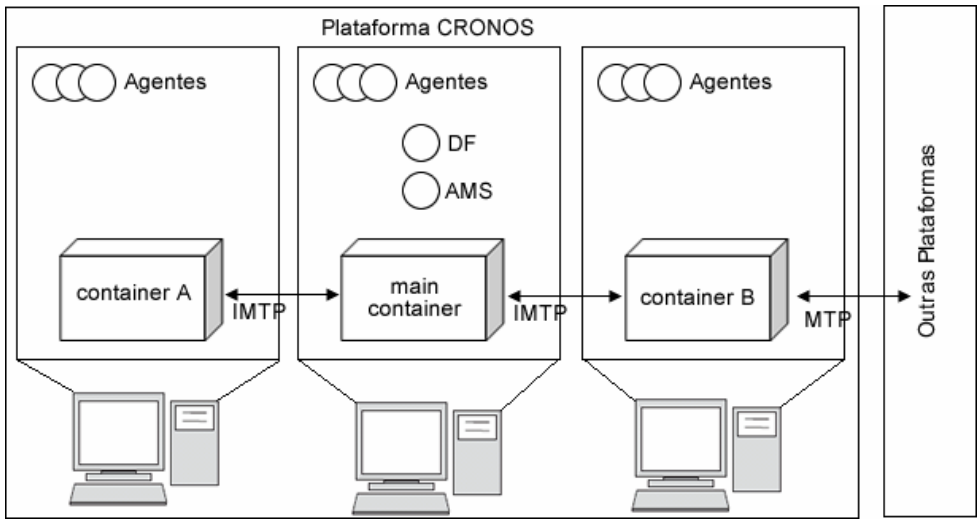

Figura 4.8 Arquitetura multiplataforma.

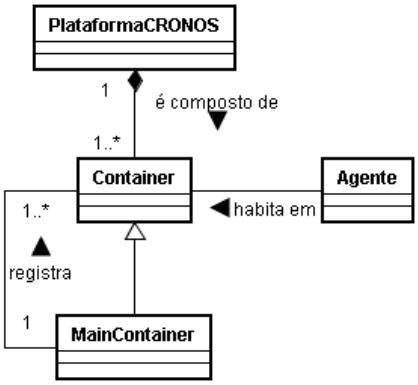

Figura 4.9 Relacionamento entre os principais elemento da arquitetura.

Conforme observado na Figura 4.9, a plataforma CRONOS é composta por um conjunto de contêineres que provem recursos necessários à hospedagem dos agentes. Apesar dos contêineres pertencerem a uma mesma plataforma, estes podem ser distribuídos sob diversos servidores em uma rede TCP-IP. Entre tais contêineres, há um contêiner especial denominado contêiner principal , que interliga os demais. Ele é o primeiro a ser criado e os demais contêineres devem ser registrados a ele como contêineres secundários.

Durante a experimentação, deste trabalho, foi necessário o mecanismo de múltiplos contêineres para balancear o processamento dos experimentos. Os agentes que pertencem ao ambiente aberto como os *especialistas* e os *investidores* foram distribuídos em contêineres secundários numa média de cem agentes por contêiner. Os agentes intrínsecos do CRONOS com os *operadores* e *certificadores* foram hospedados no contêiner principal. Desta maneira, foi possível simular um ambiente multiagente de larga escala sem a necessidade de servidores com alta capacidade de processamento.

4.4 Projeto

A construção do projeto CRONOS foi desenhada com base no Framework JADE (Jade, 2009). A escolha deste framework foi motivada pela popularidade junto a comunidade acadêmica para o desenvolvimento de sistemas multiagente (Bellifemine, 2007 & Jade, 2009). Ademais, o JADE atende todas as especificações FIPA, provendo simples integração com outros sistemas. Ele também facilita a construção de módulos essenciais à plataforma CRONOS tais como: gerenciamento de agentes, descritores de agentes, camada de transporte de mensagens, entre outros. Esta seção descreve os principais pontos de extensão do framework JADE para a elaboração do projeto CRONOS.

4.4.1 Agentes

A principal característica em comum dos agentes CRONOS é sua especialização a partir dos agentes JADE. Por meio da herança foi possível reutilizar diversos mecanismos da plataforma como, a comunicação entre agentes pelo envio de mensagens e a capacidade dinâmica de incorporar novos comportamentos. O refinamento de cada tipo de agente foi projetado em função de suas específicas competências, tais como: a integração com os modelos de confiança, cifragem de mensagens, análise do mercado de ações entre outras. A distinção dos agentes, por meio de suas competências, deu origem às especializações hierarquicamente organizadas conforme a Figura 4.10:

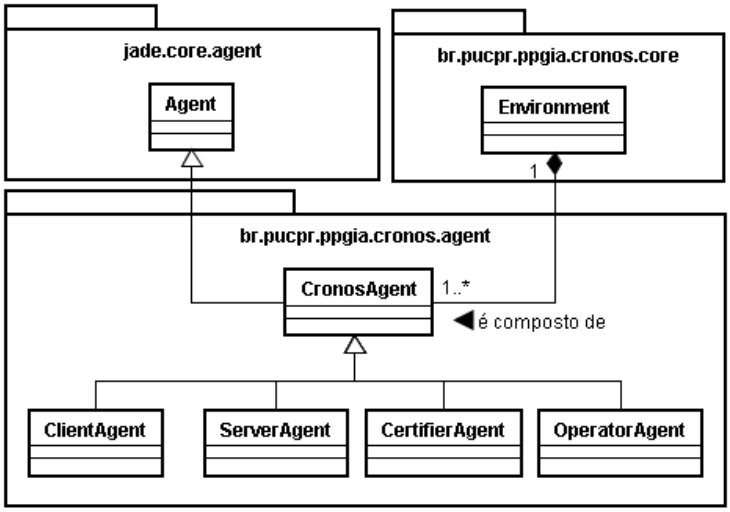

Figura 4.10 Especialização dos agentes CRONOS.

Os agentes do sistema são especializados a partir da classe *CronosAgent* que prover recursos comuns como registro no DF, envio de mensagens, interação com o ambiente (*Environment*), entre outras. Os agentes internos do CRONOS definidos na arquitetura como operadores e certificadores são representados neste projeto respectivamente pelas classes *OperatorAgent* e *CertifierAgent.* Ambos agentes são considerados parte da infra-estrutura do sistema. Os agentes especialistas e investidores, descritos pela arquitetura como agentes livres são projetados pelas classes: *ServerAgent* e *ClientAgent*. Os agentes servidores representam os especialistas do mercado financeiro. Estes agentes fornecem o serviço de recomendação de ativos. Os agentes clientes representam os investidores deste mercado, à medida que eles são consumidores de serviços.

A abstração cliente-servidor permite generalizar a construção de sistemas multiagente para diferentes domínios de problema. Por exemplo, agentes servidores em um sistema de comércio eletrônico podem representar vendedores e os agentes clientes representam os consumidores. Em Botelho, *et al.* (2009a), a hierarquia de agentes do CRONOS foi utilizada para avaliar a qualidade dos serviços prestados por Web Service.

4.4.2 Comportamentos

Os agentes possuem uma dinâmica baseada em comportamentos (*behaviour*). Neste sentido, o comportamento é considerado como um componente de software de baixo acoplamento, i.e., pode ser incluído, removido ou modificado dinamicamente

pelos agentes. O CRONOS utiliza a plataforma JADE para pré-definir um conjunto mínimo de comportamentos aos seus agentes. Tais comportamentos são acoplados aos agentes clientes e servidores conforme exibido na Figura 4.11.

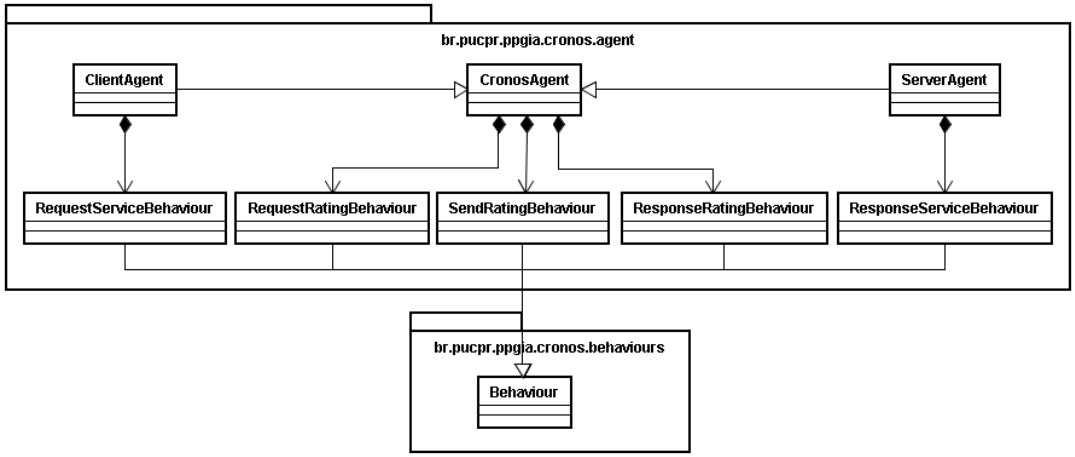

Figura 4.11 Comportamentos do CronosAgent.

O comportamento dos agentes *CronosAgent* especializa a classe *Behaviour,* que representa uma abstração do comportamento proposto pela plataforma JADE. O conjunto mínimo de comportamentos definidos pelo CRONOS foi modelado conforme a descrição do Modelo de Confiança Certificado. Sua especificação propõe um protocolo de comunicação entre agentes clientes e servidores. Este conjunto é composto pelos seguintes tipos de comportamento:

- *RequestRatingBehaviour*: acionado no momento em que um agente deseja interagir com outro agente desconhecido. O agente ativa este comportamento para solicitar a outro seu conjunto de avaliações. Normalmente ocorre do agente cliente para o agente servidor.
- *ResponseRatingBehaviour*: utilizado para responder a solicitação do comportamento anterior. Geralmente, realizado do agente servidor para o cliente. O servidor pode negar a solicitação ou atendê-lo enviando um conjunto de avaliações a seu respeito.
- *RequestServiceBehaviour*: utilizado para solicitar um serviço. Quando o agente cliente recebe as avaliações do servidor ele calcula a confiança e caso atenda as suas expectativas, esse comportamento é disparado com as informações do serviço requisitado.

- *ResponseServiceBehaviour*: utilizado para acionar o serviço solicitado. Geralmente ocorre no agente servidor. O resultado do serviço prestado é enviado ao agente cliente. No caso do mercado de ações, o resultado do serviço representa uma recomendação de compra ou vende de um ativo.
- *SendRatingBehaviour*: utilizado para informar uma avaliação. Normalmente ocorre do agente cliente para o servidor. O momento de envio da avaliação é decidido pelo agente cliente.

Apesar deste trabalho não utilizar outros comportamentos, além do conjunto mínimo apresentado, os sistemas que utilizam a arquitetura CRONOS podem, conforme a necessidade, incorporar comportamentos adicionais. A ordem com que os comportamentos são acionados descreve o fluxo de execução do Modelo de Confiança Certificado. Para mostrar a dinâmica destes comportamentos, a Figura 4.12 apresenta um diagrama de seqüência, no qual simula um ciclo completo de interação entre dois agentes.

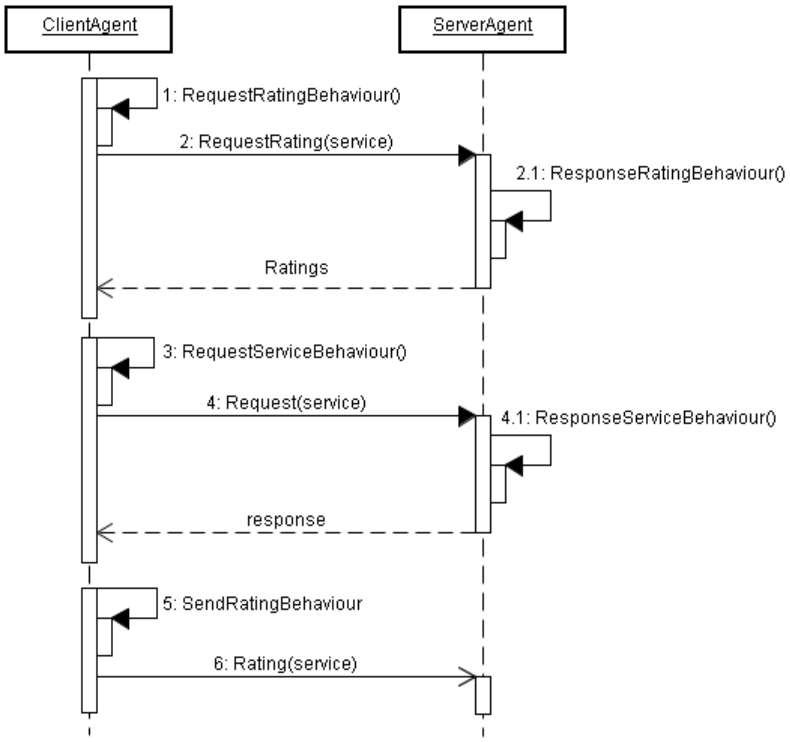

Figura 4.12 Interação de agentes baseada em comportamentos.

O diagrama descreve um cenário específico. No momento que um agente cliente apresenta interesse de interagir com um agente servidor desconhecido, o comportamento 1 : *RequestRatingBehaviour* é acionado requisitando do servidor suas

avaliações, i.e., avaliações que ele recebeu de outros clientes em interações passadas. Em resposta ao pedido, o servidor aciona o comportamento 2.1 : *ResponseRatingBehaviour* que selecionará os *ratings* a serem enviados ao cliente. No momento que o cliente opta pela interação, i.e., deseja consumir o serviço, o comportamento 3 : *RequestServiceBehaviour* envia uma mensagem ao servidor com as informações necessários para a realização do serviço. Em resposta a requisição, o servidor aciona o comportamento 4.1 : *ResponseServiceBehaviour*, que executa o serviço requerido. Opcionalmente, ao final da interação, o cliente aciona o comportamento 5 : *SendRatingBehaviour* para enviar ao servidor sua avaliação em relação ao serviço fornecido.

4.4.3 Painel de Controle

O CRONOS possui um painel de controle que permite interagir com o usuário final e monitorar em tempo real o comportamento do sistema. O foco do monitoramento está no desempenho dos modelos de confiança utilizados em cada experimento. O valor deste desempenho é calculado conforme as fórmulas descritas pela seção 5.2.1. Como pode ser observado na Figura 4.13, a representação dos resultados é feita em função dos modelos de confiança por meio de gráficos de linha.

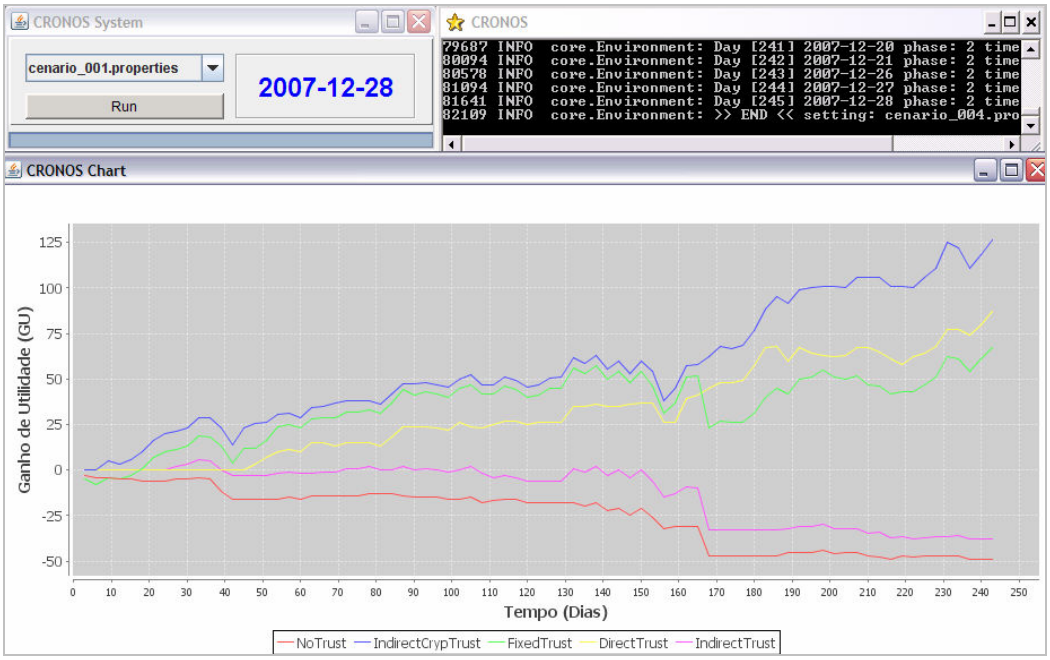

Figura 4.13 Painel de controle do CRONOS.

O painel de controle do CRONOS permite ao usuário selecionar um determinado cenário de execução. Cada cenário representa um conjunto de variáveis que são inicializadas a partir de um arquivo de configuração. Estas variáveis definem quais modelos de confiança serão utilizados, quantos agentes participaram da simulação, como os dados serão coletados entre outros parâmetros. O sistema exibe todos os arquivos disponíveis para a escolha do usuário.

A parte superior do painel controle do CRONOS é responsável por capturar do usuário o cenário a ser executado e exibir as principais informações do sistema. Ao iniciar uma execução, o CRONOS aciona o simulador da bolsa de valores que controla virtualmente a abertura e fechamento dos pregões. É exibido a data atual do pregão e os principais valores do ambiente para respectivo dia. A parte inferior é composta por um gráfico de linhas que apresenta o desempenho de cada modelo de confiança analisado. O eixo horizontal do gráfico representa o tempo em número de dias. Ao final de cada dia, o CRONOS calcula o valor de utilidade (UG) de cada modelo de confiança e os exibem no eixo vertical do gráfico. Além do gráfico de linhas, o sistema reproduz os resultados em arquivo texto para facilitar sua análise por outras ferramentas.

4.5 Experimento

Esta seção descreve o experimento realizado para avaliar o *modelo de confiança certificado* em relação aos atuais modelos existentes. Em suma, o experimento coleta o desempenho global dos modelos de confiança por meio do desempenho local dos seus respectivos agentes. Conforme o estudo de caso que trata o mercado de ações, apresentado na seção 4.3.1, o experimento utiliza o simulador da bolsa de valores, provido internamente pelo sistema CRONOS. Neste contexto, o desempenho é representado pela capacidade dos agentes em maximizar seus lucros por meio da compra e venda de ativos.

Como a finalidade do experimento é verificar se o modelo de confiança consegue distinguir os bons parceiros dos maus, foi necessário construir agentes provedores com níveis de qualidade pré-definidos. Obviamente, estes agentes não são monitorados pelo coletor de desempenho do sistema. Esses níveis de qualidade foram estabelecidos nas seguintes categorias: *good providers*, *bad providers*, *ordinary provider*s and *malicious providers*. Os agentes do primeiro grupo utilizam os melhores métodos de análise financeira e conseguem obter altas taxas de acerto em suas recomendações. O segundo grupo possui comportamento inverso ao primeiro, portanto

sua taxa de erro é semelhante à taxa de acerto do primeiro grupo. O terceiro grupo utiliza técnicas de análise financeira com desempenho mediano (ex. *média móvel*[18]). O quarto, considerado malicioso, utiliza as mesmas técnicas do terceiro grupo. Entretanto, os provedores deste grupo ordenam seus *ratings* arbitrariamente de forma a beneficiar-se. Esta estratégia visa confundir os investidores quando ao real desempenho dos agentes maliciosos. Os agentes consumidores foram organizados em quatro grupos, a saber: *No_Trust*, que não possuem nenhum modelo de confiança; *Direct_Trust*, que implementam um modelo de confiança direta; *Cr_Trust*, que implementam um modelo de confiança baseado em reputação certificada; e *Cryp_Trust*, que implementam o modelo de confiança certificado proposto neste trabalho. Os agentes consumidores interagem com os diferentes tipos de provedores e por meio de seus modelos de confiança selecionam os melhores parceiros de forma a maximizar os seus lucros.

Cada consumidor pode consultar qualquer provedor e realizar quantas ordens de compra ou venda forem necessárias. A avaliação do modelo de confiança é medida pelo desempenho da carteira[19] dos agentes. Cada carteira é iniciada com o mesmo montante de capital, ao final dos pregões verifica-se o percentual de crescimento da carteira. Este percentual pode ser positivo, quando o valor da carteira aumenta ou negativo quando diminui. É importante destacar que o percentual de crescimento da carteira do agente representa localmente o ganho de utilidade (UG) do modelo de confiança.

Os agentes atuam sob dados reais da bolsa de valores de São Paulo (Bovespa, 2008). Apesar da base de dados possuir o histórico de cotações de milhares de ativos dos últimos vintes anos, nosso simulador reproduziu uma bolsa mais simples contendo algumas dezenas de ativos e simulou os pregões referentes aos anos de 2007 e 2008 totalizando 473 dias úteis de pregão. Cada experimento é iniciado com a criação dos agentes provedores e consumidores. Foram dotados, respectivamente, cada provedor com uma única estratégia de análise financeira e cada consumidor com um único modelo de confiança. Ao final de cada dia de pregão, o *ganho de utilidade* (GU) local dos agentes consumidores é somado agrupando-se pelo modelo de confiança do agente. Este somatório é divido pelo número de agentes do modelo, o resultado representa o ganho de utilidade global do modelo de confiança para aquele respectivo dia de pregão. Tomemos como exemplo os agentes *a* e *b* que utilizam o modelo de confiança *W* e os

[18] A análise técnica realizada pela média móvel é descrita no Apêndice A, juntamente com mais informações sobre outros tipos de análise financeira.

[19] A carteira de investimentos representa um conjunto de ativos que pertencente a um investidor, seja este uma pessoa física ou jurídica. Neste trabalho cada agente cliente representa um investidor.

agentes *c* e *d* que utilizam o modelo de confiança *Y*. A Tabela 4.5 apresenta o ganho de utilidade local de cada agente para quatro dias de pregão e o ganho de utilidade global dos modelos.

Dia	Agente, *a*	Agente, *b*	Agente, *c*	Agente, *d*	Modelo, *W*	Modelo, *Y*
60	30%	35%	10%	15%	(30+35)/2 = 32,5%	(10+15)/2 = 12,5%
65	33%	34%	25%	25%	(33+34)/2 = 33,5%	(25+25)/2 = 25,0%
70	29%	31%	33%	22%	(29+31)/2 = 30,0%	(33+22)/2 = 27,5%
82	22%	28%	30%	34%	(22+28)/2 = 25,0%	(30+34)/2 = 32,0%

Tabela 4.5 Cálculo do ganho de utilidade do modelo de confiança.

A reprodução do ganho de utilidade sobre um gráfico de linhas facilita a comparação visual dos modelos de confiança ao longo de um período de tempo, conforme exibido na Figura 4.14.

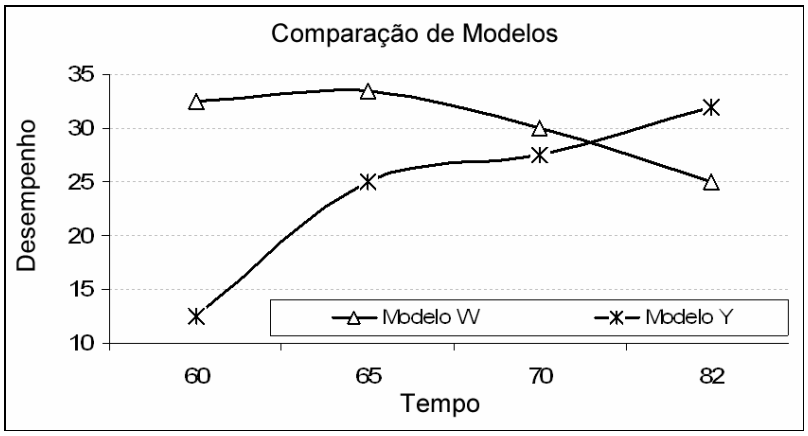

Figura 4.14 Comparação entre modelos por gráfico de linhas.

A comparação entre modelos de confiança nos parece uma abordagem essencial. Ela facilita a compreensão da sofisticação necessária dos mecanismos vis-à-vis a cenários cada vez mais desafiadores.

4.5.1 Ambiente

Para analisar a hipótese de que o *modelo de confiança certificado* pode tratar a situação de agentes maliciosos, na qual os agentes provedores selecionam os *ratings* para se beneficiar, foram elaborados dois ambientes distintos: *ambiente honesto* e *ambiente desonesto*. No ambiente considerado honesto, os agentes servidores retornam informações verdadeiras a cerca das suas avaliações e no ambiente denominado *desonesto* os servidores ocultam suas avaliações negativas e repassam aos agentes clientes apenas suas melhores avaliações. Espera-se que com a introdução dos agentes

maliciosos, pelo ambiente desonesto, haja redução no ganho de utilidade do *modelo direto* e *reputação certificada* e haja manutenção do desempenho do *modelo de confiança certificado*.

4.5.2 Ambiente Honesto

O ambiente é dito honesto, por conta da honestidade dos agentes provedores no envio dos seus *ratings*, neste ambiente não há omissões de *ratings* negativos. Isso garante as condições necessárias aos agentes clientes para calcular o valor de confiança do provedor com alta precisão. Os experimentos neste ambiente foram configurados para dois tipos de cenários. O primeiro cenário compreende agentes provedores honestos que utilizam uma única técnica de investimento durante todos os pregões, portanto seu desempenho não varia com grande intensidade. A Tabela 4.6 apresenta as variáveis utilizadas para este cenário.

Variáveis do Experimento	Símbolo	Valor
Número de pregões	N	473
Número de agentes provedores:	N_P	500
Número de *Good providers*	N_{PG}	166
Número de *Ordinary providers*	N_{OP}	168
Número de *Bad providers*	N_{PB}	166
Número de *Malicious providers*	N_{MI}	0
Número de agentes clientes	N_C	500
Variação de desempenho	V_P	não

Tabela 4.6 Cenário 1, ambiente honesto sem mudanças.

O objetivo deste cenário é avaliar os modelos de confiança sob um ambiente simples, no qual não há agentes maliciosos e nem há necessidade de detectar variações de desempenho dos provedores. A execução do primeiro cenário composto pelos 473 pregões é apresentada pela Figura 4.15.

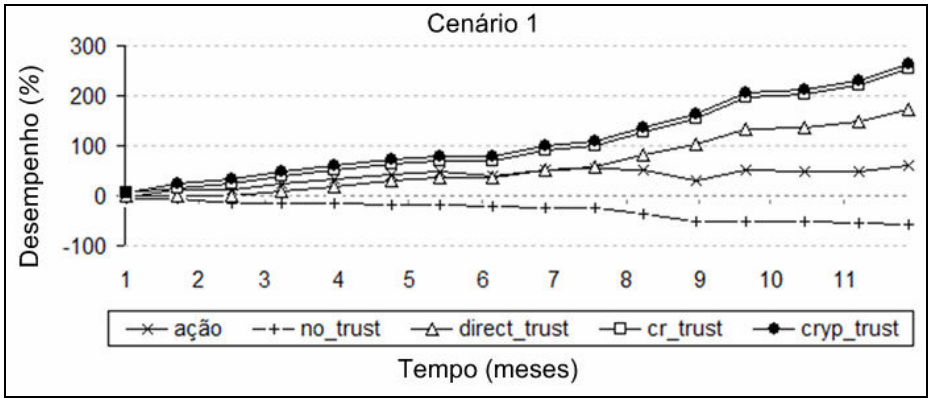

Figura 4.15 Cenário 1, ambiente honesto sem mudanças.

	ação	no_trust	direct_trust	cr_trust	cryp_trust
GU Final	61%	-57%	174%	260%	263%

Tabela 4.7 Cenário 1, ganho de utilidade final.

O resultado desse experimento mostra que para ambientes muito simples, como o apresentado no Cenário 1, não há grande diferença no desempenho dos modelos de confiança, à medida que os provedores possuem comportamento bastante previsível. É possível notar uma leve superioridade dos modelos indiretos *cryp_trust* e *cr_trust* em relação ao modelo direto *direct_trust*. Esta pequena diferença ocorre, por que os modelos diretos precisam de mais tempo para detectar, pela primeira vez os bons provedores. Entretanto, após descobri-los percebemos que seu gráfico de linha possui o mesmo ângulo de crescimento em relação aos modelos indiretos. Pode-se então concluir que o desempenho dos três modelos é equivalente.

O segundo cenário do ambiente honesto é composto por agentes provedores com alta variação no seu desempenho. Este comportamento é controlado periodicamente pelo CRONOS. A cada trinta pregões, o sistema troca o nível de qualidade dos agentes provedores da seguinte maneira: os agentes *good providers*, se tornam *bad providers*, e vice-versa. Desta forma, os agentes clientes precisam detectar rapidamente a mudança de comportamento destes provedores. A Tabela 4.8 apresenta a configuração do Cenário 2 utilizado neste segundo experimento.

Variáveis do Experimento	Símbolo	Valor
Número de pregões	N	473
Número de agentes provedores:	N_P	500
Número de *Good providers*	N_{PG}	166
Número de *Ordinary providers*	N_{OP}	168
Número de *Bad providers*	N_{PB}	166
Número de *Malicious providers*	N_{MI}	0
Número de agentes clientes	N_C	500
Variação de desempenho	V_P	sim

Tabela 4.8 Cenário 2, ambiente honesto com mudanças.

A diferença entre os cenários 1 e 2 é o valor do variável V_P que indica a variação de desempenho dos provedores. O resultado deste segundo experimento é observado gráfica da Figura 4.16.

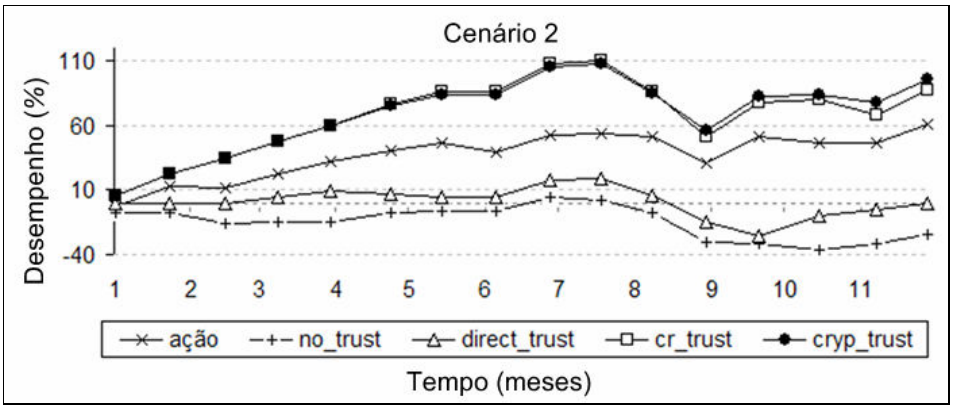

Figura 4.16 Cenário 2, ambiente honesto com mudanças.

	ação	no_trust	Direct_trust	cr_trust	cryp_trust
GU Final	61	-24%	0%	87%	96%

Tabela 4.9 Cenário 1, ganho de utilidade final.

O resultado deste experimento indica uma grande queda no desempenho do modelo direto, explicada por sua lentidão em ambientes de rápida mudança. O tempo utilizado para o modelo direto calcular com precisão o valor da confiança dos seus agentes ultrapassa o tempo normal destes agentes mudarem de desempenho. Os modelos indiretos, apesar da queda de desempenho em relação ao primeiro cenário, continuam superiores, à medida que a detecção é mais ágil. Isto se deve ao fato de que os próprios provedores fornecem as informações necessárias para a detecção da sua mudança de desempenho. Os modelos *cryp_trust* e *cr_trus* continuam tecnicamente equivalentes, visto que até o momento ambos não precisaram tratar a presença de agentes maliciosos.

4.5.3 Ambiente Desonesto

O ambiente é dito desonesto, por conta da presença de agentes provedores maliciosos que omitem dos consumidores seus *ratings* negativos, com o intuito de obter vantagem competitiva. A omissão dos *ratings* diminui a precisão do cálculo da confiança. Semelhante aos experimentos realizados no ambiente honesto, os dois cenários propostos serão aplicados juntamente com agentes maliciosos. A configuração do Cenário 1 para este ambiente é apresentada na Tabela 4.10.

Variáveis do Experimento	Símbolo	Valor
Número de pregões	N	473
Número de agentes provedores:	N_P	500
Número de *Good providers*	N_{PG}	125
Número de *Ordinary providers*	N_{OP}	125
Número de *Bad providers*	N_{PB}	125
Número de *Malicious providers*	N_{MI}	125
Número de agentes clientes	N_C	500
Variação de desempenho	V_P	Não

Tabela 4.10 Cenário 1, ambiente desonesto sem mudanças.

Conforme observado na tabela anterior, 25% dos agentes provedores são maliciosos. Esta proporção torna o ambiente altamente ruidoso aos agentes consumidores, principalmente para aqueles que não possuem um modelo de confiança certificado. O resultado para o primeiro cenário do ambiente desonesto é apresentado pela Figura 4.15 e Tabela 4.11.

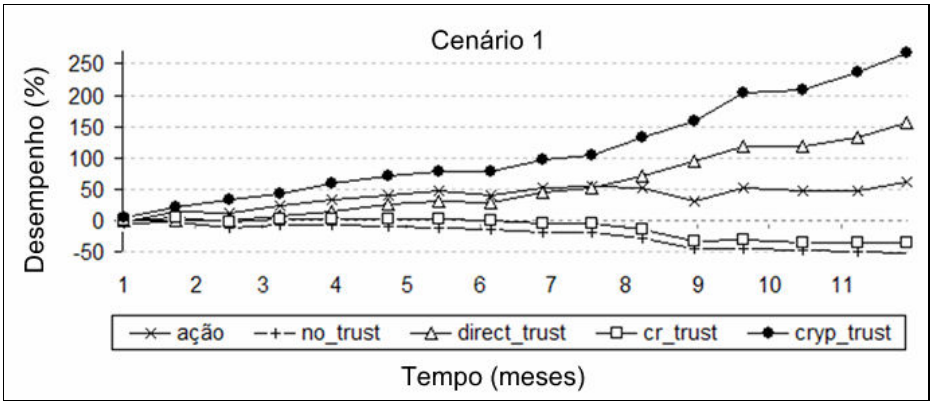

Figura 4.17 Cenário 1, ambiente desonesto sem mudanças.

	Ação	no_trust	direct_trust	cr_trust	cryp_trust
GU Final	61	-52%	156%	-36%	265%

Tabela 4.11 Cenário 1, ganho de utilidade final.

Conforme observado nas últimas três execuções, pela primeira vez o *cr_trust* não consegue acompanhar o desempenho do modelo *cryp_trust*. Este fato ocorre pela introdução de provedores maliciosos, que enviam apenas seus melhores *ratings* e confundem os consumidores *cr_trust*. O modelo direto mantém o mesmo desempenho obtido pelo Cenário 1 do ambiente honesto, à medida que seus agentes não dependem de *ratings* enviados pelos provedores.

O segundo cenário do ambiente desonesto apresenta a situação mais adversa destes experimentos. Além da presença dos provedores maliciosos, os demais

provedores variam constantemente de desempenho. A configuração deste cenário é apresenta pela Tabela 4.12.

Variáveis do Experimento	Símbolo	Valor
Número de pregões	N	473
Número de agentes provedores:	N_P	500
Número de *Good providers*	N_{PG}	125
Número de *Ordinary providers*	N_{OP}	125
Número de *Bad providers*	N_{PB}	125
Número de *Malicious providers*	N_{MI}	125
Número de agentes clientes	N_C	500
Variação de desempenho	V_P	sim

Tabela 4.12 Cenário 2, ambiente desonesto com mudanças.

Semelhante ao Cenário 2 do modelo honesto, a variação de desempenho dos provedores é controlada pelo CRONOS que periodicamente troca o comportamento dos servidores. Os resultados desde experimento são exibidos na Figura 4.18 e Tabela 4.13.

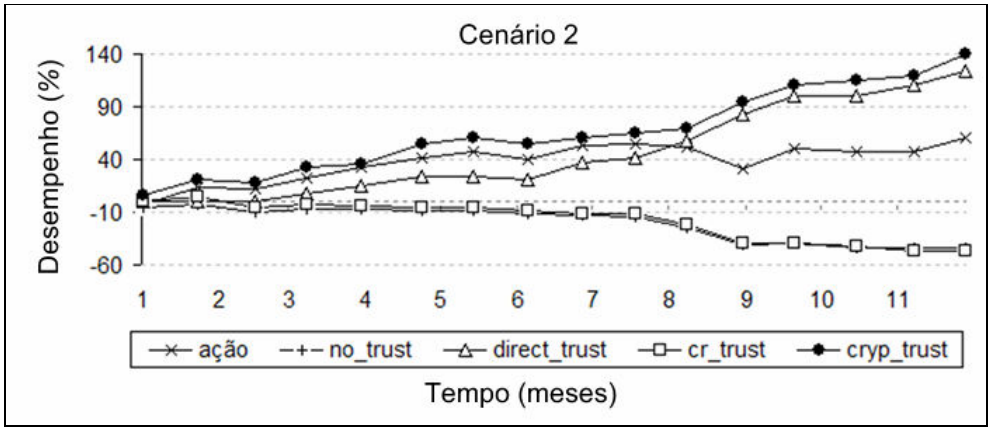

Figura 4.18 Cenário 2, ambiente desonesto com mudanças.

	Ação	no_trust	direct_trust	cr_trust	cryp_trust
GU Final	61	-44%	124%	-46%	140%

Tabela 4.13 Cenário 2, ganho de utilidade final.

Devida a presença de agentes maliciosos e a constante variação dos provedores, o segundo cenário do ambiente desonesto apresenta o pior desempenho para os modelos indiretos em especial o *cr_trust*. Apesar da queda de desempenho do *cryp_trust*, comparado com o primeiro cenário do ambiente desonesto, sua superioridade ainda é mantida em relação aos demais modelos. A proximidade de desempenho do *direct_trust* ao *cryp_trust* se deve a imunidade do modelo direto ao tipo do agente malicioso projetado para este experimento, à medida que sua atuação aplica-se apenas aos

modelos indiretos que necessitam dos *ratings* fornecidos pelos provedores para calcular a confiança.

4.6 Considerações Finais

A concepção do CRONOS como um sistema avaliador de modelos de confiança foi essencial para realização dos experimentos. A definição de quatros cenários aos experimentos proporcionou uma simulação mais próxima de um ambiente real. Fatores como escalabilidade, comportamento dos agentes e a instabilidade de desempenho dos provedores garantiram a abrangência necessária para realizar conclusões a cerca das hipóteses levantas sobre o *modelo de confiança certificado*. Com base nos resultados apresentados para cenário proposto, pode ser observado na Figura 4.19 um comparativo geral entre os modelos de confiança.

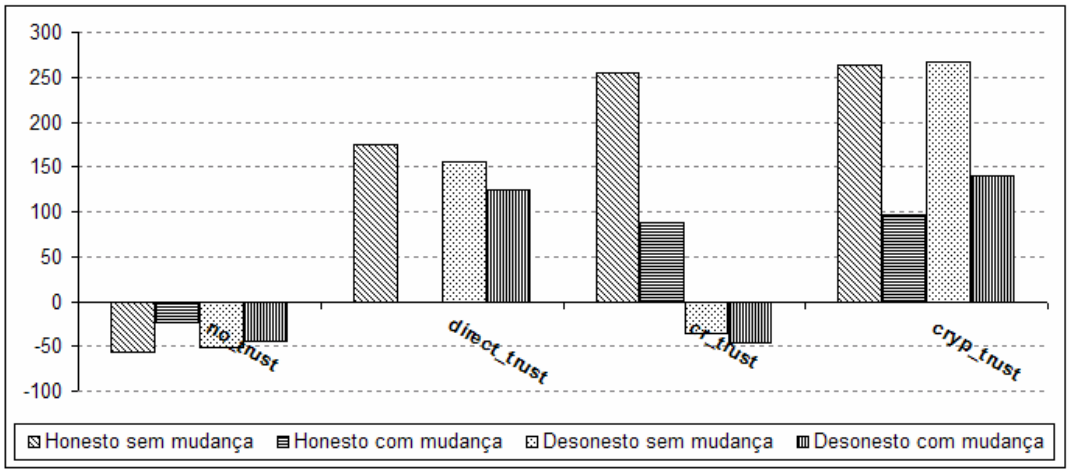

Figura 4.19 Comparativo dos modelos de confiança.

	Ação	No_trust	direct_trust	cr_trust	cryp_trust
Honesto sem mudança	-57%	174%	254%	263%	174%
Honesto com mudança	61	-24%	0%	87%	96%
Desonesto sem mudança	-52%	156%	-36%	267,5%	156%
Desonesto com mudança	-44%	124%	-46,5%	140%	124%

Tabela 4.14 Comparativo dos modelos de confiança.

Numericamente, conforme observado nos comparativos, o *modelo de confiança certificado* se apresentou superior aos demais modelos. Seu mecanismo de cifragem de mensagens para o tratamento de provedores maliciosos garantiu robustez suficiente para seu uso em ambientes multiagente abertos. Apesar da sua vantagem, é possível perceber certa fragilidade do modelo quando exposto em ambientes com alta instabilidade no

comportamento dos agentes. No estudo de caso que envolve sistemas de mercado financeiros, como a bolsa de valores, é fundamental a agilidade dos agentes para detectar, e se possível prever tendências de mercado, e no caso dos modelos de confiança as mudanças de comportamentos dos parceiros de investimento.

O Capítulo 2 apresentou um comparativo entre os modelos de confiança analisados por meio da Tabela 2.3, considerando o CRONOS como um modelo adicional a lista anterior a Figura 4.15 acrescenta o CRONOS de maneira a facilitar a comparação entre o modelo proposto com os demais.

	Paradigma	Origem da Informação	Visibilidade	Dimensões	Comportamento	Tipo de Informação
E-commerce	M	T	G	UC	0	B
ReGret	M	ED +T	L	MC	2	B
FIRE	M	ED + T	L	MC	1	B
S. Marsh	M	ED	L	MC	NA	N
Spora	M	T	G	UC	0	C
Abdul-Rahman e Stephen Hailes	M	ED + T	L	MC	2	B
TRAVOS	M	ED + T	L	MC	1	B
Esfandiary e Chandrasekharan	M	ED + OD + T	L	MC	0	B
Yu e Singh	M	ED + T	L	UC	0	B
AFRAS	M	ED + T	L	UC	2	B
Sen e Sajja	M	ED + OD + T	L	UC	2	B
Castelfranchi e Falcone	C	NA	L	MC	NA	NA
CRONOS	**M**	**ED + T**	**L**	**MC**	**2**	**C**

Tabela 4.15 Comparação dos modelos de confiança.

Portanto, o CRONOS é considerado um modelo matemático que manipula informações diretas e indiretas por meio de testemunhos, utiliza uma representação local de confiança e representa por múltiplos contextos, trata agentes maliciosos e possui uma representação contínua das informações.

Capítulo 5

Conclusão

Confiança e reputação são conceitos essenciais à construção de comunidades virtuais. Em sistemas multiagente abertos, a representação destes conceitos, por meio de modelos computacionais, permite aumentar a confiabilidade dos sistemas, melhorando a qualidade das relações entre os agentes destas comunidades.

O objetivo deste trabalho consistiu na proposta de um modelo de confiança capaz de auxiliar os agentes de uma comunidade a selecionar bons parceiros para futuras interações. Desta forma, foram analisadas as principais limitações dos modelos de confiança descritas na literatura, a fim de propor uma solução que permitisse minimizar ou eliminar alguns destes problemas. Os principais problemas elencados foram: o controle distribuído das informações, a gestão de testemunhas e a garantia de segurança dos canais comunicação.

O gerenciamento distribuído, tanto das informações de confiança, quanto das testemunhas, pode ser classificado como um problema complexo, cujo desafio, consiste respectivamente em descobrir o nível de confiança dos agentes e o nível de credibilidade das testemunhas, por meio das bases de conhecimentos da própria comunidade. A ausência de entidades controladoras resulta em maior disponibilidade do sistema e melhor distribuição dos recursos computacionais. Além disso, o tratamento da segurança sobre os canais de comunicação permite maior confiabilidade nas relações entre os agentes.

Este trabalho apresentou uma proposta aos problemas elencados por meio da construção do *modelo de confiança certificado baseado em assinatura digital* que evoluiu a abordagem da reputação certificada garantindo maior integridade das informações, mesmo na presença de agentes maliciosos, e definiu um tratamento

específico de segurança por meio de assinaturas digitais. Com relação à reputação certificada, este trabalho tratou sua principal lacuna, a ausência de tratamento a agentes maliciosos, promovendo uma estratégia para impedir que os agentes provedores manipulem suas avaliações.

Além disso, este trabalho contribuiu para a construção da ferramenta CRONOS, cujo objetivo foi viabilizar os experimentos a cerca dos modelos de confiança estudados. Utilizando um estudo de caso específico, o mercado de financeiro de ações, a ferramenta proveu um ambiente multiagente aberto, capaz de suportar as diversas abordagens de confiança e disponibilizar as informações necessárias para a avaliação dos experimentos. Modelada como uma ferramenta extensível, o CRONOS permite reutilizar sua estrutura para a construção de novos casos de estudo.

Apesar deste trabalho não ter como objetivo a análise dos mercados de ações, foi possível perceber, no caso de estudo tratado, que os modelos de confiança melhoraram a rentabilidade dos agentes em seus investimentos. Isto reafirma a utilidade da confiança em problemas práticos do mundo real.

Os resultados apresentados ao longo dos experimentos mostram que em comunidades simples, nas quais não há a presença de agentes mal intencionados e não há grandes variações no comportamento dos agentes, as abordagens de confiança possuem eficiências similares. Todavia, a medida que os cenários se tornam mais complexos, seja pela imprevisibilidade do ambiente, como em um mercado de ações, ou pelo mal comportamento dos agentes, em ambientes abertos, a qualidade dos modelos de confiança diminuem. Foi possível minimizar os impactos negativos deste tipo de cenário por meio do modelo de confiança certificado. A capacidade de impedir que os agentes selecionem arbitrariamente suas avaliações ou que modifiquem as informações trafegadas entre eles, permitiu, ao nosso modelo, maior precisão no cálculo da confiança e, portanto melhor eficiência em cenários maliciosos.

5.1 Limitações

As principais limitações deste trabalho podem ser destacadas nos seguintes tópicos:

1. A construção de modelos baseada exclusivamente nas abordagens matemáticas, não proveu análise comparativa contra os modelos cognitivos.

2. Os experimentos não abordaram outros critérios comparativos como desempenho e uso de recursos computacionais.
3. A utilização de um único estudo de caso impediu a análise dos modelos de confiança em outros cenários que poderiam levar as novas conclusões.
4. A ausência de técnicas mais sofisticadas para detectar a variação de comportamento dos agentes.
5. A ausência de ontologias que facilitassem a representação do conhecimento compartilhado entre os agentes.

5.2 Trabalhos Futuros

Como trabalhos futuros, destacamos primeiramente a revisão dos conceitos introduzidos neste trabalho, a fim de garantir mais robustez à abordagem; na seqüência poderia ser definida uma estratégica para aprimoramento do modelo de confiança certificado no tratamento de ambientes com alta variação de comportamento; novos experimentos poderiam abranger questões de desempenho que verificassem o impacto dos mecanismos de criptografia das mensagens; para enriquecer as discussões a cerca do modelo proposto, novos estudos poderiam agregar novas abordagens de confiança como os modelos probabilísticos de Teacy, *et al.* (2005) ou cognitivos de Castelfranchi e Falcone, (1998). Outras características poderiam ser incluídas ao modelo como: a criação de ontologias específicas para confiança (Mui, *et al.*, 2002); construção de novos estudos de caso a partir do sistema CRONOS e a ampliação ao tratamento de clientes maliciosos.

Referências Bibliográficas

[ABD00] ABDUL-RAHMAN, A.; HAILES, S.: *Supporting Trust in Virtual Communities*. In Proceedings of the 33rd Hawaii international Conference on System Sciences-Volume 6. IEEE Computer Society, Washington, 2000.

[ABE01] ABERER, K.; DESPOTOVIC, Z. *Managing trust in a peer-2-peer information system*. In Proceedings of the Tenth international Conference on information and Knowledge Management, Atlanta, Georgia, USA, 2001.

[AMA09] Amazon (2009). *Amazon Auctions*. http://auctions.amazon.com. Acesso em: 09 de novembro de 2008. às 20:00hs

[AND92] ANDREONI, J.; MILLER, J. H. "*Rational Cooperation in the Finitely Repeated Prisoner's Dilemma: Experimental Evidence*", The Economic Journal, 103 (418), pp. 570-585. 1992

[ASH05] ASHRI, R.; RAMCHURN, S. D.; SABATER, J.; LUCK, M.; JENNINGS, N. R. *Trust evaluation through relationship analysis*. In Proceedings of the Fourth international Joint Conference on Autonomous Agents and Multiagent Systems, The Netherlands, July, 2005.

[AXE84] AXELROD, R. *The Evolution of Cooperation*. New York: Basic Books.

[BEL07] BELLIFEMINE, F; CAIRE, G; GREENWOOD, D. *Developing multi-agente systems with JADE*, John Wiley & Sons Ltd, England, ISBN: 978-0-470-05747-6, 2007

[BER01] BERNERS-LEE, T.; HENDLER, J.; LASSILA, O. *The semantic web*. Scientific American 284(5), 34–43, 2001.

[BIM92] BINMORE, K. *Fun and Games: A Text on Game Theory*. D. C. Heath and Company, 1992.

[BON88] BOND, A. H.; GASSER, L. *An Analysis of Problems an Research in DAI*, San Mateo, CA, 1988, pg 3-35.

[BOT09a] BOTELHO, V. S.; ENEMBRECK, F.; ÁVILA, B. C.; AZEVEDO, H; SCALABRIN, E. *Encrypted Certified Trust in Multi-Agent System*; in 13TH International Conference on CSCS in Design, April 22-24, 2009, Santiago, Chile

[BOT09b] BOTELHO, V. S.; ENEMBRECK, F.; ÁVILA, B. C.; AZEVEDO, H; SCALABRIN, E. *Certified Trust Model*; in 5th IFIP Conference on Artificial Intelligence Applications & Innovations, April 23-25, 2009, Thessaloniki, Grécia.

[BOV08] BOVESPA, Web Site da Bolsa de Valores de São Paulo. http://www.bovespa.com.br, acesso em 01 de fevereiro de 2009 às 22:00hs.

[BUR82] BURT, R. S. *Toward a Structural Theory of Action. Network Models of Social Structure, Perception, and action*. New York: Academic Press. 1982.

[BUS98] BUSKENS, V. *The social structure of trust*. Social Networks 20, 265–298, 1998.

[BRA97] BRADSHAW, J. M.; DUTFIELD, S.; BENOIT, P.; WOOLEY, J. D. *KAoS: Toward an Industrialstrength Open Agent Architecture*. In J. M. Bradshaw, editor, Software Agents, pages 375–418. AAAI Press/MIT Press, Cambridge, MA, 1997.

[BRA96] BRANAGAN, J.; IPPOLITO, K.; MUSGRAVE L.; WAGGENSPACK, W "*Pretty good privacy*", The Art and interdisciplinary Programs of SIGGRAPH'96, New Orleans, United States, 1996.

[BRO93] BROMLEY, D. B. *Reputation, Image and Impression Management*. John Wiley & Sons, 1993

[BRO85] BROOKS, R. A. *A Robust Layered Control System for a Mobile Robot*. Technical Report. UMI Order Number: AIM-864., Massachusetts Institute of Technology, 1985.

[CAR02] CARBO, J.; MOLINA, J.; DAVILA, J. *Trust Management Through Fuzzy Reputation*. Int. J. in Cooperative Information Systems, 12(1):135--155, 2002.

[CAS98] CASTELFRANCHI, C.; FALCONE, R. *Principles of Trust for MAS: Cognitive Anatomy, Social Importance, and Quantification*, In Proceedings of the 3rd International Conference on Multi Agent Systems, p.72, 1998.

[CHEN86] CHEN, S. S. *Evidential logic and Dempster-Shafer theory*. In Proceedings of the ACM SIGART international Symposium on Methodologies For intelligent Systems. Z. W. Ras and M. Zemankova, Eds. ACM, New York, NY, 1986

[CLA98] CLAUS, C. ; BOUTILIER, C. *The dynamics of reinforcement learning in cooperative multiagent systems*. Proceedings of the Fifteenth National Conference on Artificial Intelligence (pp. 746--752). Menlo Park, CA, USA: AAAI, 1998.

[COH90] COHEN, P. R.; LEVESQUE, H. J. *Intention is choice with commitment*. Artificial Intelligence 42(2–3), 213–261, 1990.

[DAS98] DASGUPTA, P. *Trust as a commodity*. In Gambetta, D. (ed.), Trust: Making and Breaking Cooperative Relations. Blackwell, pp. 49–72, 1998.

[DAV00a] D. Davis, Y. A. Luo, Using KADS to Design a Multi-Agent Framework for Stock Trading, Department of Computer Science, University of Hull, 2000.

[DAV00b] D. Davis, Y. A. Luo, Multi-Agent Framework for Stock Trading, Department of Computer Science, University of Hull, 2000.

[DAV02] D. Davis, Y. A. Luo, A Multi-Agent Decision Support System for Stock Trading, IEEE Network, 2002

[DEI01] DEITEL, M. H.; DEITEL, P. J.; NIETO, T. R. *E-business and E-commerce: How to Program*. Prentice-Hall, Englewood Cliffs, NJ. 2001.

[DEN84] DENNING, D. E.; *Digital signatures with RSA and other public-key cryptosystems*. 1984.

[DUR89] DURFEE, E. H.; LESSER, V. *Negotiating task decomposition and allocation using partial global planning*. In L. Gasser and M. Huhns, editors, Distributed Artificial Intelligence Volume II. Pitman Publishing: London and Morgan Kaufmann: San Mateo, CA, 1989.

[DUR94] DURFEE, E. H.; ROSENSCHEIN, J. S. *Distributed Problem Solving and Multi-Agent Systems: Comparisons and Examples*. In Proceedings of The International Workshop on Distributed Artificial Intelligence, 1994.

[EBA09] ebay (2009). eBay. http://www.ebay.com, acesso em 05 de outubro de 2008 às 22:00hs.

[ELB00] ELBIRT, A. J. AND PAAR, C. *An FPGA implementation and performance evaluation of the Serpent block cipher*. In Proceedings of the 2000 ACM/SIGDA Eighth international Symposium on Field Programmable Gate Arrays. FPGA '00. ACM, New York, NY, 33-40, 2000.

[ESF01] ESFANDIARI, B; CHANDRASEKHARAN, S. *On How Agents Make Friends: Mechanisms for Trust Acquisition*. In Proceedings of the Fifth International Conference on Autonomous Agents Workshop on Deception, Fraud and Trust in Agent Societies, 2001.

[FER99] FERBER, J. *Multi-Agent System, An Introduction to Distributed Artificial Intelligence*, Addison-Wesley, Boston, MA, 1999.

[FIP09] FIPA, Foundation for Intelligent Physical Agents, http://www.fipa.org, acesso em 15 de dezembro de 2008 às 22:00hs.

[FIP02a] Foundation for Intelligent Physical Agents, *FIPA Abstract Architecture Specification*, http://www.fipa.org/specs/fipa00001/, 2002, acesso em 15 de dezembro de 2008 às 22:00hs.

[FIP02b] Foundation for Intelligent Physical Agents, *FIPA Contract Net Interaction Protocol Specification*, http://www.fipa.org/specs/fipa00029/, 2002, acesso em 15 de dezembro de 2008 às 22:00hs.

[FIP02c] Foundation for Intelligent Physical Agents, FIPA ACL Message Structure Specification, http://www.fipa.org/specs/fipa00061/, 2002, acesso em 15 de dezembro de 2008 às 22:00hs.

[FRI02] FRITSCHI, C. e DORER, K. *Agent-oriented software engineering for successful TAC participation*. In Proceedings of the First international Joint Conference on Autonomous Agents and Multiagent Systems, Bologna, Italy, 2002.

[FOR08] *Ease Forex, Basic Forex forecast methods: Technical analysis and fundamental analysis*, Web site, https://easy-forex.com/en/Forex.forecast.aspx, Acesso em março de 2009.

[FOS98] FOSTER, I.; KESSELMAN, C. (eds.), *The Grid, Blueprint for a New Computing Infrastructure*, Morgan Kaufmann Inc., 1998.

[FUL05a] FULLAM, K. K.; KLOS, T. B.; MULLER, G.; SABATER, J.; SCHLOSSER, A.; TOPOL, Z.; BARBER, K. S.; ROSENSCHEIN, J. S.; VERCOUTET, L.; VOSS, M. *A specification of the Agent Reputation and Trust (ART) testbed: experimentation and competition for trust in agent societies*. In Proceedings of the Fourth international Joint Conference on Autonomous Agents and Multiagent Systems, Netherlands, 2005.

[FUL05b] FULLAM, K. K.; KLOS, T. B.; MULLER, G.; SABATER, J.; SCHLOSSER, A.; TOPOL, Z.; BARBER, K. S.; ROSENSCHEIN, J. S.; VERCOUTET, L. *The Agent Reputation and Trust (ART) Testbed Architecture*. In Proc. Trust Workshop at AAMAS, 2005.

[GOL85] GOLDWASSER, S.; MICALI, S.; RACKOFF, C.; *The knowledge complexity of interactive proof-systems*. In Proceedings of the Seventeenth Annual ACM Symposium on theory of Computing, United States, 1985.

[GOY06] GOYAL, V.; PANDEY, O.; SAHAI, A.; WATERS, B.; *Attribute-based encryption for fine-grained access control of encrypted data*. In Proceedings of the 13th ACM Conference on Computer and Communications Security, Alexandria, Virginia, USA, 2006.

[GRI05] GRIFFITHS, N. *Task Delegation using Experience-Based Multi-Dimensional Trust*, in Proceedings of the 4th International Conference on Autonomous Agents and Multiagent Systems, Utrecht, Netherlands 2005.

[HAL01] HALBERSTADT, A; MUI, L. "*Group and Reputation Modeling in Multi-Agent Systems*" Proc. Goddard/JPL Workshop on Radical Agents Concepts, NASA Goddard Space Flight Center, 2001.

[HAY95] HAYES-ROTH, B. *An Architecture for Adaptive Intelligent Systems,* Artificial Intelligence. Special Issue on Agents and Interactivity, pg 329-365, 1995.

[HOF07] HOFFMAN, N. 2007. *A Simplified IDEA Algorithm*. Cryptologia 31, 2, 143-151. DOI= http://dx.doi.org/10.1080/01611190701215640, 2007.

[HUI04] HUIPING, J.; RUI, X.; SHENG, B. *Advanced DES Algorithm against Differential Power Analysis and its Hardware Implementation*. In Proceedings of the the First international Symposium on Data, Privacy, and E-Commerce (November 01 - 03, 2007). ISDPE. IEEE Computer Society, Washington, DC, 316-320. In Proc. 7th Int. Workshop on Trust in Agent Societies, pages 62–77, 2004.

[HUY06] HUYNH, T. D.; JENNINGS, N. R.; SHADBOLT, N. R. *Certified reputation: how an agent can trust a stranger*. In Proceedings of the Fifth international Joint Conference on Autonomous Agents and Multiagent Systems, Hakodate, Japan, May, 2006.

[JAD09] Jade – Java Agent Development Environment, http://jade.cselt.it/, 2009.

[JEN93] JENNINGS, N. R. *Commitments and conventions: the foundation of coordination in multi-agent systems*. The Knowledge Engineering Review 8(3), 223–250, 1993.

[JEN96] JENNINGS, N. R. *Coordination techniques for distributed artificial intelligence*. In: O'HARE, G.M.P.; JENNINGS, N. R. (Eds.). Foundations of distributed artificial intelligence. New York: John Wiley & Sons, p.187- 210, 1996

[JEN98] JENNINGS, N. R.; SYCARA, K; WOOLDRIDGE, M. *A Roadmap of Agent Research and Development, Autonomous Agents and Multi-Agent Systems*, 1:7-38, 1998.

[LAI95] LAIH, C.; YEN, S. 1995. *Improved Digital Signature Algorithm*. IEEE Trans. Comput. 44, 729-730. DOI= http://dx.doi.org/10.1109/12.381963, 1995.

[JUR06] JURCA, R.; FALTINGS, B. *Using CHI-scores to reward honest feedback from repeated interactions*. In Proceedings of the Fifth international Joint Conference on Autonomous Agents and Multiagent Systems, Japan, May, 2006.

[KAT53] KATZ, L. "*A New Status Index Derived from Sociometric Analysis*", Psychometrika, 18, pp. 39-43, 1953.

[KER83] KERCKHOFFS, A. "*La cryptographie militaire*", Journal des sciences militaires, vol. IX, pp. 5–83, Jan. 1883, pp. 161–191, Feb. 1883.

[KRA01] KRAUS, S. *Strategic Negotiation in Multi-Agent Environments*. Cambridge, MA: MIT Press, 2001.

[KREPS82] KREPS, D. M.; WILSON, R. *Sequential Equilibria*, Econometric Society, vol. 50(4), pages 863-94, July.

[LEE97] LEE, J., HEYS, H. M., AND TAVARES, S. E. *Resistance of a CAST-Like Encryption Algorithm to Linearand Differential Cryptanalysis*. Des. Codes Cryptography 12, 3, 267-282, 1997.

[LEE02] LEE, J. W.; O, J. *A Multi-agent Q-learning Framework for Optimizing Stock Trading Systems*. In Proceedings of the 13th international Conference on Database and Expert Systems Applications, September 02 - 06, A. Hameurlain, R. Cicchetti, and R. Traunmüller, Eds. Lecture Notes In Computer Science, vol. 2453. Springer-Verlag, London, 153-162, 2002.

[LU02] LU, C. AND TSENG, S. 2002. *Integrated Design of AES (Advanced Encryption Standard) Encrypter and Decrypter*. In Proceedings of the IEEE international Conference on Application-Specific Systems, Architectures, and Processors. ASAP. IEEE Computer Society, Washington, DC, 277, 2002.

[LIN00] LIN, M. C.; LIN, Y. 2000. *A VLSI implementation of the blowfish encryption/decryption algorithm*. In Proceedings of the 2000 Asia and South Pacific Design Automation Conference (Yokohama, Japan). ACM, New York, NY, 2000.

[LUH79] LUHMANN, N. "*Trust: A Mechanisk for the Reduction of Social Complexity*." In Trust and Power: Two Works by Niklas Luhann. Reprint, New York, John Wiley & Sons, 1979.

[NOR08] *Normas Brasileiras de Contabilidade*, http://www.portaldecontabilidade.com.br/nbc/index.htm, World Wide Web, acesso em 15 de março de 2009 às 21:00hs.

[MAR94] MARSH, S. P. Formalising *Trust as a Computational Concept*. Department of Computing Science and Mathematics University of Stirling, England, Submitted in partial fulfilment of the degree of Doctor of Philosophy 1994.

[MAS01] MASS, Y.; SHEHORY, O. 2001. *Distributed Trust in Open Multi-agent Systems*. In Proceedings of the Workshop on Deception, Fraud, and Trust in Agent Societies Held During the Autonomous Agents Conference: Trust in Cyber-Societies, integrating the Human and Artificial Perspectives R. Falcone, M. P. Singh, and Y. Tan, Eds. Lecture Notes In Computer Science, vol. 2246. Springer-Verlag, London, 159-174.

[MAT06] MATSURA, E. *Comprar ou vender? Como investir na Bolsa utilizando análise gráfica*, 4ª edição. – São Paulo, 2006.

[MAX05] MAXIMILIEN, E. M.; SINGH, M. P. *Agent-based trust model involving multiple qualities*. In Proceedings of the Fourth international Joint Conference on Autonomous Agents and Multiagent Systems, The Netherlands, July 25 - 29, 2005.

[MED01] *Medida Provisória nº 2.200-2, de 24 de agosto de 2001*, Presidência da República, Casa Civil, Brasil, 2001, http://www.planalto.gov.br/ccivil_03/MPV/Antigas_2001/2200-2.htm, acesso em 15 de março de 2009 às 21:00hs.

[MEN96] MENEZES, A. J.; VANSTONE, S. A.; OORSCHOT, P. C. V. *Handbook of Applied Cryptography*. CRC Press, Inc., Boca Raton, FL, USA, 1996.

[MER09] Mercado Livre (2009). http://www.mercadolivre.com.br, acesso em 05 de janeiro de 2009 às 10:00hs.

[MEY89] MEYER, C.H. *Cryptography-a state of the art review*. In VLSI and Microelectronic Applications in Intelligent Peripherals and their Interconnection Networksapos, Proceedings, Volume, Issue, Pages: 4/150 - 4/154, 1989.

[MIN86] MINSKY, M. *The Society of Mind*. Simon & Schuster, Inc, 1986.

[MOR03] MORETTIN, P. A.; BUSSAB, W. O. *Estatística Básica*, Editora Saraiva, ISBN 8502034979, São Paulo, 2003.

[MUI02] MUI, L.; MOHTASHEMI, M.; HALBERSTADT, A. *Notions of reputation in multi-agents systems: a review*. In Proceedings of the First international Joint Conference on Autonomous Agents and Multiagent Systems: Part 1, Bologna, Italy, 2002.

[MUR86] MURPHY, J. *Technical Analysis of the Futures Markets*, New York: New York Institute of Finance, 1986.

[NOW98] NOWAK, M. A.; SIGMUND, K. (1998) "*Evolution of Indirect Reciprocity by Image Scoring*" Nature, 393, pp. 573-577.

[OLI07] OLIVEIRA, M. C., CIRNE, W., MENDES, J. F., and MARQUES, P. M. *Grid computing to make viable the content based medical image retrieval through the image registration techniques*. In Proceedings of the 2007 Euro American Conference on Telematics and information Systems (Faro, Portugal, May 14 - 17, 2007).

[ONS09] OnSale (2002). *OnSale*. http://www.onsale.com, acesso em 25 de novembro de 2008 às 23:00hs.

[ORA01] ORAM, A. *Peer-to-Peer: Harnessing the Power of Disruptive Technologies*. Sebastopol, CA: O'Reilly & Associates Inc., 2001.

[PAN01] PANZARASA, P.; JENNINGS, N. R.; NORMAN, T. *Social mental shaping: modelling the impact of sociality on the mental states of autonomous agents*. Computational Intelligence 4 (17), 738–782, 2001.

[POL92] POLLOCK, G. B.; DUGATKIN, L. A. "*Reciprocity and the Evolution of Reputation*" Journal of Theoretical Biology, 159, pp. 25-37, 1992.

[POO05] POONPHON, S.; KIM, Y.; LEE, C., e SADANANDA, R. *Interactions and Cooperation between Machines as a Society of Agents*, In Proceedings of the Fourth Annual ACIS international Conference on Computer and information Science, Washington, 2005.

[PRE93] PRENEEL, B.; GOVAERTS, R.; VANDEWALLE, J. *Cryptographic hash functions: an overview*, In Proceedings of the 6th International Computer Security and Virus Conference (ICSVC 1993), 19 pages, 1993

[PUJ02] PUJOL, J. M.; SANGESA, R.; DELGADO, J. *Extracting reputation in multi-agent system by means of social network topology*. In Proceedings of the firs international joint conference on autonomous agents and multiagente system (AAMAS-02), Bologna, Italy, pages 456-474, 2002.

[PUL03] PUJOL, J. M.; SANGESA, R.; DELGADO, J. Web Intelligence, chapter A Ranking Algorithm Based on Graph Topology to Generate Reputation or Relevance. Spring Verlag. 2003.

[RAO95] RAO, A. S.; GEORGE, M. P.; *BDI Agents: From Theory to Practice*, Technical Note 56, Australian Artificial Intelligence Institute, Australian, 1995.

[RAM04] RAMCHURN, S. D.; HUNYH, D.; JENNINGS, N. R. *Trust in multi-agent systems*. Knowledge Engineering Review, 19(1):1–25, 2004.

[RAR81] BARR, A.; FEIGENBAUM, E. A. *The Handbook of Artificial Intelligence*, vol I. Pitman Books Limited, 1981.

[RES00] RESNICK, P.; ZECKHAUSER, R.; FRIEDMAN, E.; KUWABARA, K. *Reputation Systems. In: Communications of the ACM*, vol. 43, number 12, p. 45-48, 2000.

[REZ00] REZENDE, P. A. D. *Criptografia e Segurança da Informação*, Universidade de Brasília, Editora CopyMarket.com, 2000.

[REZ05] REZENDA, S. *Sistemas Inteligentes: Fundamentos e Aplicações*, Barueri, SP, 2005.

[RIV79] RIVEST, R.; SHAMIR, A.; ADLEMAN, L. *A Method for Obtaining Digital Signatures and Public-Key Cryptosystems*. Communications of the ACM, 21 (2), pp. 120-126, February 1978.

[ROL06] ROLT, C. R.; BROCARDO, M. L.; FERNANDES. R. *Introdução à Certificação Digital Da Criptografia ao Carimbo de Tempo*, Editora BRy Tecnologia – 1ª edição, 2006.

[ROS94] ROSENSCHEIN, J.; ZLOTKIN, G. *Rules of Encounter: Designing Conventions for Automated Negotiation among Computers*. Cambridge MA: MIT Press, 1994.

[RUS95] RUSSEL, S.; NORVIG P.; *Artificial Intelligence - A Moderm Approach. Prentice Hall*, Inc. 1995, pg 4-5.

[SAB03] SABATER, J. *Trust and Reputation for Agent Societies*. PhD thesis, Universitat Autònoma de Barcelona, 2003.

[SCH02] SCHMITZ, M. e HÜBNER, J. F. *Uso de SMA para avaliar estratégias de decisão no controle de tráfego urbano*. In Anais do XI Seminário de Computação, Blumenau, 2002, pg 243-254.

[SCH06] SCHWEITZER, D. AND BAIRD, L. *Discovering an RC4 anomaly through visualization*. In Proceedings of the 3rd international Workshop on Visualization For Computer Security, VizSEC '06. ACM, New York, NY, 91-94, 2006

[SCH99] SCHNEIER, B.; KELSEY, J.; WHITING, D.; WAGNER, D.; AND HALL, C. *On the Twofish Key Schedule*. In Proceedings of the Selected Areas in Cryptography. S. E. Tavares and H. Meijer, Eds. Lecture Notes In Computer Science, vol. 1556. Springer-Verlag, London, 27-42. 1999

[SEL78] SELTEN, R. "*The Chain Store Paradox*" Theory and Decision, 9, pp. 127-159, 1978.

[SEN02] SEN, S; SAJJA, N. *Robustness of reputation-based trust: boolean case*, In Proceedings of the First international Joint Conference on Autonomous Agents and Multiagent Systems: Part 1 (Bologna, Italy, July 15 - 19, 2002). AAMAS '02. ACM, New York, NY, 2002

[SEW07] SEWELL, M. *Technical Analysis*, Department of Computer Science, University College London, 2007.

[SHM99] SHMEIL, M. A. H. *Sistemas Multiagente na Modelação da Estrutura e Relações de Contratação de Organizações*, Universidade do Porto, Portugal, 1999.

[SIE01] SABATER, J.; SIERRA, C. *Regret: A reputation model for gregarious societies*. In Proceedings of the Fourth Workshop on Deception, Fraud and Trust in Agent Societies, Montreal, Canada, pages 61—69, 2001.

[SIE05] SIERRA, C.; DEBENHAM, J. *An InformationBased model for Trust*. In Proceedings of the Fourth international Joint Conference on Autonomous Agents and Multiagent Systems, Netherlands, 2005.

[SIL08] SILVA, T. H.; MACHARET, D. G.; TEIXEIRA, C. F. *Análise de Desempenho da Algoritmos Criptográficos em Dispositivos Móveis*, Wperformance Workshop de Desempenho de Sistemas Computacionais e de Comunicação, Anais do XXVIII Congresso da SBC, Belém, 2008.

[SIM79] SIMMONS, G. J. 1979. *Symmetric and Asymmetric Encryption*. ACM Comput. Surv. 11, 4 (Dec. 1979), 305-330. DOI= http://doi.acm.org/10.1145/356789.35679

[SIN01] SINGH, S. *O Livro dos Códigos*, Editora Record, 1º Edição, ISBN 8501055980, São Paulo, 2001.

[SPI85] SPIEGEL, M. R.; Estatística matemática, McGraw-Hill do Brasil, São Paulo, 1985.

[TEA05] TEACY, W. T.; PATEL, J.; JENNINGS, N. R.; LUCK, M.; *Coping with inaccurate reputation sources: experimental analysis of a probabilistic trust model*. In Proceedings of the Fourth international Joint Conference on Autonomous Agents and Multiagent Systems, Netherlands, 2005.

[TEA06] TEACY, W. T.; PATEL, J.; JENNINGS, N. R.; LUCK, M. *TRAVOS: Trust and Reputation in the Context of Inaccurate Information Sources*. Autonomous Agents and Multi-Agent Systems 12, 2006.

[TER00] TERADA, R. *Segurança de Dados. Criptografia em Redes de Computadores*, Editora Edgard Blücher Ltda, 2000.

[WANG02] WANG, H.; MYLOPOULOS, J.; LIAO, S. *Intelligent agents and financial risk monitoring systems*. Commun. ACM 45, 3 (Mar. 2002), 83-88.

[WEN06] WENG, J.; MIAO, C.; GOH, A.; SHEN, Z.; GAY, R. *Trust-based agent community for collaborative recommendation*. In Proceedings of the Fifth international Joint Conference on Autonomous Agents and Multiagent Systems, Hakodate, Japan, May, 2006.

[WIN95] WINGER, B.; FRASCA, R. *Investments: Introduction to analysis and Planning*, 3. ed. Englewood Cliffs: Prentice Hall, 1995.

[WIN07] WINIKOFF, M. *Implementing commitment-based interactions*. In Proceedings of the 6th international Joint Conference on Autonomous Agents and Multiagent Systems, New York, 2007.

[WOO02] WOOLDRIDGE, M. "*An Introduction to MultiAgent Systems*", Ed. John Wiley&Sons, Inglaterra, 2002.

[YU00] YU, B.; SINGH, M. P. *A Social Mechanism of Reputation Management in Electronic Communities*. In Proceedings of the 4th international Workshop on Cooperative information Agents Iv, the Future of information Agents in Cyberspace. M. Klusch and L. Kerschberg, Eds. Lecture Notes In Computer Science, vol. 1860. Springer-Verlag, London, 154-165, 2000.

[ZAC99] ZACHARIA, G.; MAES, P. "*Collaborative Reputation Mechanisms in Electronic Marketplaces.*" Proc. 32nd Hawaii International Conf on System Sciences, 1999.

[ZHE06] ZHENG, X.; WU, Z.; CHEN, H.; MAO, Y. *Developing a composite trust model for multi-agent systems*. In Proceedings of the Fifth international Joint Conference on Autonomous Agents and Multiagent Systems, Hakodate, Japan, May, 2006.

Apêndice A

Mercado Financeiro

Este apêndice visa examinar os conceitos envolvidos no estudo desta pesquisa e abordar trabalhos que tenham correlação com a mesma. A análise crítica aos trabalhos selecionados visa mostrar méritos ou restrições que possam contribuir para o desenvolvimento da solução proposta ao problema descrito neste projeto.

A.1. Mercado de Ações

A análise de ativo é o estudo da viabilidade, estabilidade e rentabilidade de um determinado ativo. Em contabilidade, ativos são todos os bens, direitos e valores a receber de uma entidade (Norma, 2008). Os ativos podem ser deste estoques, caixa e equipamentos a terrenos e construções imobiliárias. Outros exemplos importantes de ativo são as ações, definidas como um valor mobiliário, emitido pelas companhias, representativo de parcela do capital (Bovespa, 2008). Este ativo representa a menor parcela em que se divide o capital da companhia. Quando emitidas por companhias abertas, são negociadas em bolsa de valores que fazem movimentar o que chamamos de mercado de ações.

Desde o início das operações no mercado financeiro há o debate de idéias entre as duas principais correntes de análise do mercado de ações: a análise fundamentalista e a análise técnica. Ambas as abordagens visam determinar o preço justo das ações, porém a metodologia utilizada por elas é bastante diferente.

Os fundamentalistas fazem uso de variáveis que podem de certa forma influenciar o mercado e conseqüentemente o preço das ações. Em contraposição os grafistas ou analisadores técnicos, se baseiam no estudo gráfico das cotações da bolsa. Para os técnicos a determinação do preço das ações é influenciada por fatores psicológicos dos investidores que

podem ser reconhecidos pela análise do comportamento histórico do preço e volume das ações no passado.

A.1.1 Análise Fundamentalista

> *"A Análise fundamentalista é um método de prever os movimentos de preço futuros de um ativo baseado em fatores econômicos, políticos, ambientais e outros fatores relevantes que irão afetar a oferta e demanda de um mercado".* (Forex, 2008)

Para os fundamentalistas, o valor justo de uma ação é definido pela capacidade de geração de lucros de uma empresa no futuro. A premissa da análise fundamentalista é que existe uma correlação entre o valor real da ação e seu valor de mercado. Quando o valor de mercado está acima do valor real indica-se o momento de venda, quando o valor de mercado está abaixo do valor real indica-se o momento de compra.

Para determinar o valor real de uma ação, os analistas fundamentalistas utilizam diversas informações, desde o patrimônio da empresa até sua posição no respectivo setor de atuação. Entretanto, para garantir maior amplitude da análise Winger, (1995), sugere que a análise seja alicerçada por de três fatores:

- **Ambiente interno**: visa analisar informações internas da empresa como a avaliação histórica do seu desempenho econômico; projeções de vendas para os próximos anos; evolução dos preços; evolução dos custos e despesas.
- **Ambiente externo**: visa analisar aspectos do ambiente no qual a organização está inserida, como projeção de concorrência; mudanças de tecnologia; competitividade; questões relacionadas a barreiras protecionistas.
- **Ambiente macroeconômico**: visa analisar questões de interesse nacional e mundial como as taxas de crescimento do PIB; projeções para as taxas de juros, inflação e risco; taxa de desemprego entre outros.

Diante dos inúmeros fatores que influenciam a demanda e a oferta de um mercado, alguns deles são imprevisíveis como guerras, greves e fenômenos climáticos. O analista fundamentalista tem assim a difícil missão de tratar toda ou parte destas informações de forma a construir uma equação que permita determinar as tendências futuras no preço das ações.

Uma das críticas a esta abordagem está na subjetividade da interpretação das informações, pois um fator considerado positivo para alguns pode ser visto de forma negativa

para outros. Outra restrição da análise fundamentalista está na incerteza quanto a confiabilidade das informações por conta da especulação que ronda este meio.

A.1.2 Análise Técnica

> *"Análise técnica é a previsão dos preços de mercado por meio da análise dos dados gerados pelo processo de negociação."* (Sewell, 2007)

A análise técnica ou análise gráfica é a abordagem que faz uso de gráficos para definir os melhores momentos de compra e venda de um ativo. Esta avaliação é fundamentada por uma série de teorias que tentam prever o movimento do mercado. Porém, todos partem do princípio que os fatores econômicos, políticos, psicológicos entre outros condicionam os preços do mercado, i.e., tudo está embutido no comportamento do preço do ativo.

A análise técnica teve origem no início do século XX através das publicações do reporter Charles Dow, fundador do jornal "*The Wall Street Journal*". Os artigos publicados na coluna de economia descreviam o comportamento do mercado financeiro da época. Após sua morte seus escritos foram organizados e publicados no formato de um livro denominado "*The ABC of Stock Speculation*" que deu origem a chamada Teoria de Dow.

A.1.3 Teoria de Dow

A base conceitual da análise técnica é fundamentada nos princípios da teoria de Dow. O entendimento destes princípios é a essência para manter a coerência na interpretação dos gráficos. A Teoria de Dow é descrita por Murphy (1986) em cinco princípios:

I. ***O preço desconta tudo***: todos os fatores que podem influenciar na procura e oferta de um mercado estão embutidos no preço. Portanto, toda a análise fundamentalista está contida nos gráficos. As tendências de alta ou baixa podem ser conseqüências de fatores políticos, econômicos ou ambientais, porém não importa a causa, pois toda informação relevante está contida no preço.

II. ***O preço move-se em tendência***: o movimento dos preços pode ser dividido em três tendências: primária, secundária e terciária. A tendência primária pode ser de alta ou baixa, e durar até dois anos. A tendência secundária pode durar alguns meses. A tendência terciária dura em média algumas semanas. A Figura A.1 apresenta um exemplo das três tendências.

Figura A.1 Gráfica ilustrativa dos três tipos de tendência.

III. ***Princípio da confirmação***: a confirmação da tendência de alta ou baixa é dada pela repetição do movimento de preço em outros índices, i.e., um índice confirma o outro. A Figura A.2 ilustra o terceiro princípio de Dow, onde duas ações apresentam comportamentos semelhantes.

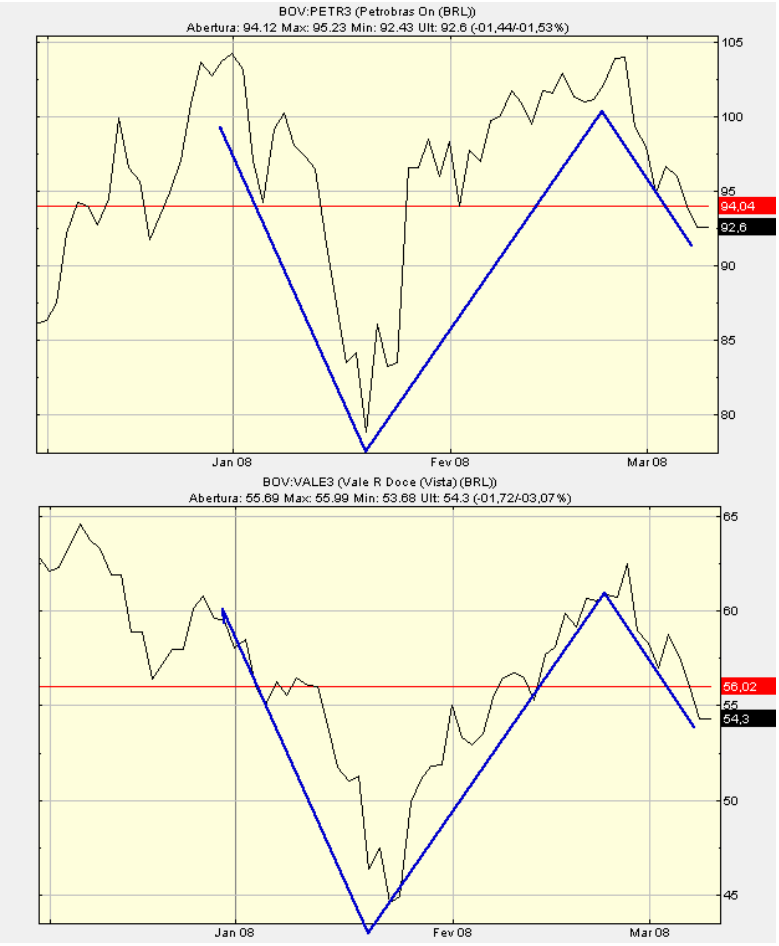

Figura A.2 Representação de comportamentos semelhantes.

IV. ***Volume converge***: a tendência é consistente se existe a participação de um número crescente de investidores para dar continuidade a sua trajetória. Portanto, o volume confirma a tendência, i.e., se a tendência é de alta o volume aumenta, se tendência é de baixa o volume diminui. A Figura A.3 apresenta um exemplo entre a relação do volume e o preço de um ativo.

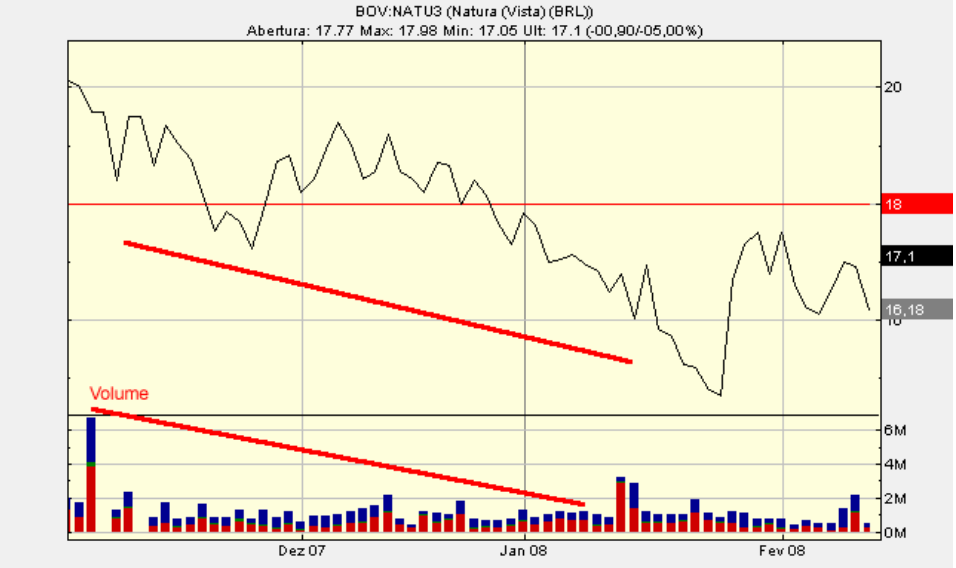

Figura A.3 Relação entre os índices de volume e preço de um ativo.

V. **Sinais evidentes de reversão**: **a tendência acaba quando surgem sinais evidentes de reversão**: Apesar desta técnica ser intuitiva, o quinto princípio de Dow é fundamental. A questão é saber até quando uma tendência vai subir ou descer. Para isso é necessário identificar falhas no movimento dos preços. A Figura A.4 apresenta uma falha no movimento de subida indicada pelo círculo, este ponto comprova o fim da tendência.

Figura A.4 Indicação da final de tendência de alta.

A.1.4 Indicadores

A análise de gráficos, antigamente realizada pela observação de desenhos elaborados em papel, modernizou-se a tal ponto que hoje, esta atividade pode ser automatizada por meio de algoritmos capazes de processar numericamente os gráficos e identificar as tendências do mercado. O processo de descoberta das tendências é feito por meio da avaliação dos indicadores específicos para cada técnica. Esta seção visa examinar os principais indicadores da análise técnica que serão utilizados na construção do trabalho proposto.

A.1.4.1 Média Móvel

A média móvel é o indicador baseado na média do preço que se desloca ao longo do tempo (Matsura, 2006). Esta técnica permite realizar o "alisamento" dos dados ruidosos no gráfico do preço o que facilita a detecção da tendência. A Figura A.5 ilustra o comportamento da uma média móvel.

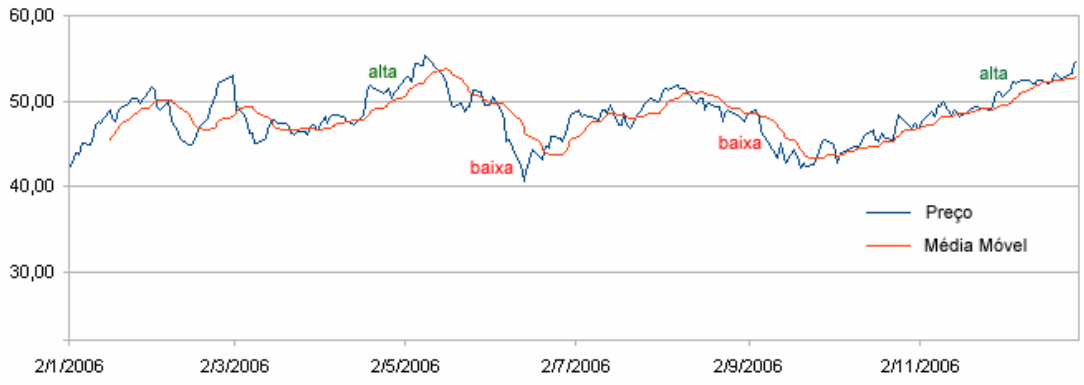

Figura A.5 Representação da média móvel para um ativo.

A partir da média móvel é possível determinar as tendências de alta e baixa de uma ação. Quando o preço encontra-se acima da média móvel, indica a tendência de alta, do contrário, quando o preço encontra-se abaixo da média móvel indica a tendência de baixa. Além da detecção da tendência, ainda é possível prever os momentos de inversão por meio da análise dos cruzamentos dos gráficos que será examinado nas seções abaixo.

A primeira maneira para detectar os momentos de inversão e conseqüentemente de compra e venda de uma ação a partir da média móvel está na observação do cruzamento entre o gráfico do preço e o gráfico da média móvel.

Quando o gráfico do preço cruza o gráfico de média móvel de cima para baixa representar um sinal de baixa, portanto recomenda-se vender. Quando o gráfico de preço cruza a média móvel de baixo para cima indica tendência de alta, logo se recomenda comprar.

```
Leia cotacao_anterior, cotacao_atual, media _anterior, media _atual

Se(cotacao_anterior >= media_anterior)e(cotacao_atual < media_atual)entao
    Indique Venda
Se(cotacao_anterior <= media_anterior)e(cotacao_atual > media_atual)entao
    Indique Compra
Senão
    Indique Mantenha
Fim se
```

Algoritmo 1. Cruzamento Média Preço

A técnica, apesar de eficiente para a maioria das ações, pode não obter bons resultados caso haja muitos ruídos no gráfico do preço. Os ruídos podem ocasionar a detecção de falsas tendências como ilustrado na **Erro! Fonte de referência não encontrada.** pelo falso ponto de compra.

Para minimizar os ruídos ocasionados pelo cruzamento da média móvel ao preço, é possível utilizar a técnica de cruzamento de duas médias móveis com períodos diferentes. A

Figura A.6 ilustra para uma ação a média móvel de cinco dias conjuntamente com a média móvel de cinqüenta dias.

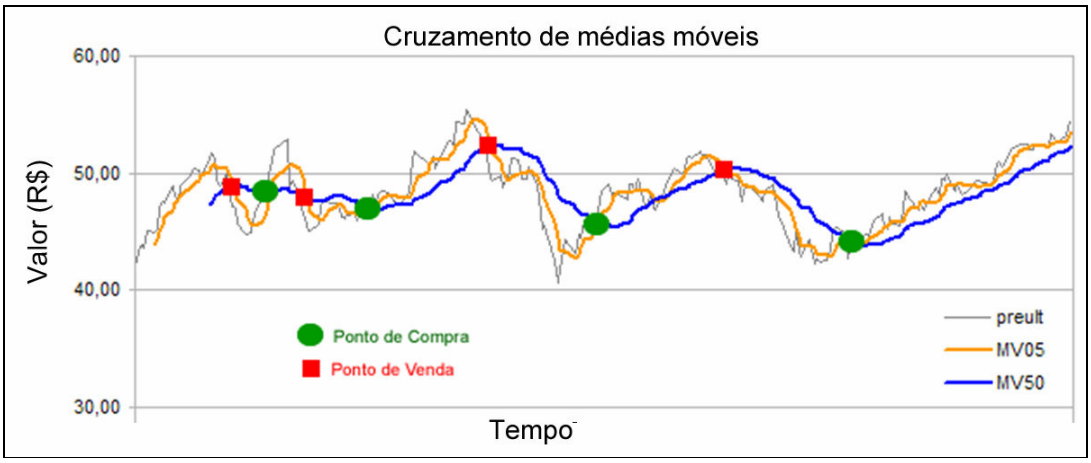

Figura A.6 Relação entre médias móveis com intervalo de tempo distinto.

Ao partir da premissa que as médias móveis diminuem ruídos em relação ao gráfico de preços, concluí-se que o cruzamento das médias móveis possui menos cruzamentos falsos. O Algoritmo 1. Cruzamento Média Preço, pode ter aplicação sem grandes alterações para o cruzamento entre médias móveis. O seguinte Algoritmo 2 implementa esses ajustes:

```
Leia media_menor_anterior, media_menor_atual
Leia media_maior_anterior, media _atual
Se (media_menor_anterior >= media_maior_anterior) e
   (media_menor_atual < media_maior_atual)entao
    Indique Venda
Se (media_menor_anterior <= media_maior_anterior) e
   (media_menor_atual > media_maior_atual) entao
    Indique Compra
Senão
    Indique Mantenha
Fim se
```

Algoritmo 2. Cruzamento de Médias

A.1.4.2 Estocástico

O índice estocástico detecta os momentos de compra e venda das ações a partir da relação entre o preço de fechamento do papel e os valores máximos e mínimos do preço para um determinado período de tempo. Este técnica cria regiões de "*sobrecompra*" e "*sobrevenda*".

A fórmula do método estocástico é definida como:

$$\%N = Número\_de\_dias$$

$$\%K = \frac{\left(fechamento\_atual - menor\_\min imo(\%N)\right)}{\left(maior\_máximo(\%N) - menor\_\min imo(\%N)\right)} * 100$$

$$\%D = MediaMovel(\%K, \%N)$$

Quando o fechamento do dia se aproxima do maior máximo, o %K se aproxima de 100 e quando o fechamento se aproxima do menor mínimo o %K se aproxima de 0. A variável %D é uma média móvel do %K em um número %N de dias.

```
Leia %K_anterior, %K_atual
Leia %D_anterior, %D_atual
Se (%K_anterior >= %D_anterior) e (%K_atual < %D_atual)entao
    Indique Venda
Se (%K_anterior <= %D_anterior) e (%K_atual > %D_atual)entao
    Indique Compra
Senão
    Indique Mantenha
Fim se
```

Algoritmo 3. Estocástico

A técnica afirma que se deve comprar quando o %K cruza o %D de baixo para cima e deve-se vender quando o %K cruza o %D de cima para baixo.

A.1.5 Considerações Finais

As duas principais escolas da análise de ações, fundamentalista e técnica, possuem vantagens e desvantagens. A análise técnica é mais flexível para uso em larga escala, pois qualquer mercado pode ser analisado sem a necessidade do profundo conhecimento das organizações. Este princípio vai à contramão da analista fundamentalista que depende do íntimo conhecimento do mercado. Outra vantagem é a maior eficiência da análise técnica para investimentos de curso prazo como no caso das operações *intraday*.

A metodologia fundamentalista examina a causa do movimento dos preços, enquanto a técnica examina o seu efeito, o que faz bastante diferença, pois a rentabilidade é maximizada quando é possível prever o movimento e não apenas segui-lo. Para os fundamentalistas, o investimento é realizado com menos riscos, pois se adquire um ativo tendo o discernimento que se está pagando um valor inferior ao preço justo. É possível

estimar o retorno esperado e monitorar as variáveis importantes que afetam exclusivamente estes investimentos. É o método mais indicado para investimentos em longo prazo.

O fato é que na prática, a grande maioria dos investidores utiliza a análise técnica juntamente com a fundamentalista para construir suas estratégias de investimento. Isto se dá porque apesar das discussões, entre os participantes do mercado de ações, sobre qual a melhor técnica, existe o consenso que uma não exclui a outra, muito pelo contrário, estudos mostram que o investimento pode ser otimizado com o auxílio de ambas as análises.

Apêndice B

Modelagem do CRONOS

Este apêndice apresenta a modelagem utilizada para a concepção e construção da ferramenta CRONOS, discutida no Capítulo 4. Sua organização estrutura-se em quatro seções: a primeira seção apresenta a visão de implantação da ferramenta sobre uma perspectiva distribuída, a segunda seção apresenta os módulos do CRONOS por meio de diagramas de pacotes; a terceira seção apresenta detalhes de construção por meio de diagramas de classes e trechos de códigos e por fim a quarta seção apresenta as dependências de bibliotecas necessárias para a execução da ferramenta.

B.1 Visão de Implantação

O CRONOS é baseado sob a arquitetura multiagente proposta pelo Framework Jade (Jade, 2009), portanto sua representação de implantação assume uma disposição distribuída entre os agentes. A Figura B.1 ilustra uma configuração de implantação do CRONOS sobre uma rede com três máquinas:

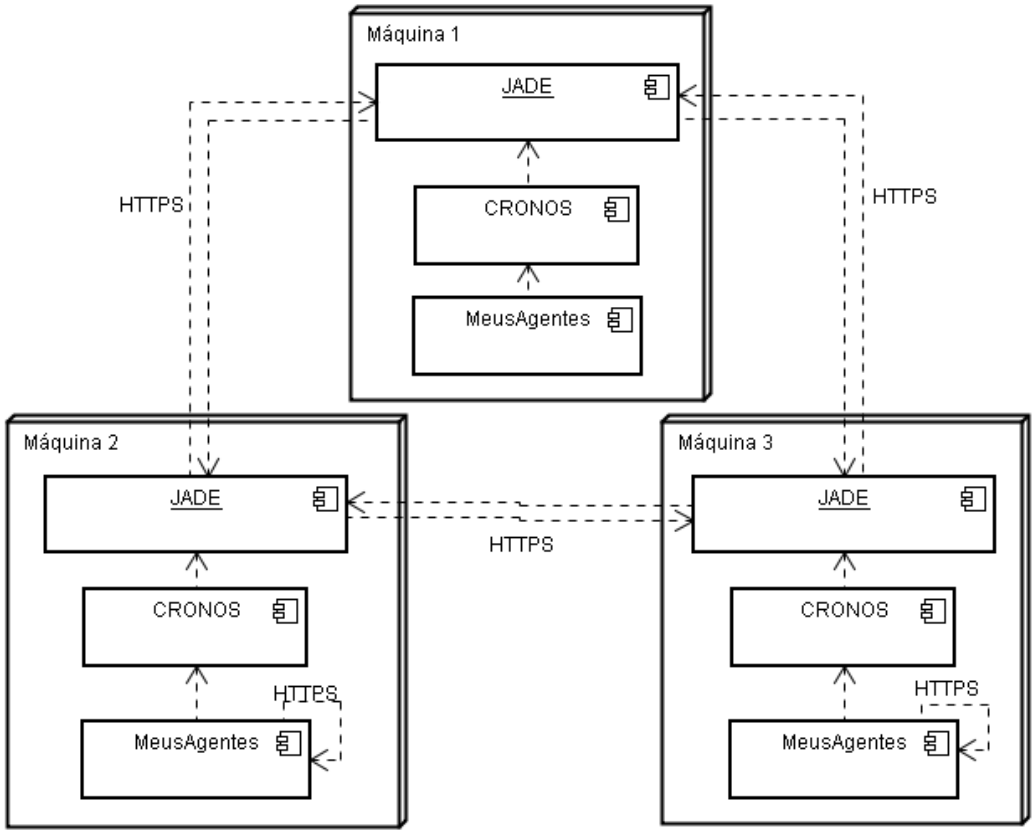

Figura B.1 Diagrama de Implantação

A comunicação entre os agentes é definida pelo protocolo HTTPS, necessário para criptografia e assinatura das mensagens. Os agentes dependem da biblioteca CRONOS que utiliza a infra-estrutura do JADE para envio e recebimento de mensagens.

B.2 Visão de Módulo

Internamente o CRONOS é estruturado em vários módulos a fim de facilitar a construção, manutenção e evolução da ferramenta. Cada módulo é representado por um pacote, no qual possui responsabilidades distintas ou sempre que possível única no sistema. A Figura B.2 ilustra os módulos do CRONOS por meio do digrama de pacotes:

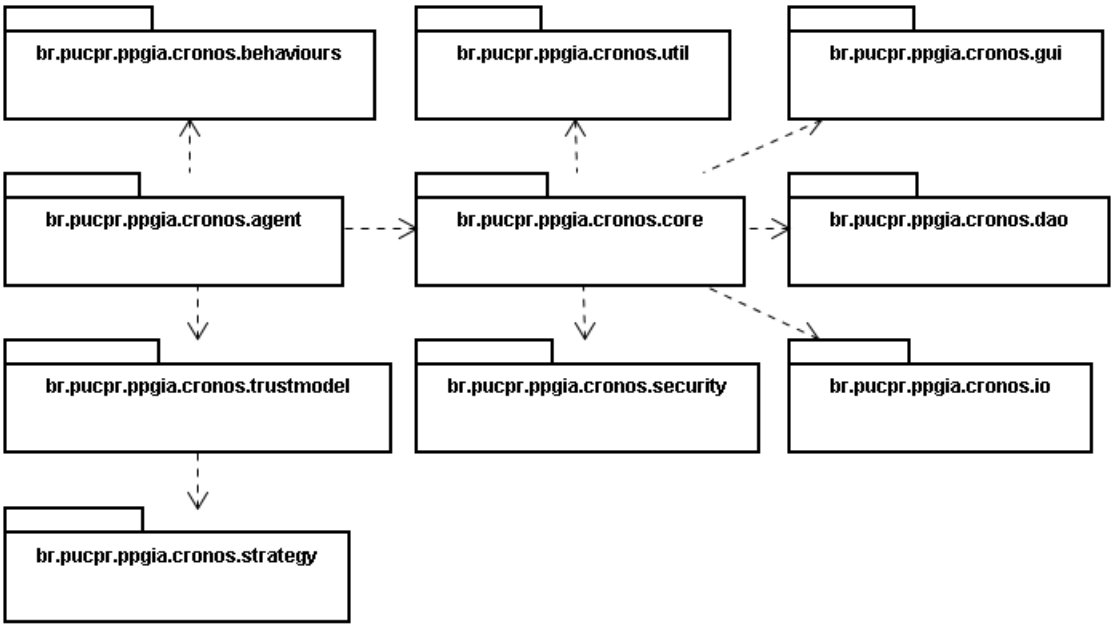

Figura B.2 Diagrama de Pacotes

Segue abaixo a descrição de cada módulo:

- *Agent*: possui as classes que representam os agentes investidores, especialistas, certificadores e operadores do simulador do CRONOS. Caso sejam construídos outros casos de estudos, além da bolsa de valores, este pacote também possuirá os novos agentes.
- *Behaviours*: possui as classes de comportamento para todos os agentes tais como: solicitar avaliação, enviar avaliação, solicitar serviço e prover serviço.
- *Trustmodel*: possui as construções dos modelos de confiança que poderão ser utilizados pela ferramenta. Conforme o Capitulo 4, há quatro representações de modelos: ausência de confiança, confiança direta, confiança indireta e confiança certificada. Novos modelos poderão ser construídos sob este pacote.
- strategy: possui as estratégias para fornecimento de serviços. Utilizado pelos agentes provedores. Para o simulador da bolsa de valores este pacote contém diferentes estratégias de investimento como as análises técnicas baseadas em média móvel, estocástico e candlestick.
- *GUI*: possui a interface gráfica do painel de controle. Baseada em *Swing* para interação com o usuário final e com forte dependência do componente

Jfreechart para construção dos gráficos de linha utilizados durante a exibição dos resultados;

- *Core*: possui a construção do ambiente multiagente. O simulador da bolsa de valores, por exemplo, é construído neste módulo;
- *DAO*: define a camada de acesso a dados utilizado pelo simulador. Garante uma abstração de persistência e integração com bancos de dados que possuam uma construção *Java Database Connectivity* (JDBC);
- *IO*: pacote utilitário para leitura e escrita de arquivos. Possui como mecanismo adicional a importação das bases de dados históricas da Bovespa realizando a conversão dos registros textos para coleções de objetos;
- *Security*: possui construções para criptografia e assinatura digital necessárias durante as trocas de mensagens entre os agentes.
- *Util*: possui um conjunto de funções utilitárias com leitura e escrita de arquivos, acesso a classes de configuração, classes de conversão de dados, entre outras;

B.3 Visão de Construção

A seção anterior apresentou uma visão geral dos módulos que estruturam a ferramenta CRONOS. Esta seção descreve detalhes de construção dos módulos apresentando suas relações e dependências entre si. Além disso, é apresentada a construção dos principais mecanismos do sistema como: o ambiente, as extensões dos agentes, as extensões dos modelos de confiança entre outros.

O módulo *Core* é composto pelo ambiente, classe *Environment*, que possui controle sobre todas as variáveis do simulador, conforme ilustrado pela Figura B.3. O controle do tempo é representado pela classe *Phase* que defini os momentos de abertura e fechamento de cada pregão. Os agentes certificadores, classe *Certificator*, são os agentes intrínsecos ao CRONOS, que possuem a responsabilidade de garantir a criptografia das mensagens trafegadas sob o ambiente.

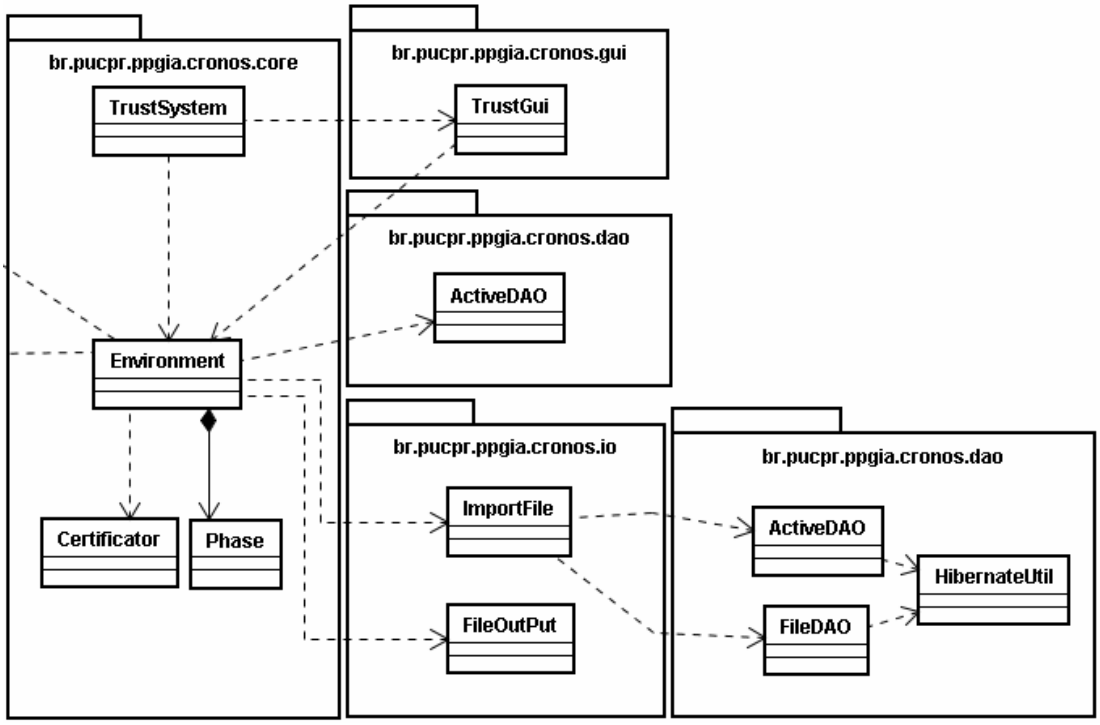

Figura B.3 Módulo Core e suas dependências

A classe *TrustSystem* representa a fachada do sistema multiagente. É a partir dela que sistema é iniciado. Seu método *main* deve aciona a interface gráfica, classe *TrustGui*, e o ambiente. Segue trecho da classe:

```
public class TrustSystem {
	public static void main(String[] args) {
		Environment envi = Environment.getInstance();
		TrustGui gui = new TrustGui();
		envi.setGUI(gui);
		gui.setEnvironment(envi);
		gui.setVisible(true);
	}
}
```

Conforme ilustrado na Figura B.3, o ambiente multiagente controla os objetos de acesso a dados, classe *ActiveDAO*, e os utilitários de leitura e escrita, *ImportFile* e *FileOutPut*. Inicialmente, quando o sistema não possui uma base dos dados populada, o objeto da classe *ImportFile* verifica a necessidade de importação no arquivo de configuração a*pplication.properties*. Este arquivo define quais arquivos serão importados, além de definir qual cenário será utilizado para o experimento. Segue exemplo do *application.properties* que define a importação das bases histórica de 2005 e 2006 para o cenário 2:

```
data.files=dados/COTAHIST_A2005.zip,dados/COTAHIST_A2006.zip
file.config=cenario_002.properties
```

Um cenário representa uma possível configuração do sistema na qual pode estabelecer: quantidade de agentes no ambiente, modelos de confiança utilizados, ativos a serem avaliados, intervalo de tempo para coleta as informações, entre outras variáveis. O cenário também é definido por um arquivo de configuração que é lido pela camada de *IO* do sistema. Segue exemplo do cenário 2 utilizado nos experimentos do Capítulo 4 que apresentaram os resultados da Tabela 4.9:

```
     #Arquivo de configuração do cenário 2 (cenário_002.properties)
     #Informa se os dados serão coletadas para todas as fase. Padrão=false
     show.allPhase=true

     #Informa se o comportamento dos servidor mudará conforme o tempo
     system.serverChange=false

     #Periodicidade das mudança
     server.change.days=10

     #Uma resposta do servidor vale entre
     response.valid.min=5
     rating.valid.max=30

     #Valor inicial de cada carteira
     agent.count.initvalue=100000

     #Configura quando as informações são coletadas (período de dias)
     tempo.coleta=15

     #Configura o tipo de coleta, client (1) (coleta as informações dos cliente),
server (2) (dos servidores, all (3) (ambos)
     colect.model=1

     #informa o número de grupos do sistema
     agent.group.size=1

     #Ativos a serem tratados
     data.actives=VALE5,CSNA3

     #Agentes Clientes
     create.clients=FixedTrust:1,NoTrust:200,DirectTrust:200,IndirectTrust:200,Indi
rectCrypTrust:200

     #Agentes Servidores
     create.servers=PerfectStrategy:200,GoodStrategy:300,BadStrategy:400,TerribleSt
rategy:500,MaliciousStrategy:250

     #Janela de tempo do método estocático
     estocatico.days=30
     #Valor limite para compra
     estocatico.max=0.95
     #Valor limite para venda
     estocatico.min=0.05
     #Janela de tempo do método Volume
     volume.days=10
```

Depois de lido os arquivos de configuração, mencionados anteriormente, o ambiente do sistema multiagente deve acionar o agentes para realizar suas tarefas diárias. Para cada dia

do intervalo de tempo estabelecido, o ambiente aciona seus agentes conforme descrito no código a seguir:

```
public void runAllDays(String cenario){
        time = getTimes();
        for (Date day : time) {
               phase.setPhase(day, time);
               setToday(day);
               itoday++;
               gui.updateDate(day, itoday);
               for (AgentGroup group : listAgentGroup) {
                     group.runDay(day);
               }
        }
 }
```

A partir daqui, o controle da execução é distribuído pelos agentes do CRONOS. Conforme ilustrado na Figura B.4, os agentes clientes e servidores estendem a classe abstrata *CronosAgent*.

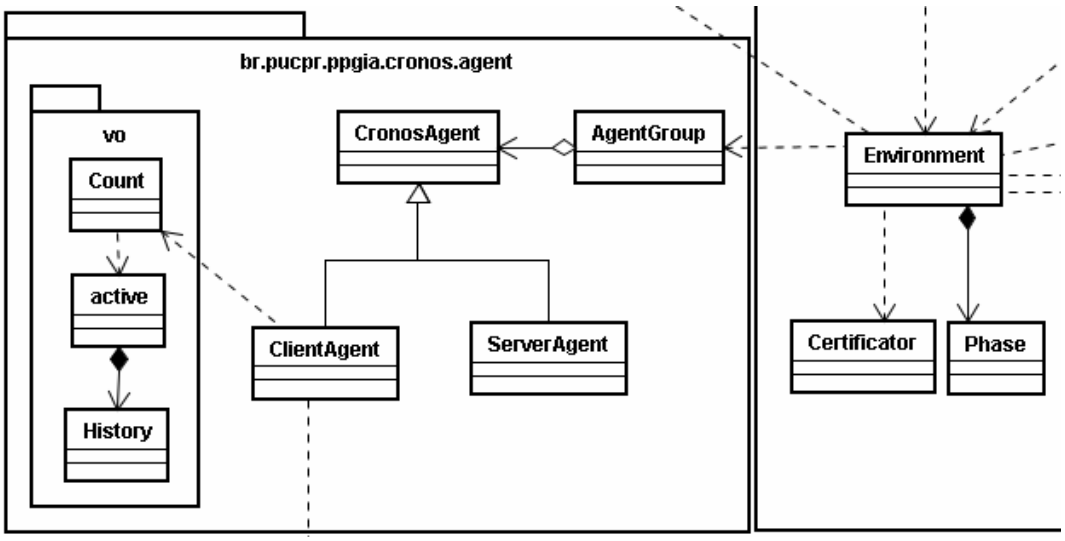

Figura B.4 Agentes Cronos.

Esta classe define dois métodos abstratos que devem ser implementados pelas classes filhas. O método *runDay* representa a ação do agente para um determinado dia e o método *getService* deve retornar uma ou mais respostas, classe *Response*, para um pedido de serviço. Segue código:

```
public abstract class CronosAgent {
        public abstract void runDay(Date today);
        public abstract List<Response> getService();
}
```

Além dos métodos abstratos, a classe *CronosAgent* prove código comum que será utilizado pelos agentes clientes e servidores. Um desses códigos é o cálculo de utilidade de um serviço provido por um servidor. Segue trecho de código:

```
public abstract class CronosAgent {
  public Double getUtility(CronosAgent server, String term){
        int count_rating = 0;
        Double utility_value = 0d;
        List<Rating> listAll = ratings.get(getKey(server, term));
        List<Rating> listToRemove = new ArrayList<Rating>();
        if (listAll != null){
              for (Rating rating : listAll) {
                    if (rating.isExpired()){
                          listToRemove.add(rating);
                    }else{
                          count_rating++;
                          utility_value += rating.getValor();
                    }
              }
        }
        utility_value = (count_rating > 0?utility_value/count_rating : null);
        Util.remove(listAll, listToRemove);
        return utility_value;
  }
}
```

O método *getUtility* analisa todas as avaliações do servidor referente a um determinado termo e realiza uma média ponderada de todas as avaliações para calcular o valor de utilidade de um servidor.

A classe *ServerAgent* implementa o método abstrato *runDay* da classe pai, buscando recomendações de compra e venda para o conjunto de ações que ele possui competência. Desta forma, quando um agente cliente solicita alguma indicação, o servidor apenas repassa as informações processadas no início do dia:

```
public class ServerAgent extends CronosAgent {
  public void runDay(Date today) {
        Date today = Environment.getInstance().getToday();
        for (Acao acao : Environment.getInstance().getAcoes()) {
              idResponse++;
              Action action = strategy.getRecomentation(acao);
              response = new Response(idResponse, acao, action, today);
              responseService.add(response);
        }
}
```

A classe *ClintAgent* implementa o método abstrato *runDay* da classe pai, delegando sua execução ao modelo de confiança. Portanto é o modelo de confiança que age no lugar do agente.

```
public class ClientAgent extends CronosAgent {
        public void runDay(Date today) {
              trust.runDay(today);
        }
}
```

As ações dos agentes clientes são controladas pelos modelos de confiança. A Figura B.5 apresenta a hierarquia dos modelos de confiança utilizados neste trabalho:

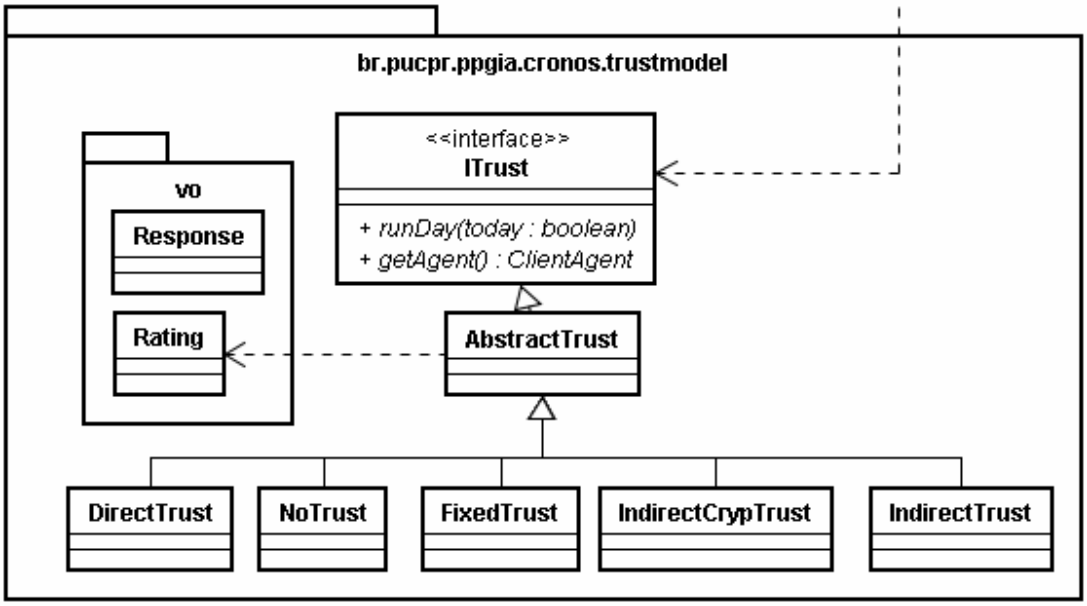

Figura B.5 Integração dos agentes com o ambiente.

No topo da hierarquia encontra-se a interface *ITrust* que define dois métodos: *runDay* e *getAgent*. O primeiro método define o comportamento que o agente deverá assumir para um determinado dia e o segundo método retorna uma referência ao agente controlado pelo modelo.

```
public interface ITrust {
        public void runDay(Date today);
        public ClientAgent getAgent();
}
```

Para exemplificar a utilização desta interface, o código a seguir apresenta a implementação do modelo de confiança direto:

```
public interface ITrust {
  public void runDay(Date today){
  double buy = 0d, sell = 0d, nothing = 0d;
  for (Acao acao : myAgent.getAcoes()) {
    List<Response> responses = responseList.get(acao);
     for (Response resp : responses) {
      double ratingV = myAgent.getUtility(resp.getFromAgent(), resp.getTerm());
         switch (resp.getValue()) {
                case BUY:
                       buy += ratingV;
                       break;
                case SELL:
                       sell += ratingV;
                       break;
                case NOTHING:
                       nothing += ratingV;
                       break;
                default:
                       break;
         }
         if (buy > sell){
                myAgent.getContas().get(acao).buy();
         }
         if (sell > buy){
                myAgent.getContas().get(acao).sell();
         }
  }
}
```

No modelo direto o agente cliente calcula a utilidade de um agente servidor para um determinado tipo de serviço e o modelo de confiança decide se ele deverá comprar ou vender certo tipo de ativo.

B.4 Visão de Dependência

Esta seção apresenta o conjunto de dependências necessárias à execução do CRONOS. As dependências diretas são bibliotecas que o CRONOS faz referência direta e utiliza suas classes como no caso do Jade, Hibernate, JFreeChart, Junit, Log4j e Hsqldb. As demais bibliotecas são denominadas de dependências indiretas, pois apesar do CRONOS não fazer referência a elas as dependências direta fazem. A Figura B.6 apresenta a árvore de dependências diretas e indiretas do sistema:

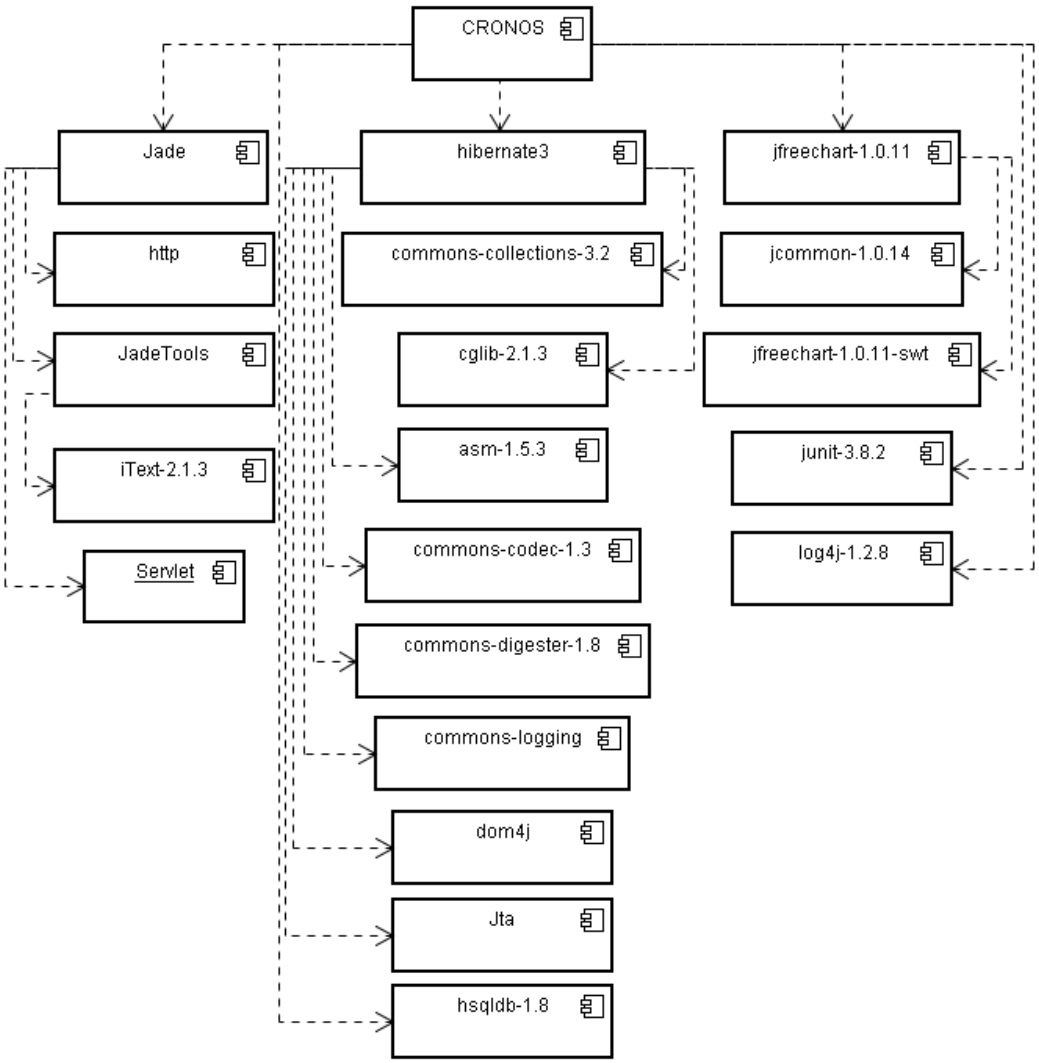

Figura B.6 Diagrama de Dependências

A Tabela B.1 discrimina cada dependência apresentando descrição e uma referência ao site do fabricante.

Biblioteca	Descrição	Site
cglib-2.1.3	Gerador de código usado para estender as classes Java.	http://cglib.sourceforge.net/

asm-1.5.3	Manipulador de bytecode. Pode dinamicamente modificar classes já existentes ou gerar dinamicamente classes, diretamente na forma binária.	http://asm.ow2.org/
commons-codec-1.3	Fornece implementações de codificadores e decodificadores comum como Base64, Hex, fonética e URLs.	http://commons.apache.org/codec/
commons-collections-3.2	Fornece estruturas de dados adicionais a linguagem Java	http://commons.apache.org/collections/
commons-digester-1.8	Prover uma api para configurar e ler arquivos xml.	http://commons.apache.org/digester/
commons-logging-1.1	Biblioteca para geração de logs durante execução	http://commons.apache.org/logging/
dom4j-1.6.1	Biblioteca para manipulação de arquivos xml.	http://www.dom4j.org/
hibernate3	Biblioteca de mapeamento objeto/relacional para persistências de objetos	https://www.hibernate.org/
Http	Biblioteca que implementa o protocolo http necessário a comunicação dos agentes	http://hc.apache.org/httpclient-3.x/
Iiop	Biblioteca para chamadas remotas (Java RMI)	http://www.j2ee.me/products/rmi-iiop/index.html
iText-2.1.3	Biblioteca para geração de arquivos PDF	http://www.lowagie.com/iText/
Jade	Framework para desenvolvimento de agentes sob a plataforma Java	http://jade.tilab.com/
JadeTools	GUI do Jade	http://jade.tilab.com/
Jfreechart-1.0.11	Biblioteca para construção de gráficos	http://www.jfree.org/jfreechart/

Jfreechart-1.0.11-swt	Biblioteca para construção de GUI em swing	http://www.jfree.org/jfreechart/
jcommon-1.0.14	Biblioteca Utilitária do Jfreechart	http://www.jfree.org/jfreechart/
Jta	Especificação Java para transações distribuídas	http://java.sun.com/javaee/technologies/jta/
Junit-3.8.2	Framework para construção de testes de unidade	http://www.junit.org/
log4j-1.2.8	Biblioteca para geração de logs	http://logging.apache.org/log4j/1.2/index.html
Servlet	Api para comunicação web da plataforma Java Entreprise Edition	http://java.sun.com/products/servlet/
hsqldb-1.8	Driver de conexão JDB ao SGBD HSQLDB	http://hsqldb.org/

Tabela B.1 Lista de Dependências