Gerenciamento de Confiança em Ambientes Pervasivos

Olympio Cipriano da Silva Filho

Dissertação de Mestrado submetida à Coordenadoria do Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande - Campus de Campina Grande como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências no Domínio da Engenharia Elétrica.

Área de Concentração: Processamento da Informação

Angelo Perkusich, Dr.
Orientador

Campina Grande, Paraíba, Brasil

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA CENTRAL DA UFCG

S586g
2007 Silva Filho, Olympio Cipriano da.
Gerenciamento de confiança em ambientes pervasivos / Olympio Cipriano da Silva Filho. — Campina Grande, 2007.
69f. : il

Dissertação (Mestrado em Engenharia Elétrica) – Universidade Federal de Campina Grande, Centro de Engenharia Elétrica e Informática.
Referências.
Orientador: Dr. Angelo Perkusich.

1. Modelo de Confiança. 2. Pervasivo. 3. Middleware. I. Título.

CDU 004:519.711 (043)

GERENCIAMENTO DE CONFIANÇA EM AMBIENTES PERVASIVOS

OLYMPIO CIPRIANO DA SILVA FILHO

Dissertação Aprovada em 14.12.2007

ANGELO PERKUSICH, D.Sc., UFCG
Orientador

ANTONIO MARCUS NOGUEIRA LIMA, Dr., UFCG
Componente da Banca

MARCO AURÉLIO SPOHN, Dr., UFCG
Componente da Banca

CAMPINA GRANDE - PB
DEZEMBRO - 2007

Resumo

Com o desenvolvimento tecnológico a computação está cada vez mais integrada ao cotidiano das pessoas. Esta integração faz parte de um novo paradigma da computação: o da *Computação Pervasiva*. Neste paradigma, a computação está sempre a disposição do usuário, em qualquer lugar, a qualquer momento, através de ambientes repletos de dispositivos computacionais embutidos em uma grande variedade de objetos do dia-a-dia das pessoas como, automóveis, telefones, etc.

Nestes ambientes, os dispositivos interagem entre si através de serviços devendo ser capazes lidar com a grande variedade de protocolos de descoberta de serviços disponíveis. Ainda, essas interações entre os dispositivos podem ocorrer em ambientes sem infra-estrutura de redes de computadores e entre dispositivos que podem agir desonestamente.

Neste trabalho, é desenvolvido um modelo para avaliar a confiança nos serviços nos ambientes pervasivos a partir de experiências e recomendações. Este modelo é implementado por um Gerenciador de Informação de Confiança o qual disponibiliza um mecanismo de tomada de decisão para realizar a seleção de clientes e provedores de serviço. Ainda, o Gerenciador de Informação de Confiança é integrado a um *middleware* para computação pervasiva, o *Wings*, possibilitando a descoberta de serviços confiáveis.

Foram utilizadas simulações e uma aplicação de teste para avaliar o Gerenciador de Informação de Confiança e sua integração ao *middleware Wings*. A partir dos resultados obtidos observou-se que a solução apresentada se mostrou satisfatória para a utilização em ambientes pervasivos.

Abstract

The development of technology is integrating computing to people every day life. This integration belongs to a new computing paradigm: the pervasive computing. In this paradigm, the computation is always available, in every place at any time, through environments crowd of computing devices embedded in a great variety of objects like cars, cell phones, etc.

In these environments, devices interact through services and should be able to deal with different service announcement solutions available. These interactions can occur without computing network infrastructure and among malicious devices.

In this work, it is developed a model to evaluate service trust in pervasive environments from experiences and recommendations. This model is implemented by a Trust Information Manager that provides a mechanism to make trust based decision. This Trust Information Manager is integrated to a pervasive computing middleware, called *Wings*, enabling trusted service discover.

Simulations and a test application were used to evaluate the Trust Information Manager and its integration with the Wings middleware. From the results it could be seen that the presented solution is efficient and can be widely used in pervasive computing environments.

Agradecimentos

Agradeço a Deus.

Agradeço a minha mãe, Maria de Lourdes, e meus irmão, Airon e Ana, pelo apoio que me deram sempre, em todas os momentos de minha vida. Agradeço ainda a Erika (bb) pelo seu esforço em me ajudar nos períodos mais difíceis desta caminhada.

Agradeço aos professores Angelo, Hyggo, Leandro, Marcos Morais, Antonio Marcus, pelas conversas, conselhos, idéias, correções, enfim, todo auxílio que deram para a realização deste trabalho.

Agradeço também aos amigos do laboratório que me ajudaram. Agradeço em especial a Danilo (tiozinho), Zé Luiz (Jack Bauer), Yuri Gomes (kblim), Adrian Lívio (arão), Diego (tuk), André Felipe (timbu), Mario Hozano (turista), Marcos Fábio (mãozinha), Paulo (gordo) com os quais tive o prazer de conviver nestes anos de trabalho.

Por fim, agradeço a todas as pessoas que sabem que seu nome deveria ser citado, mas que não consegui lembrar.

Índice

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

Os computadores pessoais foram, por muito tempo, os equipamentos de referência para o usuário comum quando se falava no termo "computação". A razão disso não advém somente da semelhança dos nomes, mas também pelo fato de que os computadores pessoais marcaram o início do processo de *descentralização* da computação, que se seguiu à era dos *mainframes*, popularizando os computadores pessoais em casas e escritórios [12]. Com isso, a computação passou a fazer parte da vida das pessoas estando presente em várias áreas como a indústria, telecomunicações, comércio, entretenimento, etc. Este processo de descentralização continua ainda hoje, observando-se como resultado, a popularização de dispositivos de caráter ainda mais pessoal como *smart phones* e *handhelds*. Tais dispositivos portáteis permitem que a computação hoje em dia seja vista de forma ainda mais integrada ao cotidiano das pessoas podendo estar presente em qualquer lugar a qualquer momento.

Este processo de descentralização e integração da computação se encaixa em um paradigma definido em 1991 por Mark Weiser [32]: o da *Computação Pervasiva*. Nele, a computação estaria embutida em objetos comuns do dia-a-dia das pessoas (i.e. celulares, roupas, automóveis, mp3 *players* , etc.), todos se comunicando através de redes inter-conectadas com o objetivo de realizar tarefas em nome do usuário, compartilhando serviços e informações em *qualquer lugar* e a *qualquer momento*. Ainda, a execução de tais tarefas deveria ser feita de modo a reduzir ao máximo a intervenção do usuário, com as interações ocorrendo entre os dispositivos e não mais exclusivamente entre as pessoas, exigindo que as aplicações ajam de maneira pró-ativa [25]. Isso possibilitaria que a computação desaparecesse da ciência do usuário, alcançando o que foi chamado por Weiser, de *invisibilidade* da computação. A aglomeração de dispositivos com estas características formaria ambientes saturados com capacidade de comunicação e computação [25], chamados ambientes pervasivos.

Segundo o paradigma de Weiser, em ambientes pervasivos, cada dispositivo é um cliente ou provedor de serviços em potencial. Isso faz com que estes ambientes possam ser utilizados como repositórios de serviços a serem compartilhados entre os dispositivos presentes no

ambiente.

Uma característica que afeta diretamente este compartilhamento de serviços é a *heterogeneidade* entre os dispositivos presentes nos ambientes pervasivos. Normalmente estão presentes dispositivos com diferentes capacidades de processamento e armazenagem. Além disso, eles podem se comunicar de maneiras diferentes utilizando diferentes protocolos de descoberta de serviços como UPnP[1], Bluetooth SDP[2], etc. Para lidar com essa grande diversidade utilizam-se *middlewares* para a descoberta e disponibilização de serviços nos ambientes pervasivos [37].

A descoberta/disponibilização e utilização de serviços realizada através dos dispositivos portáteis nos ambientes pervasivos pode ocorrer em virtualmente qualquer lugar, seja em um local com infra-estrutura através de um ponto de acesso, seja em local sem infra-estrutura através de uma rede ad hoc ou até mesmo diretamente entre dispositivos (ponto-a-ponto). Os dispositivos podem ser movidos para diferentes redes entrando em contato com outros dispositivos que podem ser conhecidos ou não. Essa *dinamicidade* é uma característica fundamental da computação pervasiva. Dentre as vantagens desta dinamicidade estão a disponibilização de serviços de forma pervasiva (i.e. a qualquer lugar, a qualquer momento) e a colaboração entre dispositivos através do compartilhamento de tais serviços. No entanto, deve-se observar que em muitos casos não existe um administrador do sistema e os dispositivos podem estar isolados, fazendo uso apenas de comunicação de curto alcance.

Por exemplo, suponha que um grupo de pessoas esteja em um restaurante, cada uma com um *smart phone* executando uma aplicação pervasiva para compartilhamento de arquivos. A aplicação, por ser pervasiva, executa a maior parte do tempo sem a necessidade de interferência do usuário, em busca de arquivos de seu interesse. No restaurante, a cooperação se inicia (i.e. compartilhamento de arquivos) quando outros dispositivos que anunciam o serviço procurado (serviço de compartilhamento de arquivos). Com a chegada de mais pessoas com seus respectivos dispositivos, a tendência é que a quantidade de serviços disponíveis aumente, elevando assim o grau de cooperação entre os dispositivos, aumentando a quantidade de informações compartilhadas. No entanto, não necessariamente, os dispositivos agem corretamente uns com os outros. Alguns deles podem simplesmente fornecer arquivos corrompidos enquanto outros podem fazer requisições excessivas, com o objetivo de alcançar resultados mais rapidamente, não respeitando as limitações dos dispositivos, exaurindo assim seus recursos. Nestes casos, onde não há a presença de um administrador que discipline as interações e os dispositivos agem de forma independente, faz-se necessário que eles tenham a capacidade de determinar com quais dispositivos devem se relacionar, evitando a utilização indevida de seus serviços bem como possibilitando a escolha adequada de provedores dos mesmos.

[1] *http://upnpforum.org*

[2] *http://www.bluetooth.com*

1.1 Problemática

Como pode ser observado a partir do cenário apresentado anteriormente, é importante que os dispositivos tenham uma maneira de decidir com quais outros dispositivos podem interagir. Pode-se adotar as soluções tradicionais de segurança, como por exemplo a utilização de Infra-estrutura de Chave Pública [10]. Tal infra-estrutura possibilita que dispositivos sem contato anterior possam se autenticar mutuamente. No entanto, esta solução não é adequada para o cenário apresentado, pois existe a possibilidade das interações ocorrerem em locais sem infra-estrutura (por exemplo, sem conexão com a Internet) o que impediria a verificação da validade dos certificados através das listas de revogação. Outro problema associado a essa solução é que a autenticação em si não garante que os dispositivos ajam de maneira correta, no caso do exemplo, compartilhando arquivos íntegros e respeitando as limitações dos dispositivos quanto ao número de requisições que eles suportam. Assim, faz-se necessário o monitoramento constante das ações tomadas tanto pelos clientes quanto pelos provedores de serviços.

Seguindo este raciocínio, a outra abordagem para solucionar este problema em ambientes dinâmicos consiste em utilizar as experiências acumuladas através das interações, para controlar o acesso a serviços, além de possibilitar a escolha de parceiros para cooperação. Essas experiências poderiam ser compartilhadas, aumentando a quantidade de informações disponíveis, elevando assim a eficácia do sistema em detectar comportamento incorreto. A partir de informações como as experiências próprias e compartilhadas pode-se determinar a confiança nestes ambientes dinâmicos.

A confiança é um julgamento útil derivado de experiências. Além disso, ela pode ser entendida como uma forma de compreensão e adaptação à complexidade do ambiente e como um meio de adicionar robustez a dispositivos independentes [19] (ver capítulo 3 para a definição de confiança). Alguns modelos utilizando a confiança como conceito central para ambientes pervasivos foram propostos na literatura, como por exemplo, Almenarez *et al.* [2] e Capra *et al.* [5] (ver Seção 2.5 para outros modelos de confiança). No entanto, as soluções apresentadas não são completas, pois elas ou abordam a determinação da confiança associada ao comportamento dos clientes, ou abordam a determinação da confiança associada aos provedores.

Ainda, a maioria das soluções apresentadas que avaliam os provedores e que estão associadas a *middlewares* de descoberta de serviços avaliam os dispositivos como um todo, isto é, elas obtêm um valor de confiança para o dispositivo não fazendo uma avaliação baseada no serviço disponibilizado, o que reduz a eficiência da análise [28, 33].

1.2 Objetivos

O objetivo central deste trabalho é habilitar a análise de confiança em ambientes pervasivos. Para isso deve ser desenvolvido um modelo de confiança descentralizado capaz de estabelecer, analisar e monitorar o nível de confiança dos serviços nestes ambientes, utilizando as experiências acumuladas e compartilhadas nas interações entre os dispositivos.

É necessário ainda desenvolver um Gerenciador de Informação de Confiança (GIC) responsável por implementar este modelo de confiança, e um mecanismo de tomada de decisão. Além disso, este gerenciador deve tratar de aspectos relacionados à armazenagem das informações de confiança e sua forma de acesso.

Por fim, o GIC deve ser integrado ao *middleware* para computação pervasiva *Wings* [9], na forma de um serviço nativo. Isso irá possibilitar o acesso transparente para as aplicações às informações do GIC. Além disso, a partir desta integração, as aplicações pervasivas que utilizem o *middleware* poderão realizar a descoberta confiável de novos serviços.

1.3 Relevância

O desenvolvimento de aplicações pervasivas em cenários cada vez mais dinâmicos faz com que as mesmas necessitem de mais informações. As informações de confiança obtidas a partir da análise do comportamento dos dispositivos são de grande importância para que os dispositivos protejam seus serviços de uso indevido e utilizem de forma eficiente os serviços disponibilizados.

A integração de um modelo de confiança a um *middleware* permite que as aplicações possam lidar com a heterogeneidade e dinamicidade da computação pervasiva. A partir desta integração, as aplicações têm acesso às informações de confiança, o que as permite tomar decisões sem a interferência do usuário, alcançando desta forma a invisibilidade da computação ambicionada por Weiser.

1.4 Estrutura da Dissertação

O restante deste trabalho está organizado da seguinte maneira:

- **No Capítulo 2**, apresenta-se uma visão geral da computação pervasiva, suas características, mostrando sua relação com a confiança.
- **No Capítulo 3**, uma fundamentação teórica é apresentada, descrevendo as definições utilizadas no desenvolvimento do modelo de confiança
- **No Capítulo 4**, o modelo de confiança desenvolvido é apresentado.

- **No Capítulo 5**, a implementação do GIC é apresentada bem como os resultados obtidos através de simulação.
- **No Capítulo 6**, a integração do GIC ao *middleware Wings* é apresentada.
- **No Capítulo 7**, são apresentadas as considerações finais e os trabalhos futuros.

Capítulo 2

Computação Pervasiva

Neste capítulo é apresentada uma visão geral sobre a computação pervasiva, destacando suas principais características, além de contemplar uma visão resumida sobre a utilização de serviços nos ambientes pervasivos. A compreensão da confiança e de seus princípios básicos de formação também é contemplada. A partir de um cenário de exemplo, verifica-se a utilidade da determinação da confiança nestes ambientes. Apresenta-se um *middleware* desenvolvido no Laboratório de Sistemas Embarcados e Computação Pervasiva[1] denominado *Wings* ao qual será adicionado um suporte a busca/seleção de serviços confiáveis. Por fim, são apresentados os modelos de confiança propostos no contexto de sistemas pervasivos.

2.1 Visão geral sobre computação pervasiva

Em 1991, Mark Weiser escreveu o primeiro artigo sobre a computação pervasiva [32] que descreveu sua visão sobre uma nova era da computação, onde as tecnologias estariam integradas ao cotidiano das pessoas de tal forma que se tornariam invisíveis ao usuário. A essência da visão de Weiser está na criação de ambientes saturados de computação e capacidade de comunicação de modo que a computação esteja naturalmente integrada ao cotidiano das pessoas podendo estar presente em *qualquer lugar, a qualquer momento*. Essa visão era muito avançada para a época, quando simplesmente não havia tecnologia disponível para torná-la viável [25]. No entanto, os avanços tecnológicos dos últimos anos possibilitaram, por exemplo, a criação de dispositivos portáteis, como *smartphones* e PDAs, com diferentes tecnologias de comunicação, como *Wi-Fi*[2] e *Bluetooth*[3]. Esses e outros avanços tecnológicos possibilitam que a visão de Weiser se torne realidade atualmente.

[1] *http://embedded.ufcg.edu.br*
[2] *Wireless Fidelity*
[3] *www.bluetooth.com*

2.1.1 Características

A visão de Weiser é considerada um novo paradigma da computação [24] e, como tal, possui algumas características que o diferenciam de outros paradigmas da computação como, por exemplo, dos Computadores Pessoais. As principais características de tal paradigma são apresentadas a seguir.

Descentralização

No início, os que hoje são chamados de Computadores Pessoais, não passavam de terminais simples que dependiam de grandes e poderosos computadores centrais para realizar suas tarefas. Na chamada Era dos *mainframes*, a centralização de informações e capacidade de processamento era a principal característica. Em seguida, surgiu a Era dos Computadores Pessoais, que popularizou a computação, fazendo com que ela estivesse presente, através dos Computadores Pessoais, em casas e escritórios. Nesta era, que marcou o início do processo de descentralização da computação, os terminais passaram a possuir poder computacional para realizar as tarefas necessárias ao usuário [12].

O processo de descentralização ainda continua. Ele pode ser observado atualmente com o surgimento de dispositivos portáteis como *smartphones* e *handhelds*. Com esses dispositivos, o poder computacional e as responsabilidades na realização das tarefas do usuário são distribuídas possibilitando que a computação pervasiva esteja presente em qualquer lugar e a qualquer momento.

Invisibilidade

Segundo o paradigma definido por Weiser, a computação pervasiva deveria estar tão integrada ao cotidiano do usuário que desapareceria da ciência do mesmo. Isto é, o usuário realizaria suas tarefas fazendo uso da computação sem, no entanto, estar ciente disto. Esta visão difere bastante do modo como as tarefas são realizadas no paradigma da computação pessoal. Neste paradigma, os usuários vão em busca da computação em um Computador Pessoal para realizar sua tarefa e o deixam quando ela é concluída [24]. No caso da computação pervasiva, pode-se se aproximar da invisibilidade através da minimização da necessidade de interferência do usuário na realização das tarefas do dia-a-dia, exigindo assim que os dispositivos possuam algum grau de pró-atividade de modo a tomar decisões autonomamente [25].

Conectividade

Possibitar que a computação esteja presente em qualquer lugar a qualquer momento não é suficiente para torná-la pervasiva. Para isso, os dispositivos portáteis presentes nos ambientes pervasivos também devem ser capazes de interagir entre si com objetivo de trocar informações

facilitando a realização das tarefas do usuário. Para isso, faz-se necessário que eles possam se comunicar de maneira eficaz. Atualmente é possível encontrar dispositivos portáteis habilitados com diferentes tecnologias de comunicação de curto alcance como *Wi-Fi* e *Bluetooth*, e de longo alcance como GPRS[4] e CDMA[5]. É a partir destas diferentes possibilidades de comunicação que os dispositivos portáteis interagem entre si de forma autônoma, criando uma *rede dinâmica de relações*, que os possibilita acessar uma grande quantidade de informações de diferentes fontes.

Percepção: Ciência do contexto

No paradigma da computação pessoal, os Computadores Pessoais usualmente não acessam certo tipo de informações como o estado do ambiente, do usuário, etc. Estas informações, que à primeira vista parecem ser irrelevantes, possuem grande importância no paradigma da computação pervasiva. As chamadas informações de contexto [20], podem ser bastante diversificadas, incluindo a localização, o papel desempenhado pelo dispositivo, a tarefa sendo realizada, os dispositivos com os quais se comunica, etc. A importância em capturar estas informações de contexto está em possibilitar que os dispositivos se adaptem às mesmas para reduzir a necessidade de interferência do usuário [25], isto é, exigindo o mínimo de atenção do mesmo.

Dinamicidade

Os ambientes pervasivos estão repletos de dispositivos com capacidade de comunicação sem-fio. Por serem portáteis, estes dispositivos podem ser movidos entre ambientes, entrando e saindo dos mesmos livremente. Isso faz com que a comunicação entre os dispositivos seja estabelecida dinamicamente de acordo com a disponibilidade de dispositivos no ambiente. Essas características dos dispositivos portáteis faz com que os ambientes pervasivos sejam considerados dinâmicos.

2.2 Serviços em Ambientes Pervasivos

De acordo com o paradigma da computação pervasiva, ambientes pervasivos são compostos por dispositivos de diversos tipos capazes de se comunicar para realizar, de forma não intrusiva, as tarefas do usuário. Para que isso seja possível é essencial a utilização de serviços e protocolos de descoberta de serviços [37]. A utilização de serviços possibilita que as relações entre os dispositivos sejam definidas em tempo de execução garantindo um alto grau de flexibilidade [9]. Ainda, a utilização de serviços torna possível a comunicação entre dispositivos heterogêneos (com diferentes características) de maneira transparente. Os protocolos de descoberta de serviço

[4] *General Packet Radio Service*
[5] *Code Division Multiple Access*

por sua vez, possibilitam que novos serviços sejam encontrados dinamicamente à medida que o ambiente muda, através da entrada de novos dispositivos, por exemplo. A partir da descoberta de serviços, os dispositivos podem se comunicar diretamente para realizar as tarefas necessárias ao usuário. Dentre os protocolos de descoberta de serviço alguns dos principais são: UPnP, da Microsoft; Bluetooth SDP, do SIG[6]; Jini, da Sun Microsystems; e Service Location Protocol, do IETF[7]. Uma característica comum a estes protocolos é seu escopo de descoberta de serviços, que é limitado à topologia da rede na qual estão inseridos (LAN[8]), e ao limite de alcance do rádio no caso de protocolos baseados em tecnologias de comunicação sem-fio.

A partir disso, observa-se que os ambientes pervasivos estão repletos de dispositivos que anunciam e utilizam serviços dinamicamente. As associações entre os dispositivos ocorrem de forma dinâmica, levando em consideração as características dos ambientes pervasivos. Os dispositivos podem assumir o papel de clientes e servidores de serviço dependendo das requisições que recebem e das tarefas que devem realizar para o usuário.

2.3 Confiança em Ambientes Pervasivos

Para que a computação seja realmente pervasiva ela deve estar disponível em todos os lugares a qualquer momento, o que envolve ambientes abertos e dinâmicos. Para que isto seja possível, os dispositivos devem ser capazes de se comunicar de forma eficiente independentemente do local em que se encontram. As interações devem poder ocorrer, por exemplo, em um ônibus quando o usuário está a caminho do trabalho ou em um restaurante, sem que um servidor dedicado esteja disponível ou acessível.

Para que a comunicação entre os dispositivos possa ser realizada em qualquer local, faz-se uso de tecnologias de curto alcance que podem formar Redes Ad Hoc Móveis, também conhecidas por MANETS[9], ou estabelecer comunicação direta ponto-a-ponto. É através dessas tecnologias que serviços podem ser anunciados/descobertos de forma transparente possibilitando a adaptação às mudanças do ambiente.

Nos ambientes pervasivos, à medida que os dispositivos se comunicam, provendo e utilizando serviços, eles obtêm experiências sobre o comportamento uns dos outros. A partir destas experiências e de informações compartilhadas entre os dispositivos, eles podem determinar para quais clientes fornecer seus serviços e de quais servidores requisitar serviços. Esta idéia pode ser melhor compreendida a partir de um cenário de exemplo.

Suponha a existência de um ambiente pervasivo qualquer apresentado na Figura 2.1. Nele, estão localizados dois dispositivos desconhecidos um do outro que disponibilizam e utilizam

[6] *Bluetooth Special Interest Group*
[7] *Internet Engineering Task Force*
[8] *Local Area Network*
[9] *Mobile Ad Hoc Networks*

serviços, *H1* e *D1*. *D1* requisita um serviço disponibilizado por *H1*. Suponha que *H1* aceite o pedido e inicie a utilização do serviço (**1**). Ao mesmo tempo, *H1* inicia a coleta de experiências que tem com *D1* e em poucas interações determina, a partir das experiências, que *D1* não segue os critérios estabelecidos para a utilização do serviço em questão (ver Seção 3.3 para detalhes sobre critérios de avaliação). *H1* então, cessa as interações com *D1* protegendo assim seus recursos. Passado algum tempo, *H2*, que já teve contato com *H1*, chega ao ambiente. *D1* por sua vez requisita a *H2* a utilização do mesmo serviço provido por *H1* (**2**). *H2*, que desconhece *D1* no entanto, requisita a *H1* sua recomendação sobre *D1* (**3**), i.e. alguma informação que lhe ajude a determinar a natureza do comportamento de *D1*. *H1* compartilha as experiências que teve com *D1* possibilitando que *H2* negue o pedido de utilização de seu serviço por *D1*, protegendo também seus recursos.

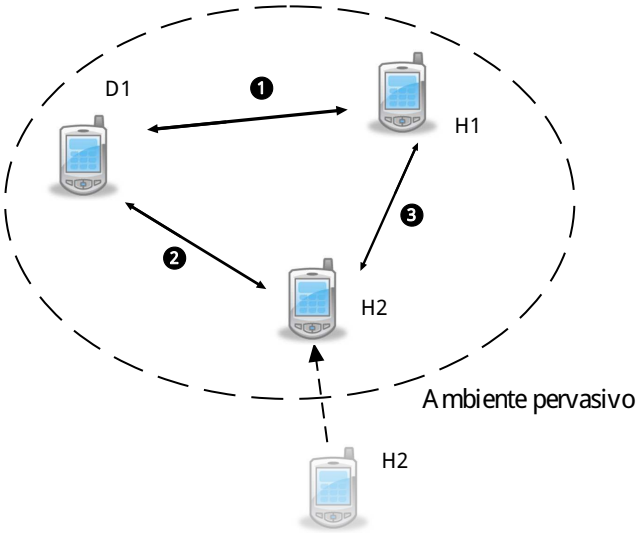


Figura 2.1: Interações baseadas em confiança em um ambiente pervasivo

A partir deste exemplo, observa-se que decisões sobre a utilização dos serviços podem ser tomadas com base em informações de experiências e recomendações. Estas duas informações formam as bases para a formação das relações de confiança. É importante notar então, que a confiança é a base para a tomada de decisão em ambientes pervasivos possibilitando que os dispositivos tomem decisões de forma autônoma.

2.4 O Middleware Wings

Os dispositivos em ambientes pervasivos podem ter diferentes tecnologias de comunicação e protocolos de anúncio/descoberta de serviços. *Middlewares* são utilizados para que não seja necessário o desenvolvimento de várias versões da mesma aplicação para lidar com tal diversidade. Esses *middlewares* devem disponibilizar um nível de abstração para as aplicações de

modo que a utilização de um serviço anunciado utilizando Bluetooth SDP ou UPnP, por exemplo, seja transparente. O *middleware* Wings [9] foi desenvolvido com este propósito. O Wings possui uma arquitetura baseada em *plug-ins* que são utilizados para Disponibilização de Serviços (PDS) e Plugin de Descoberta de Nós (PDN). Com essa arquitetura, é possível adicionar *plug-ins* que implementam diferentes soluções de rede e de protocolos de descoberta de serviço, reduzindo a complexidade para o desenvolvedor de aplicações voltadas para ambientes pervasivos.

Com essas facilidades, é de se esperar que as aplicações desenvolvidas para ambientes pervasivos utilizem *middlewares*. No entanto, faz-se necessário que estas aplicações sejam capazes de determinar o nível de confiança dos serviços dos dispositivos com os quais se comunicam. Além disso, as aplicações devem proteger seus serviços avaliando os clientes dos mesmos. É importante, então, unir as soluções de anúncio/descoberta de serviços bem como a verificação da confiança dos dispositivos e seus serviços em uma única solução. Conclui-se que se deve adicionar suporte à descoberta de serviços confiáveis ao Wings, possibilitando a escolha adequada de provedores bem como o controle de acesso de clientes aos serviços oferecidos.

2.5 Modelos de Confiança

Modelos computacionais para determinação da confiança não são uma novidade. Eles têm sido utilizados em várias áreas como sistemas ponto-a-ponto, comércio eletrônico, comunidades virtuais, etc. No entanto, apenas mais recentemente estes modelos têm sido utilizados em sistemas pervasivos. Almenarez *et al.* em [3] apresenta um modelo descentralizado para o gerenciamento de confiança para ambientes pervasivos. Ele permite a criação de relações de confiança de maneira *ad hoc*, além de monitorar o comportamento das entidades ao longo do tempo. Ele provê ainda, um protocolo que permite a troca de recomendações entre as entidades. Apesar dessas características, não são levados em consideração informações de contexto no estabelecimento das relações de confiança, como por exemplo, o serviço envolvido na comunicação, o que limita a flexibilidade na tomada de decisões. Almenarez *et al.* em [2], apresenta dois modelos de confiança cuja perspectiva principal é voltada para as entidades que anunciam serviços, isto é, permitindo apenas a análise do comportamento dos clientes.

Capra *et al.* em [5], apresenta um modelo de predição de confiança baseado em um filtro de Kalman [17]. Os autores argumentam que o modelo é leve do ponto de vista computacional propiciando a determinação autônoma do nível de confiança das entidades. Novamente, não são considerados aspectos relevantes como o contexto das interações nem a troca de recomendações. Além disso, apenas os provedores de serviço são avaliados.

Xiu *et al.* em [35] apresenta um modelo dinâmico de alto nível para o estabelecimento de relações de confiança. São consideradas diferentes fontes de informação no estabelecimento da

confiança como reputação, contexto e credenciais. No entanto, não há mecanismos para realizar a análise contínua do comportamento das entidades durante as interações, o que impede que variações no comportamento sejam detectadas com eficiência.

Com o desenvolvimento dos modelos de confiança, alguns deles foram empregados juntamente com *middlewares* para computação pervasiva. Por exemplo, Wolfe *et al.* [33] propõe o suporte à confiança ao *middleware* MARKS [29]. Este suporte é modelado como um serviço agrupado com outros serviços do middleware ditos essenciais. O modelo de confiança utilizado neste caso, leva em consideração o monitoramento contínuo da confiança através de experiências considerando ainda a utilização de recomendações na formação da confiança. No entanto, novamente este modelo não leva em consideração o contexto das interações. Sharmin *et al.* em [28] propõe um método para a descoberta segura de recursos em ambientes pervasivos. A proposta leva em consideração o monitoramento das ações dos dispositivos bem como a utilização de recomendações para determinar a confiança dos mesmos. No entanto, o método utilizado para agregar as recomendações provenientes das várias fontes disponíveis não distingue com eficiência recomendações verdadeiras das falsas.

Capítulo 3

Fundamentação Teórica

A computação pervasiva envolve ambientes abertos e dinâmicos com uma grande quantidade de dispositivos interagindo sem a presença de uma entidade central que regulamente tais interações. Nestes ambientes, entidades disponibilizam e utilizam serviços umas das outras possibilitando o compartilhamento de informações e facilitando, por parte das aplicações, a execução das tarefas necessárias ao usuário.

A confiança se torna essencial para a análise das relações que são estabelecidas entre as entidades nestes ambientes. Neste capítulo são apresentados os fundamentos teóricos para a determinação da confiança dessas relações. Os conceitos abordados abrangem confiança, relação de confiança, recomendação e reputação. É apresentada ainda, uma metodologia para determinação do valor de confiança das interações, que é uma das bases para a determinação do valor de confiança de uma relação.

3.1 Definindo a Confiança

Um dos primeiros trabalhos a utilizar confiança em sistemas computacionais é o apresentado por Marsh [19]. Além dele, existem vários outros, cada um fornecendo sua própria definição de confiança. Josang [16] é um dos mais citados. Para ele, confiança *"é a convicção que uma entidade tem em outra, obtida a partir de experiências passadas, conhecimento sobre a natureza da entidade e/ou recomendações de entidades confiáveis. Esta convicção expressa a crença no comportamento da entidade, que implica em um risco"*. Gambetta [11] define confiança como *"um nível particular de probabilidade subjetiva com o qual uma entidade irá realizar uma determinada ação, antes que se possa monitorar essa ação e em um contexto..."* Wang [31] define confiança como a *"crença de uma entidade na honestidade e capacidade de outra baseada em experiências diretas"*. Estas definições ajudam a compreender alguns aspectos importantes acerca da confiança. O primeiro deles está relacionado às informações que constituem a base para a formação da confiança. Josang determina que confiança é produto de experiências

passadas, da natureza da entidade e de recomendações provenientes de terceiros. O segundo está associado às características da confiança. Gambetta afirma que confiança é subjetiva e dependente do contexto. Além desta, a dinamicidade é uma característica não mencionada diretamente, mas que pode ser inferida pelo fato de a confiança ser constituída de experiências, que podem mudar. O terceiro diz respeito a quais características está associada a confiança. Segundo Wang, confiança está associada à honestidade e capacidade das entidades. No contexto deste trabalho define-se confiança como descrito na Definição 1.

Definição 1 (Confiança) *A crença subjetiva e dinâmica na honestidade e capacidade que uma entidade tem em outra, num determinado contexto, baseada em experiências passadas e/ou em recomendações provenientes de terceiros.*

3.1.1 Propriedades da Confiança

Uma vez definida a confiança, deve-se determinar as propriedades que são seguidas por ela. Dado três entidades A, B e C, e um contexto l definido entre elas. As propriedades apresentadas a seguir são obtidas de [3]:

- **Reflexiva**. Toda entidade confia em si mesma.
- **Não-simétrica**. Se uma entidade A confia numa entidade B num contexto l, não necessariamente B confia em A neste contexto.
- **Condicionalmente transitiva**. Se a entidade A confia em B num contexto l e B confia em C no contexto l, então A confia condicionamente em C neste contexto. A quantidade de confiança que A deposita em C depende da confiança que A deposita em B para emitir recomendações.
- **Dinâmica**. Confiança se modifica de acordo com as interações que ocorrem entre as entidades nos contextos.

3.2 Relação de confiança

Uma relação de confiança é determinada pela associação entre duas entidades e é caracterizada pelo nível de confiança, pelo contexto, e pelo tempo. Na Figura 3.1, apresenta-se a associação entre estes elementos. As relações de confiança são definidas pelas entidades observadoras e estão associadas a cada entidade avaliada com a qual se comunicam.

O contexto é uma das características mais importantes na determinação do valor de confiança de uma relação. Isso ocorre porque duas entidades podem possuir diferentes valores de confiança entre si dependendo do contexto em que as interações ocorreram. Por exemplo, existe

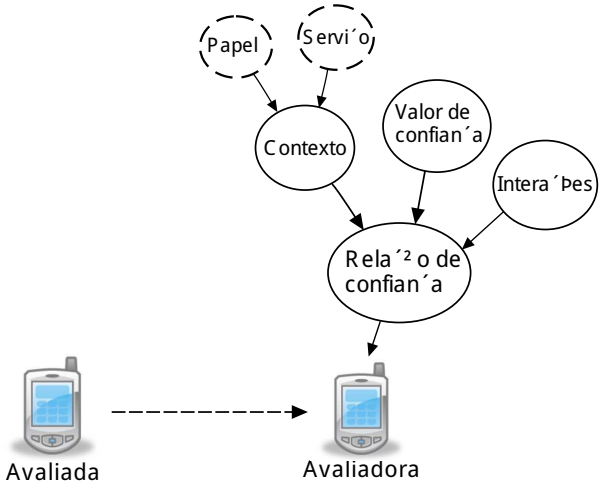


Figura 3.1: Elementos associados a uma relação de confiança

uma grande diferença entre confiar em uma pessoa para dirigir um carro e confiar nesta mesma pessoa para pilotar um avião. Trata-se da mesma pessoa que, no entanto, deve ser vista sobre óticas de capacidades diferentes pois se trata de duas tarefas diferentes, i.e. a habilidade de realizar uma tarefa não implica em habilidade para realizar a outra. Assim, o contexto nas relações de confiança caracteriza unicamente a situação na qual a interação está ocorrendo. Este contexto, como explicado na Seção 2.1, pode ser por exemplo, o serviço sendo utilizado, o papel que uma entidade possui na relação de confiança (provedor ou cliente), etc. Com isso, as experiências passadas obtidas em cada contexto podem ser utilizadas para auxiliar na determinação do comportamento das entidades quando o mesmo contexto for identificado no futuro. No contexto deste trabalho o contexto de uma relação de confiança é determinado pelo serviço e pelo papel desempenhado pela entidade avaliada como pode ser observado na Figura 3.1.

Outra característica da relação de confiança observada a partir da Figura 3.1 são as interações. A relação de confiança evolui de acordo com as interações que ocorrem nos contextos específicos entre as entidades.

Por fim, associa-se a cada relação de confiança um valor de confiança que caracteriza a quantidade de confiança depositada na relação. Basicamente há duas formas de representação deste valor: a primeira, numérica, baseia-se em uma faixa de valores de números inteiros ou reais; a segunda, chamada de não numérica, baseia-se em etiquetas de classificação que definem a semântica da informação. Um exemplo de representação não numérica consiste na classificação da confiança como baixa, média ou alta. Em [1] as relações de confiança são representadas através das etiquetas: não confiável, ignorância, mínimo, médio, bom ou completo. Fabrizio Cornelli *et al* propõem em [7] um esquema não numérico em termos de quantidade de '*', onde quanto maior o número de '*' maior o nível de confiança. Em [28] e [21] os autores fazem uso de números reais para representar os valores de confiança com faixas [0,1] e [-1,1] respectivamente. Neste trabalho utiliza-se uma representação contínua na faixa $[0, 1]$ onde 0 representa a

total falta de confiança e 1 representa confiança completa. Esta faixa é utilizada, pois reflete a natureza contínua da confiança.

3.3 Determinando a Confiança de uma Interação

A determinação do valor de confiança é o processo através do qual uma entidade, chamada de avaliadora ou observadora, determina o valor de confiança de uma outra entidade denominada avaliada ou observada. Essa determinação, que se baseia na análise de comportamento, deve ser feita a cada interação da relação de confiança e possibilita que o valor de confiança da relação possa ser calculado.

3.3.1 Metodologia

Para determinar o valor de confiança em uma interação deve-se tomar como base a relação entre o comportamento esperado e o comportamento obtido da entidade sob observação [13]. A partir desta relação pode-se determinar o valor de confiança da interação em num determinado contexto.

Para determinar o valor de confiança de uma interação são necessários critérios e métricas de avaliação. Os critérios de avaliação devem ser escolhidos de modo a contemplar o contexto da relação de confiança. Isto é, para cada relação de confiança, que está associada a um contexto, deve-se ter um conjunto de critérios de avaliação. Como mencionado anteriormente, o contexto de uma relação de confiança pode ser definido, de forma geral, pelo serviço utilizado e pelo papel assumido pela entidade que está sendo analisada. Na Figura 3.2 apresenta-se um exemplo. Nesta figura observa-se que uma entidade x anuncia um serviço A e uma entidade y utiliza este serviço. A entidade x então, deve avaliar a utilização do serviço A por y de acordo com o conjunto de critérios determinado pelo contexto definido por A e pelo papel desempenhado pela entidade observada, um cliente. De forma semelhante, y avalia a disponibilização de A por x. Desta forma, x possui um conjunto de critérios específicos para avaliar a utilização do serviço A por parte dos clientes (y) e y possui um conjunto de critérios específicos para avaliar a disponibilização do serviço A por parte dos provedores (x).

Em um serviço de compartilhamento de arquivos, por exemplo, onde é possível a existência de arquivos falsos ou até mesmo corrompidos, o estado do arquivo recebido (arquivo correto, sem erros, etc.) pode ser classificado como um critério de avaliação da disponibilização do serviço. Outros critérios incluem o tempo de resposta e o número de mensagens num intervalo de tempo. Já os critérios de avaliação da utilização do serviço podem ser, por exemplo, o número máximo de requisições de arquivos por entidade num intervalo de tempo, número máximo de pedidos de busca por arquivos, etc.

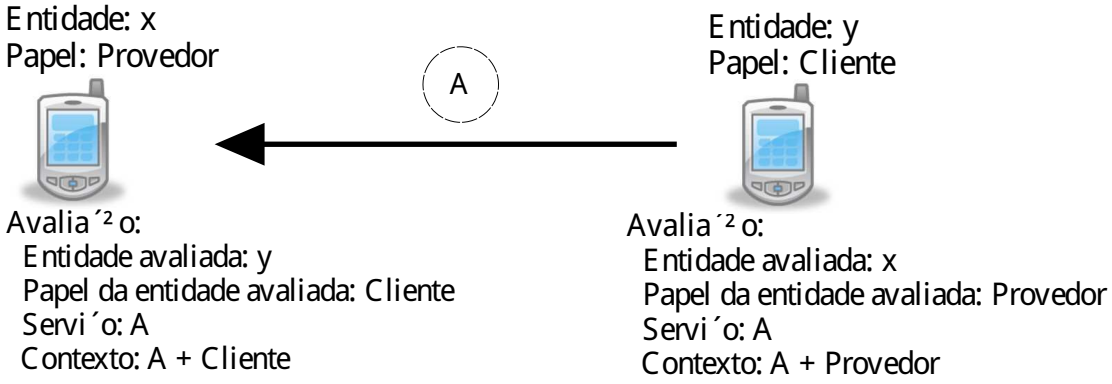


Figura 3.2: Contexto de avaliação de um serviço

O conjunto de critérios de avaliação pode ser definido *a priori*, juntamente com a definição do serviço. Outra opção consiste em definir um conjunto de critérios gerais para avaliação de serviços em ambientes pervasivos. Por fim, pode-se ainda realizar a negociação dos critérios de avaliação através de algum protocolo específico para este fim. A determinação da opção mais favorável para a determinação dos critérios de avaliação está fora do escopo deste trabalho. Assim, assume-se que os critérios de avaliação do serviço são conhecidos pelas entidades que fazem uso dos mesmos. A Definição 2 apresenta o conjunto de critérios de avaliação.

Definição 2 (Critérios de avaliação) *Define-se o conjunto dos critérios de avaliação de um determinado serviço como sendo:*

$$C = C_{fs} \cup C_{cs} \tag{3.1}$$

Onde C_{fs} e C_{cs} são disjuntos com C_{fs} correspondendo ao conjunto dos critérios de avaliação da disponibilização do serviço e C_{cs} ao conjunto de critérios de avaliação da utilização do serviço.

Métricas de Avaliação

Uma vez determinados os critérios através dos quais os serviços serão avaliados, deve-se determinar que métricas utilizar para avaliar tais critérios. As métricas de avaliação devem ser aplicadas aos critérios definidos para cada contexto a cada interação. Neste trabalho, são utilizadas duas métricas, baseadas em [8]. A primeira delas é o comprometimento ($Comp$). Através dela, cada entidade avalia o quanto a sua entidade par se comprometeu em cumprir um determinado critério. Isto é, para cada critério, a entidade avaliadora determina o grau de comprometimento da entidade avaliada. $Comp$ pode assumir 6 níveis distintos a depender do grau de comprometimento do critério em análise os quais variam de 0 a 5 com as respectivas semânticas: 'nenhum comprometimento','pouco comprometimento', 'comprometimento parcial', 'comprometimento razoável', 'comprometimento elevado' e 'comprometimento total'.

É necessário ainda que cada critério seja avaliado de acordo com o seu grau de influência em relação aos demais critérios. Assim, a segunda métrica utilizada é a influência ($Infl$). Esta métrica possui 6 níveis que variam de 0 a 5 com as seguintes semânticas: 'não importante','pouco importante','parcialmente importante','largamente importante' e 'muito importante'.

A avaliação de um critério c, dado que $c \in M$ onde $M = C_{fs}$ ou $M = C_{cs}$, é uma função determinada pelas métricas de avaliação de comprometimento ($Comp_c$) e influência ($Infl_c$). Na Equação 3.2, apresenta-se a função de avaliação $Aval_c$ que é o produto das métricas de avaliação.

$$Aval_c = Comp_c Infl_c \tag{3.2}$$

Avaliação e valor de confiança

Para avaliar o conjunto de critérios M deve-se somar a avaliação de cada um dos critérios do conjunto. Na Equação 3.3, apresenta-se esta avaliação.

$$\begin{aligned} Aval^M &= \sum_{c=1}^{|M|} Aval_c \\ &= \sum_{c=1}^{|M|} (Comp_c Infl_c) \end{aligned} \tag{3.3}$$

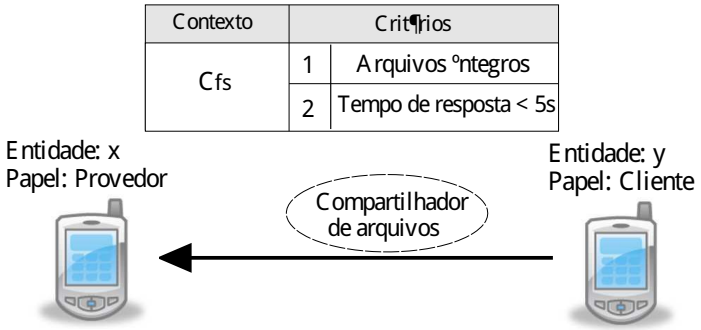
Pode-se, então estabelecer que a determinação da confiança para uma dada interação depende da avaliação dos critérios aos quais ela está sujeita, como definido na Equação 3.3. No entanto, faz-se necessário ainda que estes valores estejam na faixa $[0, 1]$, como definido na Seção 3.2. Para isso, pode-se relacionar a avaliação dos critérios ao seu valor máximo. Na Equação 3.4, apresenta-se este resultado utilizando o valor máximo de comprometimento $MaxComp_c = 5$, que significa o maior comprometimento possível de um critério.

$$MaxAval^M = \sum_{c=1}^{|M|} (MaxComp_c Infl_c) \tag{3.4}$$

Definição 3 (Valor de confiança de uma interação) *O valor de confiança de uma interação i que é calculado pela relação entre $Aval^M$ e $MaxAval^M$ é dada por $\widehat{\tau^M}$ podendo assumir valores na faixa $[0, 1]$.*

$$\begin{aligned} \widehat{\tau^M}(i) &= \frac{Aval^M}{MaxAval^M} \\ &= \frac{\sum_{c=1}^{|M|} (Comp_c Infl_c)}{\sum_{c=1}^{|M|} (MaxComp_c Infl_c)} \end{aligned} \tag{3.5}$$

A determinação do valor de confiança de um provedor em uma interação é exemplificado na Figura 3.3. Neste exemplo utiliza-se um serviço de compartilhamento de arquivos com dois critérios de avaliação: integridade do arquivo recebido e tempo de resposta para início da transferência. O conjunto de critérios de avaliação do provedor (C_{fs}) é utilizado pela entidade y para determinar o valor de confiança do provedor x em uma interação qualquer. Supondo uma interação que resulte no recebimento de um arquivo íntegro com um tempo de resposta de 6s. Neste caso, o recebimento de um arquivo íntegro leva a avaliação máxima de comprometimento. Já a métrica de comprometimento do critério de avaliação do tempo de resposta é igual a 4 para um tempo de $6s$. Estes valores das métricas obtidos para esta interação são apresentados na Figura 3.3 e, como mencionado anteriormente, são definidos *a priori* para cada serviço. O valor resultante para a confiança da interação é determinado pela Equação 3.5 e é igual a $0,93$.



Crit¶rio (c)	$Comp_c$	$Infl_c$	$Aval_c$ / $MaxAval_c$	$Aval^{Cfs}$	$MaxAval^{Cfs}$	$\widehat{Cfs}$
Arquivos ºntegros	5	5	25 / 25	37	40	0,93
Tempo de resposta = 6s	4	3	12 / 15			

Figura 3.3: Determinação do valor de confiança de uma interação com um provedor

A definição do valor de confiança de uma interação (Definição 3) pode ser utilizada tanto pelo provedor de serviço como pelo cliente na determinação do valor de confiança. No entanto ela abrange apenas o contexto representado por um conjunto de parâmetros M. Como mencionado anteriormente, o valor de confiança está associada diretamente à relação de confiança, que por sua vez, está relacionada a duas entidades, avaliadora e avaliada, e ao contexto. Desta forma, supondo que uma entidade x é a avaliadora em uma relação de confiança, com a entidade y sendo avaliada num contexto definido por l, o valor de confiança da relação em uma interação é representada de forma mais geral por $\widehat{\tau_y^{x,l}}(i)$. Esta notação é utilizada no restante do trabalho.

3.4 Recomendações

Como pode ser observado no exemplo apresentado na Seção 2.3, a troca de experiências entre as entidades é bastante importante na determinação da confiança principalmente quando a entidade

avaliada é desconhecida. Essa troca de experiências é feita através de recomendações.

Recomendações são formadas pelas experiências adquiridas através da observação do comportamento de uma determinada entidade. Essas experiências estão associadas ao contexto das interações, como definido anteriormente. Isso implica que as recomendações também devem considerar o contexto. Além disso, uma recomendação deve especificar a entidade emissora e a entidade recomendada. No contexto deste trabalho, recomendação é descrita na Definição 4.

Definição 4 (Recomendação) *Recomendação é a informação de experiência de uma entidade x acerca de uma entidade y num contexto l representada por $Rec(x, y, l)$.*

3.5 Reputação

Reputação é a avaliação de uma entidade criada a partir do ponto de vista de outras entidades. Essa avaliação é determinada pelas experiências obtidas a partir de recomendações. Uma entidade não é capaz de decidir sobre sua reputação diretamente. Ela pode apenas influenciá-la alterando seu comportamento. Essa é uma das características da reputação. As demais características são: dinamicidade, subjetividade e dependência do contexto. Estas últimas são decorrentes da associação que reputação tem com o conceito de confiança. No contexto deste trabalho, reputação é considerada um tipo de confiança específico que é proveniente de recomendações. A definição da reputação é apresentada na Definição 5.

Definição 5 (Reputação) *Reputação é a crença na capacidade e honestidade de uma entidade num certo contexto baseada na análise de avaliações de terceiros obtidas através de recomendações. A reputação de uma entidade x avaliada por uma entidade y num contexto l é dada por $\eta_x^{y,l}$.*

3.6 Sumário

Neste capítulo foram apresentados os conceitos necessários para o desenvolvimento de um modelo de confiança baseado em experiências diretas e recomendações. Estes conceitos incluem a confiança, relação de confiança, recomendação e reputação. Foi apresentada ainda uma metodologia para determinar o valor de confiança das interações. Este valor de confiança é utilizado como base para a determinação do modelo de evolução da confiança apresentado no capítulo a seguir.

Capítulo 4

Modelo de Confiança

Neste capítulo é apresentado um modelo desenvolvido com o objetivo de determinar o valor de confiança das relações de confiança. Este valor é baseado nas principais informações utilizadas para a formação da confiança que são as obtidas pela própria entidade e de terceiros, através de recomendações. Inicialmente são apresentados os requisitos do modelo de confiança que são baseados nas características e restrições próprias dos ambientes pervasivos. Apresenta-se o modelo para a evolução da confiança a partir do valor de confiança das interações. Em seguida, é apresentada a forma de determinação da reputação das entidades a partir de um método de combinação das recomendações, bem como um modelo simples para determinação da credibilidade dos recomendadores possibilitando que recomendações falsas sejam descartadas. Por fim, é apresentada a forma de determinação do valor de confiança geral que é obtido a partir da combinação do valor de confiança dinâmico e da reputação.

4.1 Requisitos

A computação pervasiva permite que as interações entre as entidades ocorram em qualquer lugar a qualquer momento, sempre que necessário, com o objetivo de auxiliar o usuário nas suas tarefas do dia-a-dia. As aplicações pervasivas devem agir de forma a minimizar a necessidade de interferência do usuário, assumindo comportamento autônomo sempre que possível.

Baseando-se nesta visão e considerando que as interações entre os dispositivos podem ocorrem em locais sem infra-estrutura, determina-se que os requisitos do modelo de confiança são:

- Descentralizado: a partir do cenário imaginado por Weiser pode-se observar que a consideração da existência de uma entidade controladora central que venha a facilitar o estabelecimento de comunicação entre as entidades não é aceitável. Isto por que a comunicação, como requer a computação pervasiva, deve poder ocorrer em qualquer lugar. Assim, é necessário que o modelo de confiança seja descentralizado, computado em cada entidade.

- Independente de infraestrutura: o modelo de confiança não deve depender de nenhum tipo de infraestrutura para seu pleno funcionamento. Assim, os valores de confiança das entidades derivados pelo modelo devem ser frutos de informações colhidas pelo mesmo sem a dependência de um servidor específico para este fim.

- Cooperativo: deve ser capaz de utilizar o conhecimento comum em seu benefício. Este conhecimento é obtido a partir das outras entidades através de recomendações.

- Descartar informações falsas: recomendações estão expostas a ações de entidades que venham a fornecer informações falsas com objetivo de aumentar ou diminuir o valor de confiança de uma outra entidade no ambiente. O modelo de confiança deve ser capaz de determinar a qualidade das recomendações evitando este tipo de ação.

4.2 Confiança obtida a partir de experiências diretas

As experiências diretas, que ocorrem entre duas entidades, são de grande importância na determinação do valor de confiança das relações de confiança. A partir destas experiências é possível monitorar o comportamento das entidades determinando o quanto elas seguem os critérios de avaliação. Esse monitoramento é feito continuamente através da determinação do valor de confiança de interação ($\widehat{\tau_y^{x,l}}(i)$), como definido no Capítulo 3. Deste modo, a cada interação entre as entidades, um novo valor de confiança está disponível.

Os diferentes valores de confiança correspondem à avaliação do comportamento da entidade apenas em uma interação, podendo ser independentes entre interações sucessivas. Isto significa que eles não levam em consideração uma das bases para a construção da confiança, as experiências passadas. Ainda, por serem independentes, estes valores de confiança não capturam a evolução da confiança.

Em razão disto, faz-se necessário empregar um modelo de evolução da confiança que leve em consideração as avaliações feitas a cada interação, montando a partir delas um histórico de interações que possa ser utilizado para determinar a forma de evolução da confiança. Este modelo faz uso do valor de confiança das interações e resulta em um valor de confiança que possui caráter dinâmico e evolutivo. O valor de confiança gerado pelo modelo é denominado *valor de confiança dinâmico* e é representado por $\tau_y^{x,l}(i)$.

4.2.1 Modelo de evolução da confiança

O modelo de evolução da confiança deve representar de forma consistente o valor de confiança das relações entre as entidades. Essa necessidade surge porque as entidades podem tentar obter maiores ganhos na utilização dos serviços variando entre períodos em que seguem os critérios

de avaliação dos contextos para cada interação e períodos em que não seguem. O modelo de evolução da confiança deve ser capaz de detectar e minimizar os efeitos deste tipo de comportamento. Para que isso seja possível, o modelo não deve estar suscetível a elevações passageiras do valor de confiança da interação, capturando apenas tendências consistentes de mudanças. Além disso, o modelo deve detectar rapidamente reduções nos valores de confiança da interação, refletindo diretamente os novos valores no *valor de confiança dinâmico*. Isso é feito tendo como informações o valor de confiança das interações e o histórico de experiências.

Para melhor compreensão das informações de confiança associadas ao modelo de evolução, apresenta-se na Figura 4.1 o fluxo da informação de confiança. Este fluxo tem origem no conjunto de critérios e métricas de avaliação do contexto. A partir deste conjunto é determinado o valor de confiança da interação e, a partir desta informação, o modelo de evolução determina o valor de confiança dinâmico da relação de confiança.

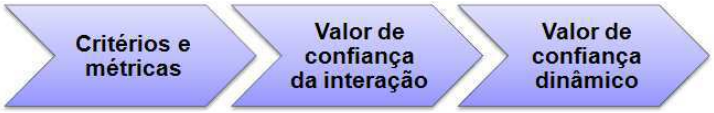


Figura 4.1: Propagação da informação

A partir disso, determina-se as principais características do modelo de evolução da confiança:

- Baseado em experiências. O modelo de evolução da confiança se baseia em experiências passadas provenientes de interações diretas entre entidades.
- Representa a confiança de modo consistente.
- Assimétrico. O modelo deve ser capaz de aumentar e diminuir o valor de confiança de uma relação de modo diferente para capturar o aspecto natural da confiança que faz com que ela seja difícil de se obter e fácil de se perder [36].
- Os valores máximo e mínimo do valor de confiança dinâmico estão limitados aos valores máximo e mínimo do valor de confiança da interação.

Uma vez definidas as características necessárias ao modelo de evolução, pode-se determinar sua forma de implementação. A alternativa mais simples consiste na utilização de uma função de atualização.

Tem-se assim que o valor de confiança dinâmico $\tau_y^{x,l}(i)$ é calculado a partir de uma função de atualização que utiliza o valor anterior adicionado a um degrau de evolução. Ela é apresentado na Equação 4.1

$$\tau_y^{x,l}(i) = \tau_y^{x,l}(i-1) + g(\cdot) \tag{4.1}$$

Onde, $g(\cdot)$ é uma função que determina o acréscimo ou decréscimo do valor e depende do histórico de interações e da diferença entre o valor de confiança da interação atual ($\widehat{\tau_y^{x,l}}(i)$) e o valor de confiança dinâmico anterior ($\tau_y^{x,l}(i-1)$). Esta diferença é determinada por $d = \widehat{\tau_y^{x,l}}(i) - \tau_y^{x,l}(i-1)$.

Uma vez definida a função de atualização do valor de confiança dinâmico, faz-se necessário determinar a forma de evolução da função $g(\cdot)$. Devido à característica assimétrica do modelo de evolução, considera-se a existência de duas funções $g^+(\cdot)$ e $g^-(\cdot)$ que representam a função de acréscimo e a função de decréscimo do valor de confiança dinâmico, respectivamente. A utilização de cada uma delas é determinada pelo valor de d, assim como apresentado na Equação 4.2. Isto é, quando $d \geq 0$, o valor de confiança da interação é maior que o dinâmico, o que resulta na utilização da função de acréscimo no valor de confiança dinâmico. Caso contrário, a função de decréscimo deve ser utilizada.

$$\tau_y^{x,l}(i) = \tau_y^{x,l}(i-1) + \begin{cases} g^+(\cdot) & \text{se } d \geq 0 \\ g^-(\cdot) & \text{se } d < 0 \end{cases} \tag{4.2}$$

Antes de prosseguir na determinação das funções $g^+(\cdot)$ e $g^-(\cdot)$ deve-se determinar o histórico das interações. Este histórico deve consistir de informações sobre o comportamento passado da entidade na relação de confiança. A diferença entre o valor de confiança da interação e o dinâmico pode ser utilizado com este propósito. Determina-se assim, que o histórico das interações é constituído de um conjunto com W amostras desta diferença (d) obtidas a cada nova interação. Como determinado pelas características do modelo, o degrau de evolução do valor de confiança dinâmico, especificado por $g^+(\cdot)$, deve ser determinado de modo a não ser suscetível a mudanças passageiras no valor de confiança da interação. Isso pode ser feito através do histórico das interações, elevando-se o valor de confiança dinâmico apenas quando for observado no histórico consistência no comportamento. Existe uma família de funções matemáticas capazes de modelar esta característica. Essas funções são chamadas de funções logísticas ou funções S, pois apresentam forma semelhante a um S. As funções desta família encontram uso em várias áreas incluindo economia, química e biologia. Sua maior utilidade está em modelar o crescimento. Uma forma generalizada desenvolvida para uso empírico foi criada por Richard tornando-se conhecida como função de Richard [23]. Esta função é utilizada como base para a determinação de $g^+(\cdot)$, que por sua vez é apresentada na Equação 4.3.

$$g^+(Win(i-1), d) = \frac{w_p d}{1 + e^{Wd - Win(i-1)}} \tag{4.3}$$

Onde $Win(i-1)$ é uma medida do histórico dada pela Equação 4.4. Esta medida é determinada pelo somatório das entradas do histórico que consistem na diferença entre o valor de confiança das interações e o valor de confiança dinâmico. Além disso, w_p é um fator que determina a taxa de crescimento do valor de confiança dinâmico.

$$Win(i) = \sum_{k=1}^{W} (\widehat{\tau_y^{x,l}(i-k)} - \tau_y^{x,l}(i-k)) \tag{4.4}$$

Analisando a Equação 4.3, observa-se que enquanto não há experiências suficientes no histórico, isto é, enquanto $Wd - Win(i-1) > 0$, o valor de confiança dinâmico mantém-se aproximadamente estável. Quando as experiências no histórico são suficientes para que $Win(i-1) > Wd$ o valor de confiança dinâmico cresce a uma taxa aproximada de $w_p d$.

Uma vez determinada $g^+(\cdot)$, o próximo passo consiste em determinar a função $g^-(\cdot)$. Esta função é responsável por reduzir o valor de confiança dinâmico sempre que houver uma redução no valor de confiança das interações. Para refletir o novo valor de confiança da interação na confiança dinâmica utiliza-se o valor de d multiplicado por um fator que determina a taxa de decrescimento do valor de confiança dinâmico, dado por w_n. Desta forma, quanto maior a discrepância entre os valores da confiança da interação e da confiança dinâmica, maior a redução no valor deste último. Esta função é apresentada pela Equação 4.5.

$$g^-(d) = w_n d \tag{4.5}$$

Com ambas as funções $g^+(\cdot)$ e $g^-(\cdot)$ determinadas, o valor de confiança dinâmico de uma relação de confiança é apresentado na Definição 6.

Definição 6 (Confiança dinâmica) *O valor de confiança dinâmico de uma entidade y avaliada por x em uma relação de confiança caracterizada pelo contexto l em uma interação i assume a seguinte forma:*

$$\tau_y^{x,l}(i) = \tau_y^{x,l}(i-1) + \begin{cases} \frac{w_p d}{1+e^{Wd-Win(i-1)}} & \textit{se } d > 0 \\ w_n d & \textit{se } d \leq 0 \end{cases} \tag{4.6}$$

4.2.2 Confiança em provedores e clientes

O valor de confiança de uma relação está associado a um contexto específico que determina a situação em que as interações ocorrem. Uma entidade pode estabelecer diversas relações de confiança com outra entidade diferenciando-as pelo seu contexto. À primeira vista pode parecer que todas essas relações de confiança não possuem relação entre si. No entanto, observa-se que elas podem ser divididas entre as que estão relacionadas à disponibilização e à utilização de serviços. A partir desta divisão, pode-se determinar o valor de confiança de uma entidade relacionada à disponibilização e outro associado à utilização do serviço. Estes novos valores de confiança não estão associados diretamente a contextos de serviços, mas a um contexto definido apenas pelo papel desempenhado pelas entidades nas relações de confiança. A partir disso, pode-se ter uma visão geral acerca do comportamento das entidades associados aos contextos de disponibilização e utilização de serviço. Essa visão é útil, por exemplo, quando não está

disponível o valor de confiança em um contexto específico, podendo ser utilizada no lugar o valor de confiança em um contexto mais abrangente, definido pelo papel desempenhado pela entidade (provedor ou cliente).

Estes novos valores de confiança são determinados a partir da combinação dos valores de confiança das relações cujas entidades observadas estão associadas ao mesmo papel. Para determinar a confiança em um provedor de serviço, por exemplo, pode-se determinar a média dos valores de confiança das relações associadas a este papel. Uma outra abordagem consiste em ponderar o valor de confiança de cada serviço de acordo com sua utilidade [19] para em seguida determinar a média. A utilidade de um serviço é obtida pesando-se custos e benefícios da utilização do mesmo e pode ser definida juntamente com os critérios de avaliação. Ela pode assumir valores na faixa $[0, 1]$ e sua utilização traz maior flexibilidade na determinação da confiança de provedores e clientes, pois possibilita que o valor de confiança dinâmico dos contextos tenham diferentes influências na formação do novo valor de confiança. Por exemplo, pode-se determinar que um serviço de data e hora tenha menor influência na formação da confiança do que de um serviço de compartilhamento de arquivos. A seguir, nas Definições 7 e 8, são apresentados os valores de confiança de provedores e clientes de serviço considerando-se que a utilidade do contexto é determinada pela utilidade do serviço associado.

Definição 7 (Valor de confiança de um provedor) *O valor de confiança de um provedor x avaliado pela entidade y é dado pela média do produto dos valores de confiança dinâmicos de cada contexto $l \in FS_{x,y}$, onde $FS_{x,y}$ é o conjunto de contextos estabelecidos entre x e y associados a disponibilização de serviço, pelo valor de utilidade de cada contexto.*

$$\tau_x^{y,fs} = \frac{\sum_{l \in FS_{x,y}} U^l \tau_x^{y,l}}{|FS_{x,y}|}$$

Definição 8 (Valor de confiança de um cliente) *O valor de confiança de um cliente x avaliado pela entidade y é dado pela média do produto da confiança dinâmica de cada contexto $l \in CS_{x,y}$, onde $CS_{x,y}$ é o conjunto de contextos estabelecidos entre x e y associados a utilização de serviço, pela utilidade.*

$$\tau_x^{y,cs} = \frac{\sum_{l \in CS_{x,y}} U^l \tau_x^{y,l}}{|CS_{x,y}|}$$

4.3 Confiança obtida a partir de experiências indiretas

Informações de confiança provenientes de interações diretas possibilitam que entidades conhecidas tenham seu comportamento determinado. No entanto, devido à natureza dinâmica da computação pervasiva, entidades tornam-se indisponíveis enquanto novas e desconhecidas surgem. Para obter máximo proveito da computação pervasiva, entidades desconhecidas devem ser

consideradas, pois representam possíveis novos colaboradores. Nestes casos, observa-se que informações de confiança provenientes de interações diretas entre entidades não são suficientes para alcançar o maior desempenho. Outras fontes de informação, além das interações diretas, devem ser utilizadas pela entidade para avaliar as demais que estão disponíveis. A principal consiste na utilização de recomendações.

O compartilhamento de experiências através de recomendações é bastante útil em casos onde não existe nenhuma informação sobre a entidade em análise, o que pode ocorrer nos ambientes pervasivos devido à natureza dinâmica de suas entidades. Além disso, recomendações corretas, podem elevar a qualidade da determinação da confiança, mesmo quando a relação de confiança já está estabelecida. Na Figura 4.2 ilustra-se um cenário onde as recomendações são utilizadas. Neste cenário, o dispositivo y recebe uma requisição de um dispositivo desconhecido, x (**1**). Por não possuir nenhuma experiência com x, y requisita recomendações aos dispositivos presentes no ambiente pervasivo $p1$, $p2$ e $p3$ (**2**). Estes, por sua vez, enviam a y suas recomendações (**3**).

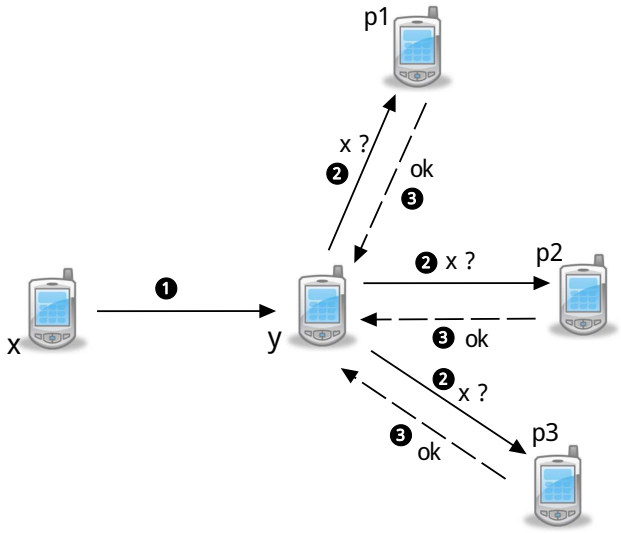


Figura 4.2: Utilização de recomendações

No entanto, o emprego deste recurso apresenta suas dificuldades. Uma das principais está em distinguir recomendações falsas de recomendações verdadeiras. Existem algumas razões pelas quais uma entidade teria este tipo de comportamento como explicado por Liu *et al.* em [18]. Uma delas consiste no não repasse de recomendações positivas de um provedor com recursos limitados com o objetivo de manter exclusividade de acesso ao mesmo. Outra questão é que essa técnica de distribuição de informações de experiência pode ser alvo de um tipo de ação conhecido como colusão. Neste tipo de ação, formam-se grupos de recomendadores que emitem recomendações elevadas para membros do grupo e recomendações baixas para entidades que não fazem parte dele, com o objetivo, por exemplo, de enganar outras entidades que requisitam recomendações e obter acesso a seus serviços. Na Figura 4.3 ilustra-se uma ação de colusão.

Nesta figura, as entidades x, $p2$ e $p3$, formam um grupo que emite recomendações falsas, enquanto as entidades y e $p1$ agem normalmente. A entidade x que faz parte do grupo, emite uma requisição a entidade y. Esta, por não possuir informações sobre x, requisita recomendações às entidades presentes, $p1$, $p2$ e $p3$. Estas entidades respondem, mas $p2$ e $p3$ que fazem parte da colusão, emitem recomendações falsas, isto é, diferente do comportamento real da entidade, afetando o julgamento de y. Este tipo de ação utilizando recomendações não deve impedir que a reputação de uma entidade seja determinada corretamente.

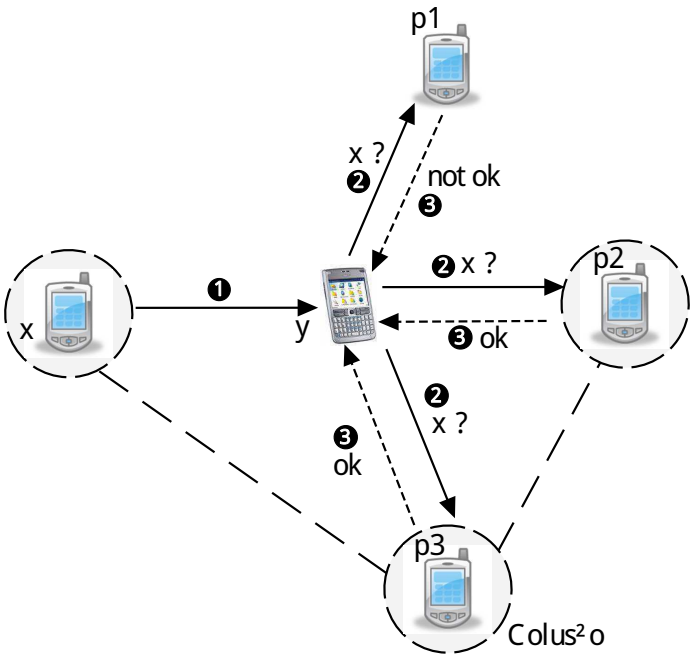


Figura 4.3: Colusão

4.3.1 Determinação da Recomendação

Um pedido de recomendação está associado a uma entidade a ser avaliada e a um contexto. Ao receber este pedido, a entidade deve respondê-lo, fornecendo informações de experiências sobre a entidade e contexto especificados. Duas opções de informações podem ser fornecidas: o valor de confiança da interação ou o valor de confiança dinâmico. O primeiro determina a confiança em um contexto na última interação realizada entre as entidades. O valor de confiança dinâmico, por sua vez, leva em consideração mais informações como o histórico das interações e a dinâmica de evolução da confiança. Em razão disto, este valor é utilizado para determinar as recomendações. Na Equação 4.7 é apresentada a fórmula utilizada para determinar as recomendações.

$$Rec(x, y, l) = \tau_x^{y,l} \tag{4.7}$$

Observa-se a partir da Figura 4.2 que, ao emitir um pedido de recomendação, uma entidade pode receber respostas de vários dispositivos. Estas respostas contêm a confiança dinâmica obtida por cada uma delas nas interações com a entidade e no contexto especificados no pedido. No entanto, é pouco provável que todos os recomendadores tenham obtido suas experiências ao mesmo tempo. Assim, verifica-se que as recomendações apresentam uma defasagem temporal do momento em que as experiências ocorreram até o momento em que o pedido de recomendação é feito. Essa defasagem pode ser utilizada para classificar as recomendações, dando uma maior ênfase para as mais novas, pois elas se aproximam mais do comportamento atual da entidade no contexto especificado. A defasagem temporal de uma recomendação é determinada por dt.

4.3.2 Determinação da Reputação

A reputação é determinada a partir das recomendações recebidas de outras entidades. Na sua forma mais simples, a reputação leva em consideração a defasagem temporal das recomendações. Através da determinação da média ponderada das recomendações com a defasagem temporal que possuem, a reputação pode ser calculada dando mais ênfase às recomendações mais recentes, pois se aproximam mais do comportamento atual da entidade. Considera-se um conjunto P que contém todos os recomendadores presentes no ambiente pervasivo, $p \in P$ um recomendador pertencente a este grupo e dt_p a defasagem temporal da recomendação proveniente da entidade p. O cálculo da reputação é apresentado na Equação 4.8

$$\eta_x^{y,l} = \frac{\sum_{p \in P} Rec(p, x, l) 1/dt_p}{1/dt_p} \tag{4.8}$$

A forma de determinação da reputação tem grande impacto na qualidade da reputação resultante. Caso não tenha sido corretamente especificada, pode levar a resultados incorretos. Por exemplo, observa-se que a reputação determinada a partir da Equação 4.8 não leva em consideração a possibilidade de recebimento de recomendações falsas, como exemplificado na Figura 4.3. Nos casos em que vários recomendadores estão presentes, o recebimento de uma recomendação falsa tem pouca influência na determinação da reputação. No entanto, um grupo de recomendadores é capaz de modificar completamente o valor da reputação. A forma de determinação da reputação deve ser capaz de lidar com recomendações falsas, reduzindo seu impacto na determinação da reputação. Para que isso seja possível, as recomendações devem ser ponderadas por um fator que determine a qualidade dos recomendadores, denominado credibilidade dos recomendadores.

Credibilidade dos recomendadores

A análise das entidades no que diz respeito à qualidade de suas recomendações é essencial à qualquer modelo de confiança que faça uso de recomendações. A credibilidade é uma métrica utilizada pelas entidades para avaliar a qualidade das recomendações recebidas. Esta métrica é utilizada para determinar quais entidades enviam recomendações falsas reduzindo o impacto destas no cálculo da reputação. Caso a credibilidade dos recomendadores não seja determinada corretamente, recomendações falsas provenientes de uma ou de um grupo de entidades podem afetar a determinação da reputação das entidades reduzindo a qualidade dos resultados obtidos.

A credibilidade de um recomendador pode ser determinada a partir de informações externas ou internas. Informações externas são as obtidas a partir de outras entidades. Pode-se, por exemplo, requisitar recomendações para avaliar especificamente os recomendadores. No entanto, isto torna o problema mais complexo. A alternativa consiste em determinar a credibilidade dos recomendadores a partir de informações internas, isto é, determinadas pela própria entidade avaliadora. Neste caso, uma opção consiste na reutilização da informação de confiança para determinar diretamente a credibilidade dos recomendadores. Essa opção se baseia na idéia de que entidades confiáveis do ponto de vista da disponibilização e/ou utilização de serviços também o são para emissão de recomendações. No entanto é possível que uma entidade mantenha um alto nível de confiança nas interações e envie recomendações falsas. Uma outra opção consiste em utilizar a similaridade entre as avaliações feitas pelas entidades como base para modelar a credibilidade. Nesta opção considera-se a diferença entre a avaliação feita pela própria entidade e a recebida através de recomendações como o fator que determina a qualidade da recomendação. Neste trabalho, a credibilidade é determinada a partir de um modelo de evolução baseado na similaridade das avaliações das entidades. Este modelo permite a evolução da credibilidade dos recomendadores de acordo com a qualidade de suas recomendações.

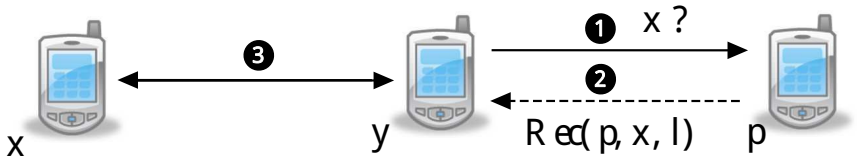


Figura 4.4: Diferença entre avaliação direta e de recomendação

Para se determinar a credibilidade de um recomendador deve-se determinar a qualidade das recomendações emitidas por ele. Essa qualidade pode ser inferida a partir da diferença entre a avaliação recebida através da recomendação e a determinada pela própria entidade. Quanto menor esta diferença, maior a similaridade entre as avaliações e maior a qualidade da recomendação. Um exemplo da determinação desta diferença pode observado na Figura 4.4. Nela observa-se que a entidade y requisita recomendações sobre a entidade x à entidade p (**1**). Para avaliar o recomendador p, y determina a diferença entre a recomendação recebida do mesmo

(**2**), dada por $Rec(p, x, l)$, e o valor de confiança dinâmico obtido por ele após interagir com x (**3**), dado por $\tau_x^{y,l}$. Essa diferença é apresentada na Equação 4.9.

$$q(p) = |Rec(p, x, l) - \tau_x^{y,l}| \tag{4.9}$$

A partir desta diferença, o modelo de credibilidade pode evoluir a cada nova recomendação para determinar a credibilidade do recomendador. O modelo de evolução da credibilidade assume a forma de uma função de atualização semelhante à utilizada para determinar o valor de confiança dinâmico, dado pela Equação 4.2. A partir deste modelo determina-se que a credibilidade de um recomendador deve ser aumentada quando a qualidade da recomendação for alta e diminuída quando for baixa. Utiliza-se a diferença determinada pela Equação 4.9 e um fator de tolerância (ε) para decidir entre aumentar ou diminuir o valor da credibilidade de um recomendador. Quando esta diferença é menor que o fator de tolerância (ε), isto é, as avaliações são semelhantes, a credibilidade é aumentada, caso contrário ela é diminuída. Quando um recomendador emite uma recomendação falsa, esteja ele participando ou não de uma colusão, esta recomendação é muito diferente do comportamento da entidade, possibilitando que o modelo de evolução diminua a credibilidade do recomendador. Na Equação 4.10 é apresentado o modelo de evolução da credibilidade.

$$Cred_x^y(i) = Cred_x^y(i-1) + \begin{cases} h^+(\cdot) & \text{se } q(p) \leq \varepsilon \\ h^-(\cdot) & \text{se } q(p) > \varepsilon \end{cases} \tag{4.10}$$

A dinâmica de evolução da credibilidade dos recomendadores é determinada a partir das funções h^+ e h^-. Esta dinâmica deve ser selecionada de modo que seja mais fácil para os recomendadores perder a credibilidade do que ganhar [36]. Para isso, determina-se que a taxa de crescimento da credibilidade, dada por, w_{cred} seja menor que $0,5$. A credibilidade é determinada pela Equação 4.11.

$$Cred_x^y(i) = Cred_x^y(i-1) + \begin{cases} w_{cred}(1 - Cred_x^y(i-1)) & \text{se } q(p) \leq \varepsilon \\ (w_{cred} - 1)Cred_x^y(i-1) & \text{se } q(p) > \varepsilon \end{cases} \tag{4.11}$$

Uma vez obtida a credibilidade dos recomendadores pode-se utilizá-la para determinar a reputação das entidades. Isto é feito através da seleção dos recomendadores que serão considerados e da ponderação das recomendações de acordo com sua credibilidade. Na Equação 4.12 é apresentado o cálculo da reputação baseado na Equação 4.8, mas levando em consideração a credibilidade dos recomendadores.

$$\eta_x^{y,l} = \frac{\sum_{p \in P'} Cred_p^y Rec(p, x, l) 1/dt_p}{\sum_{p \in P'} Cred_p^y 1/dt_p} \tag{4.12}$$

Onde P' consiste no subconjunto dos recomendadores selecionados de acordo com a credibilidade. Determina-se um limite mínimo para que a recomendação do recomendador seja considerada. Este subconjunto é dado por $P' = \{p \in P \mid Cred_p^y \geq LIM\_CRED\}$.

4.4 Valor de Confiança Geral

De acordo com a Definição 1, a confiança é proveniente de experiências diretas e/ou recomendações. A confiança proveniente de experiências diretas é dada pelo valor de confiança dinâmico (Definição 6). Já a confiança proveniente de recomendações é dada pela reputação e é determinada na Seção 4.3. Estas informações podem ser combinadas com o objetivo de construir uma métrica mais completa que leve em consideração estas duas fontes de informação. Esta combinação pode ser feita através de uma ponderação que determine qual o impacto de cada uma na formação da nova métrica. Portanto, a questão principal está em definir uma política de ponderação que determinaria o peso que cada informação deve assumir para que a combinação das informações seja feita com maior eficiência. Com este objetivo, a política de ponderação pode realizar a combinação das informações ou ignorar uma delas em detrimento da outra. Isto pode acontecer em casos em que uma das informações não está disponível. Yaniv *et al.* em [36] apresentada um estudo que analisa a política de ponderação entre informações provenientes de conhecimento próprio e provenientes de recomendações. O estudo mostra através de uma série de experimentos que, sem a noção da qualidade exata das recomendações, o peso para informações próprias é maior que $0,5$. Na Equação 4.13, apresenta-se o valor de confiança geral de uma entidade x sob o ponto de vista de y em um contexto l, onde se considera que o peso de informações próprias, dado por w_{dir}, é maior que $0,5$.

$$\Psi_x^{y,l} = w_{dir}\tau_x^{y,l} + (1 - w_{dir})\eta_x^{y,l} \tag{4.13}$$

4.5 Sumário

Neste capítulo foi apresentado o modelo de confiança desenvolvido para ambientes pervasivos. Este modelo determina a confiança dinâmica, a partir de interações diretas, e a reputação, a partir das recomendações trocadas entre as entidades. A partir destas informações, a confiança geral das relações é formada. Este modelo constitui a base para a implementação de um mecanismo de tomada de decisão baseada na confiança e é implementado pelo Gerenciador de Informação de Confiança apresentado no capítulo seguinte.

Capítulo 5

Gerenciador de Informação de Confiança

Neste capítulo é apresentado o Gerenciador de Informação de Confiança (GIC). Ele é responsável por implementar o modelo de confiança e, a partir de suas informações, implementar um mecanismo de tomada de decisão que se baseia em um limiar de cooperação. Neste capítulo apresenta-se ainda uma avaliação do GIC feita a partir de simulações cujos resultados são apresentados e analisados.

5.1 GIC

O GIC é responsável por determinar o valor de confiança das relações de confiança. Para isso, ele determina o valor de confiança das interações a partir das experiências obtidas, determinando a partir dele, a confiança dinâmica. Além disso, o GIC emite pedidos de recomendação, processando as respostas e determinando a reputação e a credibilidade das entidades. Outra tarefa que cabe ao GIC é implementar um mecanismo de cooperação que permita às aplicações tomar decisões de controle de acesso e seleção de provedores. O GIC é responsável ainda pela armazenagem de informações do modelo de confiança como histórico das interações, valor de confiança dinâmico, reputação, credibilidade dos recomendadores, etc.

O GIC pode ser dividido em três módulos lógicos, cada um responsável por obter e analisar um tipo de informação. Estes módulos são apresentados a seguir:

- Módulo de Análise Direta (MD) é responsável por obter e analisar informações de interações que ocorrem diretamente entre duas entidades. Ele mantém um histórico de interações associado ao contexto que é utilizado para determinar a evolução da confiança de uma relação de confiança.

- Módulo de Análise Indireta (MI) é responsável por obter e analisar informações obtidas a partir de outras entidades na forma de recomendações. A partir destas informações a reputação e a credibilidade das entidades são determinadas.

- Módulo de Cooperação (MC) tem o papel de determinar quando a cooperação entre as entidades nos ambientes pervasivos pode acontecer.

5.1.1 Módulo de Análise Direta

O Módulo de Análise Direta trata as informações provenientes de experiências diretas. Ele é responsável por determinar o valor de confiança das interações e, a partir dele, determinar o valor de confiança dinâmico. Na Figura 5.1 é apresentada a arquitetura do MD juntamente com o fluxo da informação de confiança. Este fluxo de informação é iniciado pela aplicação, que identifica os critérios de avaliação da interação em questão determinando o valor das métricas especificadas na Seção 3.3. Esta avaliação preliminar é enviada ao GIC através do MD. Ao receber o conjunto de critérios de avaliação, o MD determina o valor de confiança da interação. Essa informação é utilizada pelo modelo de evolução para determinar a confiança dinâmica da relação estabelecida entre as entidades participantes da interação. Por fim, as informações relacionadas a esta relação de confiança são armazenadas em uma base de dados de forma que possam ser recuperadas posteriormente. A seguir são fornecidos mais detalhes sobre a determinação do valor de confiança dinâmico e sobre a armazenagem das informações da relação de confiança.

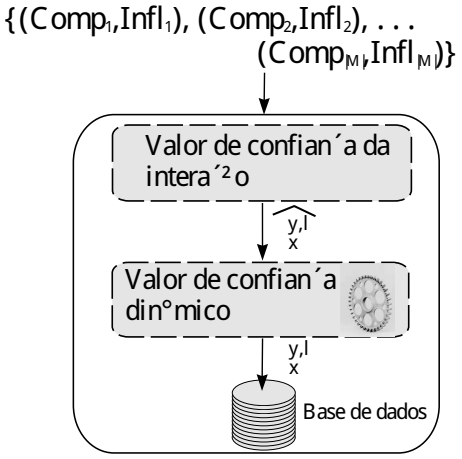


Figura 5.1: Módulo de Análise Direta

Computação do valor de confiança dinâmico

Um dos principais papéis da confiança dinâmica é verificar a consistência (Seção 4.2) do comportamento das entidades de modo a garantir que o nível de confiança obtido por cada uma delas reflita com corretamente o seu comportamento. Isto é obtido através da janela de observação do comportamento de tamanho W, que garante a elevação do nível de confiança apenas quando a entidade apresenta comportamento consistente. No entanto, o tamanho da janela de obser-

vações não precisa ser o mesmo para todas as relações de confiança. Por exemplo, entidades que não apresentem mudanças no comportamento podem ter uma janela menor. Já no caso de entidades que apresentem comportamento variável, mais interações são necessárias para permitir a elevação do valor de confiança dinâmico. Em virtude dessas características, a janela de observações pode ser adaptada ao comportamento que as entidades observadas apresentam na relação de confiança.

No Algoritmo 5.1 apresenta-se a determinação da confiança dinâmica da entidade x pela entidade y no contexto l, bem como a adaptação da janela de observações W. Neste algoritmo considera-se que já estão determinadas a diferença d e o histórico das interações, $hist$. A determinação do próximo valor da confiança dinâmico é feita a partir de d (**linha 2**), que seleciona a função de evolução a ser utilizada, g^+ ou g^-. O ajuste da janela de observação é feito da seguinte forma: quando é observada alguma redução no valor de confiança dinâmico, o tamanho da janela de observações é aumentado, o histórico das interações é anulado e um contador ($counter(x, y, l)$) recebe o valor correspondente ao tamanho da nova janela de observações (**linhas 6, 7 e 8**). A cada nova interação, o contador é decrementado (**linha 10**) e, ao se esgotar, o tamanho da janela de observações é reduzido e $counter(x, y, l)$ recebe o novo tamanho da janela (**linhas 12 e 13**). Caso uma nova redução no valor de confiança dinâmico seja detectada, o tamanho da janela é novamente aumentado. O processo segue até que um dos limites para o tamanho da janela seja alcançado.

Algoritmo 5.1 Computação da confiança

Entrada(s): $d, hist$

Saída(s): $\tau_x^{y,l}, hist$

1: ***Recupera*** $counter(x, y, l)$
2: **se** $d > 0$ **então**
3: $\quad \tau_x^{y,l} \Leftarrow g^+$
4: **senão**
5: $\quad \tau_x^{y,l} \Leftarrow g^-$
6: $\quad$ ***Aumenta*** $W(x, y, l)$
7: $\quad hist \Leftarrow 0$
8: $\quad counter(x, y, l) \Leftarrow W(x, y, l)$
9: **fim se**
10: ***Reduz*** $counter(x, y, l)$
11: **se** $counter(x, y, l) < 0$ **então**
12: $\quad$ ***Reduz*** $W(x, y, l)$
13: $\quad counter(x, y, l) \Leftarrow W(x, y, l)$
14: **fim se**

Armazenamento das Informações

Para que seja possível determinar a confiança dinâmica é necessário armazenar algumas informações. Estas informações estão relacionadas a relação de confiança e ao modelo de confiança. As principais informações são apresentadas na Figura 5.2. A partir dela observa-se que cada entidade analisada está associada a um ou mais contextos. A cada contexto estão associados o histórico de observações e o último valor determinado para a confiança dinâmica. Estas informações são utilizadas pelo modelo de confiança para determinar o próximo valor da confiança dinâmica. O campo marca temporal armazena o tempo em que ocorreu a última determinação do valor de confiança dinâmico. Este valor é utilizado para determinar a defasagem temporal sobre esta entidade, neste contexto, no momento da criação de uma recomendação.

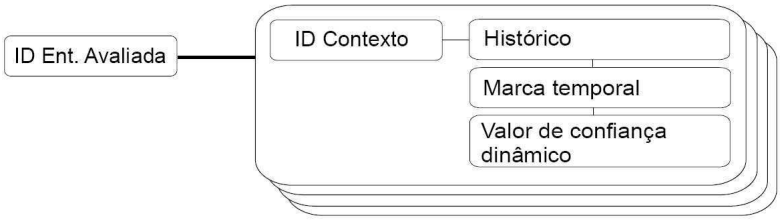


Figura 5.2: Principais informações armazenadas pelo MD

5.1.2 Módulo de Análise Indireta

O Módulo de Análise Indireta trata de informações provenientes de experiências indiretas obtidas através de recomendações. Ele é responsável por determinar a reputação e a credibilidade das entidades. Na Figura 5.3 é apresentada a arquitetura do MI. A partir desta Figura observa-se o recebimento de recomendações provenientes de várias entidades, $Rec(*, y, l)$, acerca de uma entidade qualquer y no contexto dado por l. A partir destas recomendações e da credibilidade de seus recomendadores, $Cred^z_*$, a reputação da entidade y no contexto l é determinada. A credibilidade, por sua vez, é determinada a partir das recomendações e do valor de confiança dinâmico. Estas informações são armazenadas de modo que possam ser utilizadas posteriormente.

A seguir são apresentados mais detalhes relacionados às recomendações como quando emitir pedidos de recomendações, para quais entidades estes pedidos devem ser enviados e que informações uma recomendação possui. Além disso são apresentados detalhes sobre a computação da credibilidade e sobre o armazenamento das informações.

Recomendações

A determinação da reputação de uma entidade em um contexto passa pela obtenção de recomendações sobre esta entidade. Em razão disto, a necessidade de emitir pedidos de recomendação

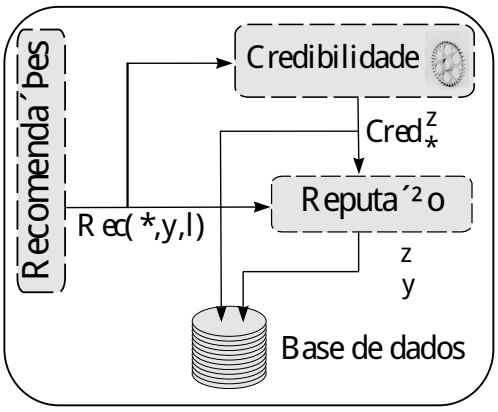


Figura 5.3: Módulo de Análise Indireta

depende diretamente da necessidade em determinar a reputação. A reputação, como determinado na Seção 4.4, forma juntamente com as experiências obtidas de forma direta, o valor de confiança de uma relação de confiança. Este valor deve ser determinado sempre que uma nova relação é estabelecida ou quando uma decisão precisa ser tomada. Assim, recomendações devem ser obtidas sempre que for necessário determinar o valor de confiança geral da relação de confiança. Deve-se determinar ainda para quais entidades os pedidos de recomendação devem ser emitidos para obter estas recomendações. Para isso, duas abordagens podem ser seguidas: a primeira considera que os pedidos de recomendação são emitidos apenas para um grupo específico de entidades; a segunda considera que os pedidos são enviados a todas as entidades disponíveis no ambiente pervasivo. Na primeira abordagem, o grupo de recomendadores pode ser selecionado de acordo com a credibilidade que possuem. No entanto, isto impede que novos recomendadores sejam adicionados. Em razão disto, a segunda abordagem é utilizada. Nela, novos recomendadores tem sua credibilidade avaliada permitindo que ela evolua até o ponto em que suas recomendações possam ser consideradas para determinação da reputação, como observado na Seção 4.3.2. Como os pedidos de recomendação são emitidos para todas as entidades disponíveis no ambiente, um intervalo de tempo mínimo entre pedidos de recomendação é considerado visando reduzir o impacto dos pedidos de recomendação na infra-estrutura de comunicação. As informações contidas em uma recomendação são apresentadas a seguir:

- **ID Recomendador**: identificador do emissor da recomendação
- **ID Ent. Avaliada**: identificador da entidade avaliada
- **ID Contexto**: identificador do contexto do valor de confiança
- **Defasagem temporal**: tempo decorrido desde a última atualização do valor de confiança dinâmico
- **Valor de confiança**: valor de confiança dinâmico determinado pelo recomendador

As informações sobre a entidade avaliada e o contexto são obtidos a partir do pedido de recomendação e o valor de confiança dinâmico é obtido a partir de uma consulta à base de dados do Módulo de Análise Direta.

Computação da credibilidade

A avaliação da credibilidade de um recomendador é feita com base nas recomendações recebidas e nas avaliações feitas pela própria entidade. A computação do modelo de evolução da credibilidade é descrito no Algoritmo 5.2. Este algoritmo determina a credibilidade de um grupo de recomendadores P que emitiu recomendações sobre uma entidade y no contexto l. Para isso, considera-se que estão disponíveis as recomendações e o valor de confiança dinâmico obtido pela entidade avaliadora ao interagir com y. O primeiro passo consiste em determinar a diferença entre as avaliações de um recomendador qualquer, j, e a obtida pela entidade avaliadora (**linha 2**). A partir desta diferença determina-se se a credibilidade do recomendador j deve aumentar ou diminuir (**linha 3**). Observa-se que todos os recomendadores tem sua credibilidade avaliada, permitindo que novos recomendadores possam ser adicionados.

Algoritmo 5.2 Computação da credibilidade

Entrada(s): $Rec(*, y, l)$, $\tau_y^{x,l}$

Saída(s): $Cred_*^x$

1: **para** $j = 1$ a P **faça**
2: $\quad q(j) = |Rec(j, y, l) - \tau_y^{x,l}|$
3: $\quad$ **se** $q(j) \leq \varepsilon$ **então**
4: $\quad\quad Cred_j^x \Leftarrow h^+$
5: $\quad$ **senão**
6: $\quad\quad Cred_j^x \Leftarrow h^-$
7: $\quad$ **fim se**
8: **fim para**

Armazenamento das informações

Algumas informações relacionadas ao modelo de evolução da credibilidade e à reputação necessitam ser armazenadas para serem recuperadas posteriormente. Estas informações são apresentadas na Figura 5.4. No caso do modelo, apenas a própria credibilidade necessita ser armazenada para possibilitar a evolução do modelo. A cada entidade avaliada que emita recomendações, um valor de credibilidade é associado. No caso da reputação, cada entidade avaliada pode ter valores de reputação associados a diferentes contextos. Ainda, a cada valor de reputação está associada o tempo no qual sua determinação foi realizada. Isto permite que pedidos de recomendação sejam feitos a partir de um certo intervalo de tempo a partir da última determinação

da reputação como observado na Seção 5.1.2.

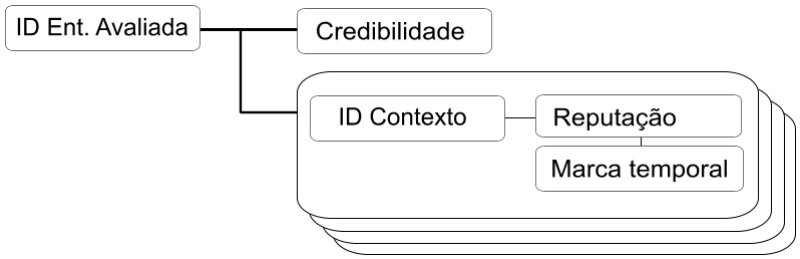


Figura 5.4: Informações armazenadas pelo MI

5.1.3 Módulo de Cooperação

Como observado na Seção 2.1.1, uma das principais características almejadas no contexto da computação pervasiva é a invisibilidade. O primeiro passo para alcançá-la consiste em minimizar a necessidade de interferência do usuário. Para que seja possível a entidade deve determinar o valor de confiança necessário para decidir se deve ou não cooperar em um determinado contexto. Esse nível de confiança normalmente é comparado com outro fator, o risco. Isto resulta em uma estratégia que determina um limiar de cooperação utilizada em alguns outros trabalhos [34], [5], [28] e [19].

O Módulo de Cooperação determina um limiar de cooperação que pode ser utilizado como base para tomada de decisão por parte das aplicações. Este limiar é determinado a partir de uma análise de risco e da confiança das relações. Para isso, o *nível de risco* em cada contexto é comparado com o *nível de cooperação*, determinado a partir do valor de confiança, para determinar se a cooperação entre as entidades deve ocorrer. Este limiar de cooperação pode ser utilizado para selecionar provedores ou clientes de serviço.

Risco

Risco pode ser compreendido como o produto da probabilidade de ocorrência de um evento não desejado e da conseqüência deste evento [6] e [15]. Hussain *et al.*, como descrito em [14], utiliza o contexto para classificar as interações, também o considerando para modelar o risco. A partir disto, considera-se que risco pode ser determinado a cada interação, levando em consideração o contexto, a probabilidade de ocorrência de um evento não desejado e a conseqüência da ocorrência deste evento.

Neste trabalho considera-se um modelo simples para determinar o *nível de risco* em uma interação. Este modelo considera que a probabilidade de ocorrência de um evento não desejado pode ser obtida a partir da análise do comportamento passado da entidade. Determina-se que evento não desejado é aquele que leva à uma redução do valor de confiança dinâmico. A probabilidade pode então, ser obtida a partir da determinação da freqüência de ocorrência deste tipo

de evento na janela de observações. Considera-se ainda que a conseqüência de um evento não desejado é dada pela utilidade do contexto da relação. Quanto maior a utilidade de um contexto, maior a conseqüência associada a uma diminuição do valor de confiança dinâmico. O *nível de risco* é dado pela Equação 5.1.

$$\textit{nível de risco} = U^l \times fn \tag{5.1}$$

Onde, fn determina a freqüência de ocorrências de redução do valor de confiança dinâmico na janela de observações.

Limiar de cooperação

Uma vez determinados os fatores que influenciam o risco e os valores de confiança de uma entidade, pode-se determinar quando a mesma deve ou não cooperar em uma situação. Pode-se utilizar a confiança para decidir sobre a aceitação de pedido de um cliente, ou a escolha de um provedor de serviço, protegendo desta forma os recursos da entidade, não os fornecendo a outras que não correspondam às expectativas. Para isso, faz-se uso da abordagem do limiar de cooperação onde os fatores que influenciam o risco em uma situação são comparados aos fatores que influenciam a cooperação. Quando os fatores de cooperação pesam mais que os fatores de risco, a cooperação pode ocorrer. Apesar de ser uma abordagem simples, ela captura a idéia de cooperação e pode ser considerada uma ferramenta importante para auxiliar a entidade na tomada de decisão baseada na confiança. A regra geral para a tomada de decisão é dada pela Equação 5.2

$$\textit{nível de cooperação} \geq \textit{nível de risco} \tag{5.2}$$

O *nível de cooperação* depende do valor de confiança da relação e do Fator de Intenção. Este fator, dado por FI, é subjetivo, devendo ser determinado pelo usuário. Através deste fator, o *nível de cooperação* da entidade, que depende da confiança, pode aumentar aumentando o *nível de risco* aceitável pela entidade. Determina-se que $0 < FI < 0,5$ e que o *nível de cooperação* é baseado no valor de confiança geral da relação de confiança. Assim, o *nível de cooperação* é determinado pela Equação 5.3.

$$\textit{nível de cooperação} = \Psi_x^{y,l} \times (1 + FI) \tag{5.3}$$

Com isso, o limiar de cooperação é determinado a partir das Equações 5.1 e 5.3. A Equação 5.4 determina o limiar de cooperação que deve ser seguido pelas entidades para tomar decisões acerca de cooperação com entidades em ambientes pervasivos.

$$\Psi_x^{y,l} \times (1 + FI) \geq U^l(1 + fn) \tag{5.4}$$

Nota-se que o limiar de cooperação não realiza a seleção da entidade com a qual se deve cooperar, ele determina o limite mínimo necessário para que a cooperação ocorra. Na presença de várias entidades disponibilizando o mesmo serviço no ambiente pervasivo, por exemplo, outro critério pode ser utilizado para determinar especificamente com qual entidade a cooperação deve ocorrer. É importante notar que este limiar de cooperação representa uma sugestão de como a decisão de cooperação baseada na confiança pode ser tomada. Outros mecanismos podem ser adotados pelas entidades.

Computação do limiar de cooperação

A tomada de decisão fazendo uso do limiar de cooperação leva em consideração a confiança e o risco determinado para cada entidade. No entanto, sistemas que utilizam confiança para realizar a tomada de decisão são afetados por um problema denominado *início frio*. Este problema, identificado por alguns autores incluindo [27] e [22], consiste na dificuldade que entidades desconhecidas encontram em se comunicar. Isto é decorrente do fato de novas entidades, por não terem se comunicado em nenhuma situação, não possuírem confiança de nenhuma outra entidade no ambiente pervasivo. Desta forma, a inexistência de experiências impede que recomendações sejam emitidas sobre . Logo, entidades desconhecidas não são confiáveis por não terem se comunicado previamente e não o fazem por não possuírem confiança. Este problema é estudado com profundidade em [26], no entanto com a consideração de um repositório central de recomendações o que não se encaixa no contexto deste trabalho.

Para solucionar este problema deve-se determinar os estados possíveis que a relação de confiança entre entidades pode assumir. Eles são apresentados a seguir:

1. Desconhecimento completo. A entidade que se apresenta é completamente desconhecida de todas as outras entidades do ambiente pervasivo.

2. Conhecimento parcial. A entidade que se apresenta já interagiu com alguma outra em outros contextos que não o do foco da tomada de decisão a qual está sujeita.

3. Conhecimento completo. A entidade já é conhecida no contexto alvo da tomada de decisão.

A partir destes estados, a estratégia que se adota para solucionar este problema consiste na construção gradual da confiança. Para que isto seja possível, informações em contextos semelhantes e o Fator de Intenção (FI) são utilizados.

No caso de desconhecimento completo (**Estado 1**), o Fator de Intenção é utilizado para decidir os contextos nos quais as interações são permitidas. Como não há nenhum conhecimento sobre a entidade e seu comportamento e como $0 < FI < 0,5$, é razoável propor que apenas

serviços de utilidade menor que FI sejam permitidos quando a relação de confiança se encontra neste estado.

No caso de conhecimento parcial ou incompleto (**Estado 2**), existe alguma informação sobre a entidade em questão, caso contrário estar-se-ia no **Estado 1**. Esta informação é incompleta, pois não se trata do contexto alvo da decisão (**Estado 3**). Assim, esta informação que está associada a outro contexto, é utilizada para decidir se a cooperação deve ou não acontecer, segundo a Equação 5.4.

Quando há conhecimento completo sobre a entidade, a Equação 5.4 é utilizada diretamente para determinar a decisão a ser tomada.

O processo de computação da decisão é apresentado no Algoritmo 5.3. Neste algoritmo, observa-se a tomada de decisão de uma entidade x acerca da entidade y no contexto l. O algoritmo começa pela verificação se x já interagiu com y no contexto l (**linha 1**). Caso isso não tenha acontecido e caso $U^l < FI$, determina-se que: o valor de confiança dinâmico inicial é igual a FI (**linha 2**) e que cooperação pode acontecer (**linha 3**). Caso x não tenha interagido com y no contexto l ou a utilidade do contexto l seja maior que o Fator de Intenção, o valor de confiança geral no contexto l é requisitado (**linha 5**). Caso não seja possível computá-lo (**linha 6**), o valor de confiança geral em um contexto similar é requisitado (**linha 7**). Por contexto similar entende-se o contexto da entidade atuando com o mesmo papel especificado no contexto l. Assim, são utilizadas o valor de confiança da entidade y como cliente ou provedor, de acordo com as Definições 7 e 8. Quando a confiança geral no contexto similar não pode ser determinada (**linha 8**), não há nenhuma informação que possa ser utilizada no limiar de cooperação, impedindo que a cooperação ocorra (**linha 9**). Caso algum dos valores de confiança geral possa ser determinado (Ψ), ele é utilizado na Equação 5.4 para tomar a decisão (**linha 12**).

5.2 Avaliação

Nesta Seção apresenta-se uma avaliação do Gerenciador de Informação de Confiança feita através de simulações. São avaliados o desempenho da determinação do valor de confiança dinâmico, da credibilidade, da reputação e do limiar de cooperação. Além disso, duas novas implementações para a determinação da reputação são apresentadas e comparadas à implementação proposta na Seção 4.3.2.

5.2.1 Opções de implementação

A reputação é um aspecto importante na análise da confiança. Em virtude disto, são apresentadas duas opções de implementação que se diferenciam da previamente apresentada pelo modo de determinação da credibilidade e da reputação. A principal característica destas im-

Algoritmo 5.3 Computação do limiar de cooperação

Entrada(s): U^l, FI

Saída(s): $Cooperate$

se $\nexists\ \tau_y^{x,l}$ e $U^l < FI$ **então**
  $\tau_y^{x,l} \Leftarrow FI$
  $Cooperate = true$
fim se
Requisita $\Psi_y^{x,l}$
se $\nexists\ \Psi_y^{x,l}$ **então**
  Requisita $\Psi_y^{x,contexto\ similar}$
  se $\nexists\ \Psi_y^{x,contexto\ similar}$ **então**
    $Cooperate = false$
  fim se
fim se
$Cooperate \Leftarrow$ ***Computa*** Equação 5.4 com Ψ_y^x

plementações está em não possuir um sistema dedicado para determinar a credibilidade dos recomendadores, como o apresentado na Seção 4.3.2. Ao invés disso, reutilizam informações de outras fontes para determinar esta medida, sendo assim mais simples que a implementação apresentada.

A primeira opção consiste na reutilização da informação de confiança para determinar a credibilidade dos recomendadores. Com ela, as recomendações das entidades são ponderadas diretamente pelo valor de confiança da relação de confiança no contexto da recomendação. Esta forma de utilização da confiança já foi objeto de outros estudos como [3] e [34]. Ela se baseia na consideração de que é mais provável que entidades não confiáveis emitam recomendações falsas e entidades confiáveis emitam recomendações verdadeiras. A determinação da reputação das entidades baseada na confiança dinâmica é feita seguindo a Equação 5.5.

$$\underset{conf}{\eta_x^{y,l}} = \frac{\sum_{p \in P} \tau_p^{y,l} \times Rec(p,x,l) \times 1/dt_p}{\sum_{p \in P} \tau_p^{y,l} \times 1/dt_p} \tag{5.5}$$

Onde $\underset{conf}{\eta_x^{y,l}}$ é utilizado para diferenciar esta forma de determinação da reputação da previamente apresentada, que era baseada na credibilidade. A reputação baseada na credibilidade será referenciada pelo resto do texto como $\underset{cred}{\eta_x^{y,l}}$.

A segunda opção é mais simples e considera que todas as entidades emitem recomendações verdadeiras. A reputação de cada uma delas em um determinado contexto é dada pela maioria dos votos, isto é, o valor de recomendação da maioria. Para determinar a opinião da maioria utiliza-se a mediana das recomendações que consiste na recomendação que separa a metade

mais alta das recomendações da metade mais baixa passando uma idéia da tendência central das recomendações. A determinação da reputação de cada entidade utilizando esta opção é apresentada pela Equação 5.6.

$$\eta_{x\ med}^{y,l} = \underset{p \in P}{mediana}\ Rec(p, x, l) \tag{5.6}$$

5.2.2 Simulação

A simulação foi implementada utilizando o *framework* OverSim [4] para o ambiente de simulação OMNeT++ [30]. Neste ambiente, as entidades se comunicam diretamente umas com as outras através de troca de mensagens o que possibilitou a simulação da utilização de serviços anunciados pelas entidades bem como a troca de informações de recomendações. As simulações são implementadas em C/C++ possibilitando que a implementação desenvolvida para a simulação pudesse ser considerada a implementação de referência do GIC. Na Seção 6.1 a interface desta implementação de referência é apresentada.

Critérios de avaliação e modelo de ataque

O GIC é avaliado segundo três critérios. O primeiro trata da avaliação da qualidade da reputação na presença de comportamento desonesto das entidades. O segundo trata da avaliação da qualidade da decisão feita através da utilização limiar de cooperação na presença de entidades desonestas. O terceiro trata da avaliação da eficiência do sistema ao lidar com variação do comportamento das entidades.

O comportamento das entidades presentes em cada um destes experimentos é classificado como honesto ou desonesto. No primeiro, as entidades seguem os critérios de avaliação especificados gerando valores elevados para a confiança das interações, além de fornecerem somente recomendações verdadeiras. No segundo, as entidades fornecem recomendações falsas e não seguem os critérios de avaliação especificados. Arbitra-se que o valor de confiança das interações entre as entidades no ambiente de simulação é igual a $1,0$ para entidades honestas e igual a $U/2$ para entidades desonestas. O percentual de entidades que agem desonestamente é determinado por PED.

Além disso, as entidades desonestas não necessariamente agem desta forma em todas as interações. Elas podem agir corretamente com uma certa freqüência com objetivo de enganar o sistema. Este tipo de variação no comportamento é determinado pela Taxa de Comportamento Desonesto TCD que determina a freqüência com que as entidades agem desonestamente.

É possível ainda que as entidades formem grupos (colusões) para agir contra outras entidades. Nestes casos, entidades fornecem recomendações elevadas para as que fazem parte do grupo e recomendações baixas para as que não fazem parte, aumentando a reputação das enti-

dades do grupo e reduzindo a reputação para entidades fora do grupo. Este critério também é utilizado para avaliar o GIC.

Métricas de avaliação

São utilizadas duas métricas na avaliação: o Erro na Computação da Confiança (ECC) e a Taxa de Sucesso de Interações (TSI). Onde a primeira é utilizada para avaliar o desempenho da determinação da reputação e a segunda métrica é utilizada para determinar a eficiência do limiar de cooperação.

Para se obter ECC determina-se a diferença entre a reputação da entidade e um valor de referência da reputação que depende do tipo da entidade (honesta ou desonesta) e é igual ao valor de confiança da interação no ambiente de simulação determinada anteriormente. O valor de ECC é então, definido como o valor RMS (*Root mean square*) da diferença entre o valor da reputação e seu valor de referência.

A Taxa de Sucesso de Interação (TSI) é determinada pela razão entre a quantidade de interações bem sucedidas pelo número total de interações. Uma interação é considerada bem sucedida quando nenhuma das partes age de maneira desonesta, isto é, quando a avaliação dos critérios é sempre maior que o valor de utilidade do serviço.

São utilizadas duas formas de interações. Na primeira a escolha das entidades é aleatória, isto é, não há restrição sobre quais entidades se comunicam. Esta forma é utilizada na avaliação da reputação e da confiança dinâmica. Na segunda, o limiar de cooperação é utilizado para selecionar as entidades com as quais a interações ocorrem. Esta é utilizada na avaliação da eficiência do limiar de cooperação em bloquear a comunicação com entidades desonestas.

Parâmetros da simulação

A simulação foi feita utilizando 10 entidades. A taxa padrão de entidades desonestas é fixada em 30%. Além disso, as entidades desonestas podem estar agir em grupo ou não. Caso, não seja explicitado, as mesmas não estarão agindo em grupo. Estes e outros parâmetros são apresentados na Tabela 5.1. Cada uma das entidades disponibiliza 5 serviços com níveis de utilidade diferentes que variam de 0.1 a 0.9. A variação nas recomendações emitidas pelas entidades desonestas é determinada pelo parâmetro Δrec.

5.2.3 Resultados

A seguir são apresentados os resultados das simulações realizadas utilizando os critérios de avaliação definidos previamente. Considera-se que todas as entidades são desconhecidas no início da simulação. Em razão disto, um período de transição é necessário para que o GIC

Parâmetro	Descrição	Valor padrão
N	Número de entidades	10
W	Janela de observações	$10 < W < 60$
w_{dir}	Influência da informações próprias	$0,60$
w_p	Influência de ações positivas	$0,15$
w_n	Influência das ações negativas	$0,40$
FI	Fator de Intenção	$0,35$
ε	Fator de tolerância	15%
LIM_CRED	Credibilidade mínima	$0,9$
Δrec	Variação na recomendação	$0,3$

Tabela 5.1: Parâmetros da simulação

ajuste os valores de confiança. Os resultados foram obtidos após a estabilização dos valores de confiança.

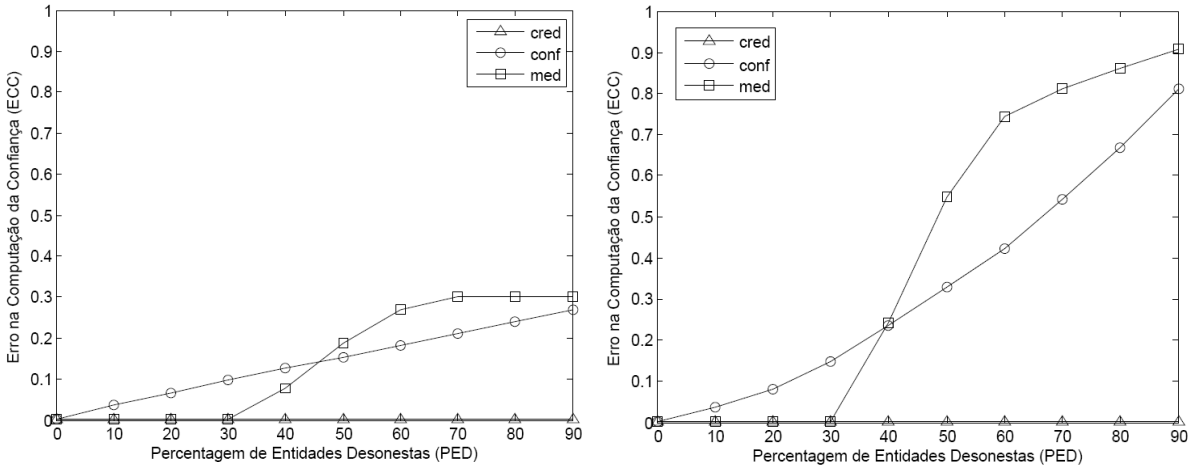
Qualidade da determinação da reputação

Este conjunto de experimentos tem por objetivo mostrar a qualidade da determinação da reputação na presença de entidades desonestas fornecendo recomendações falsas. São avaliadas as três implementações apresentadas para determinação da reputação: credibilidade (*cred*), confiança (*conf*) e mediana (*med*).

Para avaliar essas implementações, realizam-se duas simulações a partir das quais se obtêm ECC. A primeira é obtida variando-se o percentual de entidades desonestas PED de 0 a 90%, e a segunda a partir da variação da TCD de 0 a 100%. Através da primeira simulação é possível observar o desempenho das implementações em lidar com um número crescente de entidades desonestas que fornecem recomendações falsas. Através da segunda simulação, pode-se observar o desempenho com a variação do comportamento das entidades. Ambos os experimentos são executado ainda com as entidades desonestas trabalhando em grupo. Com isto, pode-se verificar o impacto da colusão de entidades desonestas na determinação da reputação.

Na Figura 5.5 é apresentado o desempenho das diferentes implementações da reputação ($cred$, $conf$ e med) com a variação do percentual de entidades desonestas. Na Figura 5.5(a) é apresentada a simulação sem colusão. Pode-se observar que o erro na computação da confiança, para o sistema de reputação que utiliza a credibilidade das entidades, é sempre baixo e independente do percentual de entidades desonestas presentes. Isso mostra a eficiência da avaliação dos recomendadores em rejeitar recomendações falsas. A implementação que utiliza a confiança como credibilidade apresenta um crescimento praticamente constante do erro com o aumento de PED, chegando a valores próximos de 30%. A implementação que utiliza a mediana apre-

senta bons resultados quando a quantidade de entidades desonestas é baixa. Isso acontece, pois esta implementação leva em consideração a tendência central das recomendações, i.e., quando a maioria dos recomendadores é composta por entidades honestas, esta implementação obtém bons resultados. O desempenho desta implementação piora quando o número de entidades desonesta aumenta.

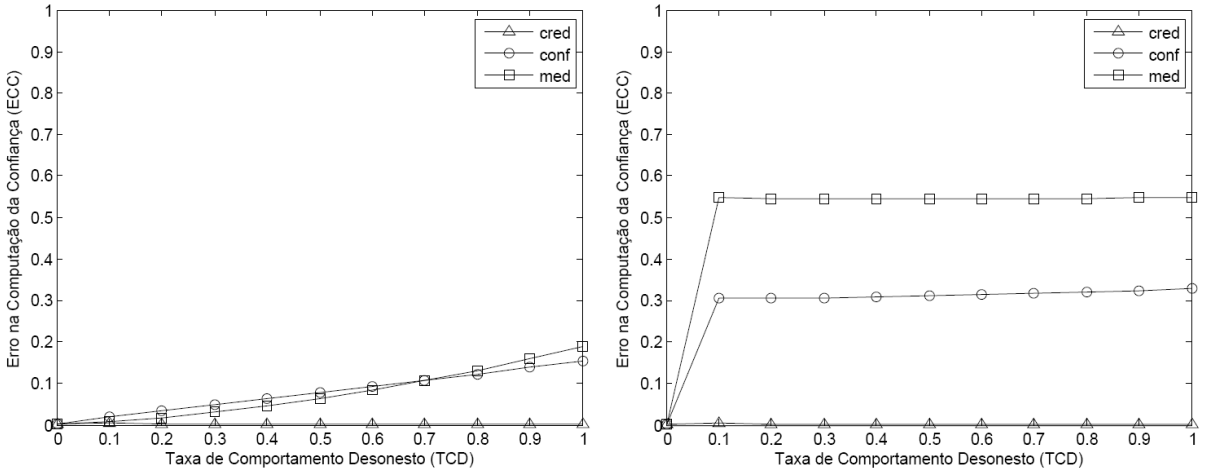


(a) Sem colusão (b) Com colusão

Figura 5.5: Erro na Computação da Confiança com a variação do percentual de entidades desonestas

Na Figura 5.5(b) são apresentados os resultados obtidos quando as entidades desonestas agem em grupo. Observa-se que o crescimento do erro com o aumento do PED para a implementação $conf$ já não é mais constante como visto anteriormente. Os efeitos da colusão das entidades desonestas podem ser vistos pelo aumento não linear e mais acentuado do erro percentual com o aumento do PED. Isto é, o aumento no poder do grupo de entidades desonestas, através do aumento da quantidade de entidades, resulta em resultados piores para esta implementação. No caso da implementação med, os resultados também são piores que o caso sem colusão, apresentando os piores resultados dentre todas as implementações nesta configuração. Observa-se que no caso da implementação $cred$ o percentual de erro mantém-se pequeno mesmo com percentuais elevados de entidades desonestas trabalhando em grupo.

Na Figura 5.6 ilustra-se a evolução do ECC com a variação da Taxa de Comportamento Desonesto (TCD) para as configurações com e sem colusão. Esta simulação foi obtida com 50% de entidades desonestas no ambiente pervasivo. Observa-se a partir da Figura 5.6(a) que as implementações $conf$ e med apresentam resultados similares na configuração sem colusão, com um crescimento pequeno do erro com aumento da TCD. No caso com colusão, ilustrado na Figura 5.6(b), os resultados obtidos pelas implementações med e $cred$ foram piores do que na configuração sem colusão. Observa-se ainda, que neste caso, os erro variou pouco com a variação de TCD, sendo praticamente igual ao seu valor máximo, obtido com TCD em 100%.



(a) Sem colusão (b) Com colusão

Figura 5.6: Erro na Computação da Confiança com a variação da Taxa de Comportamento Desonesto.

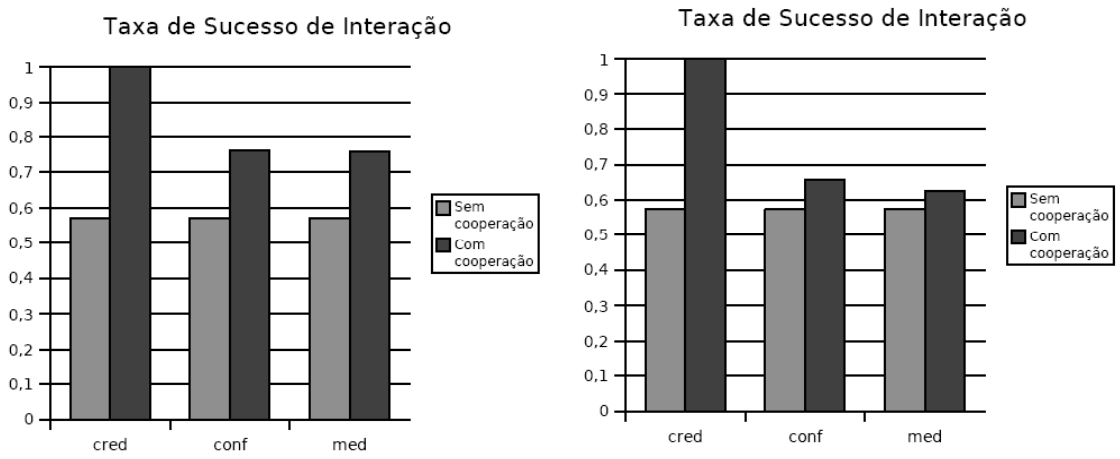
Os resultados da implementação $cred$ foram melhores para ambos os casos, com e sem colusão, mostrando que mesmo com variação do comportamento das entidades, esta implementação é capaz de evitar reduzir o impacto das recomendações falsas.

Eficiência do limiar de cooperação

Os experimentos realizados para verificar a eficiência da utilização do limiar de cooperação em limitar o acesso de entidades desonestas consistem em analisar a TSI com e sem a utilização do limiar de cooperação com todas as implementações. Os resultados foram ainda comparados com a configuração com colusão. Para este experimento, o percentual de entidades desonestas foi ajustado para 50%.

Na Figura 5.7 são apresentados os resultados obtidos na utilização do limiar de cooperação. Na Figura 5.7(a), as três implementações com e sem a utilização do limiar de cooperação são comparadas e seus efeitos na Taxa de Sucesso das Interações são observados. Observa-se que houve um ganho de desempenho pela utilização do limiar de cooperação em todas as implementações. Confirma-se assim, a eficiência do limiar de cooperação em evitar cooperação com entidades desonestas. O melhor resultado coube à implementação $cred$ devido ao seu bom desempenho em determinar a confiança das entidades. As outras implementações obtiveram rendimento inferior com relação a Taxa de Sucesso de Interações em virtude de seu desempenho também inferior na determinação da confiança.

Na Figura 5.7(b) são apresentados os resultados em uma configuração com colusão. Assim como no caso anterior, a TSI da implementação $cred$ é bastante elevada o que mostra o sucesso da combinação desta implementação e do limiar de cooperação em evitar entidades desonestas mesmo em situações em que há colusão. Observa-se que as outras implementações tem re-



(a) Sem colusão (b) Com colusão

Figura 5.7: Taxa de Sucesso das Interações com a utilização da cooperação

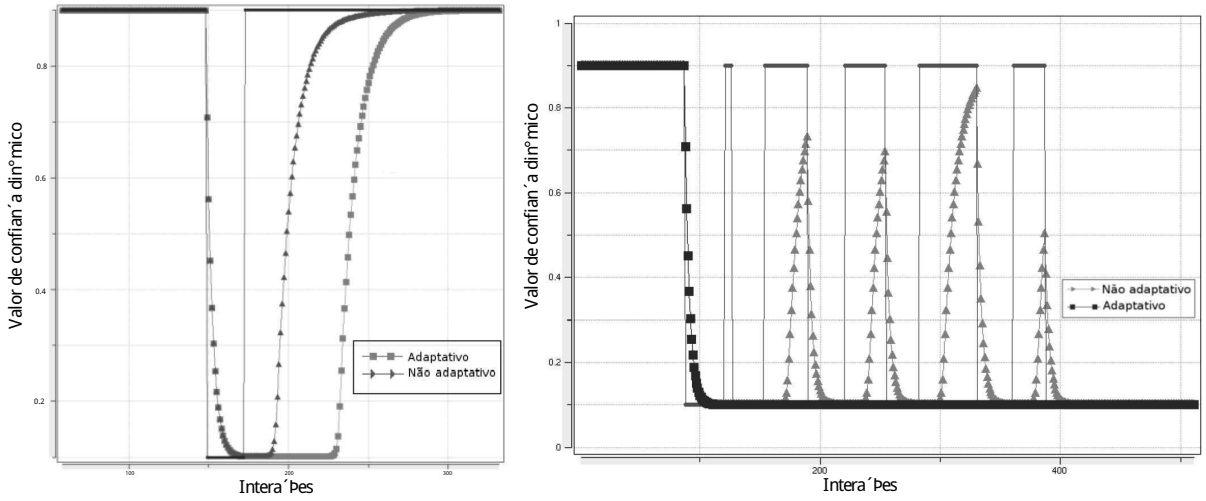
sultados inferiores nesta configuração em relação à configuração sem colusão, devido ao seus respectivos desempenhos na determinação da confiança das entidades.

Eficiência contra comportamento variável

Este experimento apresenta o desempenho do GIC em lidar com comportamento variável das entidades. Por comportamento variável entende-se entidades que hora cumprem os critérios definidos para o contexto hora não seguem. Esta variação deve se refletir na confiança da entidade. São apresentados os resultados para duas implementações do modelo de evolução da confiança: uma que utiliza um esquema adaptativo de ajuste de janela e outra que não utiliza.

A partir do resultado apresentado na Figura 5.8(a), observa-se a evolução da confiança dinâmica de uma entidade que apresenta este tipo de comportamento. Nesta figura observa-se o valor da confiança das interações e seu impacto na confiança dinâmica para as implementações não adaptativa e adaptativa da janela de observações. Quando ocorre a queda do valor de confiança das interações, a confiança dinâmica determinada pelas duas implementações, acompanha rapidamente este valor. Quando, logo em seguida, a entidade tenta recuperar a confiança, a implementação adaptativa necessita de mais interações para recuperar o valor de confiança do que a implementação não adaptativa. Isso se deve à redução do valor de confiança que levou o algoritmo de adaptação a aumentar o tamanho da janela.

Na Figura 5.8(b) ilustra-se um cenário em que a entidade desonesta apresenta comportamento variável. Observa-se que com a implementação não adaptativa, a entidade desonesta é capaz de recuperar parte do valor de confiança perdido para logo em seguida, agir desonestamente novamente. No caso da implementação adaptativa isto não é possível, pois o algoritmo de adaptação da janela aumenta o tamanho da mesma impedindo que a entidade aja honestamente por um curto período de tempo e consiga recuperar o valor de confiança perdido em seguida.



(a) Perda e recuperação da confiança (b) Comportamento variável

Figura 5.8: Desempenho com comportamento variável

5.2.4 Análise

Como pôde ser observado a partir dos resultados, a implementação para determinação da reputação que apresentou os melhores resultados adota um modelo dedicado para a avaliação da credibilidade dos recomendadores. A utilização da implementação *conf* apresenta resultados que podem ser considerados razoáveis quando o percentual de entidades desonestas é baixo (menor que 25%) com e sem colusão, como pode ser observado na Figura 5.5. O mesmo ocorre para a implementação *med* que apresenta resultados comparáveis aos obtidos com a implementação *cred* quando o percentual de entidades desonestas é inferior a 30%. No entanto, devido à natureza dinâmica dos ambientes pervasivos, percentuais elevados de entidades desonestas podem ser obtidos facilmente. Um ambiente com apenas dois recomendadores, por exemplo, e com apenas um deles emitindo recomendações falsas, eleva o PED para 50% o que reduz significativamente o desempenho das implementações *conf* e *med*. Caso apenas um recomendador esteja presente e caso este seja desonesto, o desempenho é reduzido ainda mais. Como pode ser observado a partir dos resultados, isso não ocorre com a implementação *cred*. Conclui-se assim, que as implementações *conf* e *med* apresentam desempenho aceitável para percentuais de entidades desonestas baixos, no entanto, isso não é suficiente para que elas sejam consideradas adequadas para ambientes pervasivos, onde estes percentuais podem ser bastante elevadas.

A partir das simulações que avaliaram o desempenho do limiar de cooperação pôde-se observar que o mesmo propiciou aumento na Taxa de Sucesso das Interações considerando as três implementações nas configurações com e sem colusão. Isso significa que o limiar de cooperação foi capaz de reduzir a cooperação com entidades desonestas. A partir da Figura 5.7 observa-se que os melhores resultados foram obtidos utilizando a implementação *cred*. Conclui-se que a utilização do limiar de cooperação com uma implementação que avalie a credibilidade das

entidades é capaz de evitar a cooperação com entidades desonestas em ambientes pervasivos mesmo em configurações com colusões.

Entidades em ambientes pervasivos podem agir dinamicamente variando seu comportamento. As entidades devem ser capazes de determinar corretamente o valor de confiança dinâmico das entidades. Observa-se pelos resultados que a utilização do modelo de evolução da confiança não é suficiente lidar com as variações no comportamento das entidades. A utilização do algoritmo de adaptação da janela de observações se mostrou eficiente na detecção deste tipo de comportamento não permitindo que as variações no comportamento das entidades afetassem o valor de confiança dinâmico indevidamente. Assim, se faz necessário que o modelo de evolução da confiança seja utilizado juntamente com o algoritmo de adaptação da janela de observações para que a determinação do valor de confiança dinâmico não seja suscetível a variações no comportamento das entidades.

5.3 Sumário

Neste capítulo o GIC foi descrito de acordo com seus módulos lógicos: análise direta, análise indireta e cooperação. O modo de funcionamento de cada um destes módulos foi apresentado através dos algoritmos utilizados para computar a confiança dinâmica, a reputação, a credibilidade e o limiar de cooperação. Apresentou-se ainda uma avaliação do GIC feita através de simulações cujos resultados mostraram sua eficiência em lidar com comportamento variável das entidades, recomendações falsas e ações em colusão.

Capítulo 6

Implementação do GIC e integração ao *middleware Wings*

Neste Capítulo é apresentada a implementação do GIC, dando ênfase para seus módulos que definem as interfaces que dão acesso às suas funções. Apresenta-se ainda a integração desta implementação ao *middleware Wings*. Para isso, uma visão geral sobre o mesmo é apresentada bem como os serviços criados e adicionados a ele para dar suporte a análise da confiança. A descrição destes serviços é feita e uma aplicação de teste é apresentada.

6.1 Implementação do GIC

A implementação do GIC foi feita utilizando a linguagem C e foi dividida em cinco módulos: comunicação, experiência, reputação, cooperação e configuração. Cada um destes módulos possui uma interface. Na Figura 6.1 apresenta-se a implementação do GIC e sua relação com a camada superior e com a camada de comunicação às quais o GIC está associado. Observa-se que a camada superior, que pode ser uma aplicação acessando o GIC diretamente ou através de um *middleware*, tem acesso aos módulos de experiência, cooperação e reputação. A camada de comunicação é acessada pelo módulo de comunicação e é utilizada para emitir pedidos de recomendação. O módulo de cooperação acessa o módulo de experiência e o módulo de reputação para avaliar o limiar de cooperação. A interface de cada um destes módulos é descrita a seguir.

6.1.1 Experiências

As experiências obtidas pelas aplicações a cada interação são utilizadas pelo GIC para determinar a confiança dinâmica da relação de confiança. O módulo de experiências é responsável por implementar o modelo de evolução da confiança e por prover uma interface para que as aplicações possam atualizar as experiências do GIC.

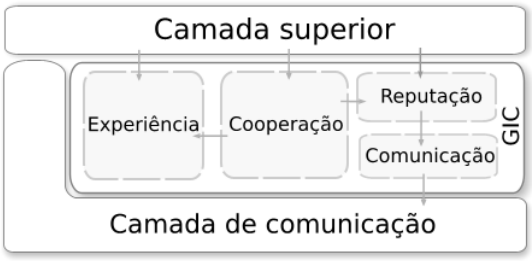


Figura 6.1: Implementação do GIC

As experiências são determinadas a partir dos critérios definidos para cada contexto, como descrito na Seção 3.3. Desta forma, a cada interação, o valor das métricas de cada critério é determinado e o valor de confiança da interação é obtido. O GIC possui dois tipos de funções para atualizar o valor de confiança dinâmico. O primeiro tipo ($GIC\_API\_update\_xp$) as atualiza diretamente, recebendo o valor de confiança da interação. O segundo tipo ($GIC\_API\_estimate\_xp$) recebe apenas os valores das métricas para cada critério. Internamente, esta função determina o valor de confiança da interação e atualiza o valor de confiança dinâmico. Além destas, o módulo de experiências disponibiliza funções de acesso ao valor de confiança dinâmico ($GIC\_API\_get\_dinTrustValue$), que retornam uma `struct` denominada $DinTrustToken$, que contém além do valor de confiança dinâmico, o tempo decorrido desde a ocorrência da última experiência. Na listagem de código 6.1 são apresentadas todas as funções relacionadas ao módulo de experiências. Observa-se que todas as funções possuem um sufixo diferenciador P e C cujo propósito é diferenciar o papel desempenhado por entidades provedoras do papel desempenhado por entidades clientes de serviço.

Código 6.1: Funções relacionadas a experiência

```
int GIC_API_update_xpC(const char * sourceKey, const char* ctxID,
                                    double estTrustValue);
int GIC_API_update_xpP(const char * sourceKey, const char* ctxID,
                                    double estTrustValue);
void GIC_API_estimate_xpC(const char * sourceKey,
                const char * ctxID, double comp, double infl,... );
void GIC_API_estimate_xpP(const char * sourceKey,
                const char * ctxID, double comp, double infl,... );
DinTrustToken GIC_API_get_dinTrustValueC(const char * key,
                                    const char* ctxID);
DinTrustToken GIC_API_get_dinTrustValueP(const char * key,
                                    const char* ctxID);
```

6.1.2 Comunicação

Como mencionado anteriormente, o GIC é responsável por emitir pedidos de recomendações às entidades quando necessário bem como tratar as respostas destes pedidos determinando a reputação e credibilidade das entidades. A emissão e tratamento dos pedidos de recomendação é feita através de duas funções que são apresentadas na listagem de código 6.2. Essas duas funções fornecem um nível de abstração para que o GIC seja capaz de lidar com a grande variedade de protocolos de comunicação presentes nos ambientes pervasivos.

A função $issue\_RE\_RQST$ é responsável pela emissão dos pedidos de recomendação feitos pelo GIC. Quando se faz necessário determinar a reputação de uma entidade, o GIC faz uso desta função para emitir os pedidos de recomendação, passando para a mesma o identificador e o contexto nos quais a entidade deve ser avaliada. Fica a cargo da camada de comunicação o envio das mensagens de recomendação às entidades disponíveis no ambiente pervasivo.

Sempre que uma nova recomendação estiver disponível, a camada de comunicação deve repassá-la ao GIC através da função $GIC\_API\_receive\_RRPLY$. Os parâmetros desta função identificam a fonte da recomendação, a entidade avaliada por ela, o contexto, a defasagem temporal da recomendação e o valor de confiança obtido pelo recomendador.

Código 6.2: Funções de comunicação

```
void GIC_API_receive_RRPLY(const char* srcKey,
                const char* targetKey, const char* ctxID,
                double ellapsedTime, double recomendation);
                ...
typedef int (*issue_RE_RQST)(const char* targetKey,
                const char* context, void* ptr_obj);
```

6.1.3 Reputação e credibilidade

Este módulo é responsável por determinar a reputação e a credibilidade das entidades. Como pode ser observado a partir da Figura 6.1, este módulo acessa diretamente o módulo de comunicação para emitir os pedidos de recomendação, já que estas são necessárias para se determinar a reputação. Como o recebimento das recomendações depende da tecnologia de comunicação e das entidades disponíveis no ambiente pervasivo, a determinação da reputação é assíncrona. Assim, são disponibilizadas funções síncronas e assíncronas para a obtenção da reputação e do valor de confiança geral, que depende da reputação. As funções síncronas acessam diretamente a base de dados do GIC (e.g. $GIC\_API\_get\_ctxRepValue$ e $GIC\_API\_get\_generalTrustValue$), enquanto as funções assíncronas (e.g. $GIC\_API\_request\_generalTrustValue$ e $GIC\_API\_request\_update\_reputation$) recebem além dos parâmetros como contexto e identificação da entidade, o ponteiro para uma

função que deve ser chamada quando a reputação ou o valor de confiança geral for determinada. Deve-se notar que apenas as funções assíncronas fazem com que pedidos de recomendações sejam emitidos. A interface deste módulo é composta pelas funções apresentadas na listagem de código 6.3. Além do valor de confiança geral e da reputação também é possível ter acesso à credibilidade da entidade através da função $GIC\_API\_get\_recRepValue$.

Código 6.3: Funções relacionadas a reputação

```
double GIC_API_get_recRepValue( const char* queriedKey);
double GIC_API_get_ctxRepValueP( const char* queriedKey,
                                 const char* ctxID );
double GIC_API_get_ctxRepValueC( const char* queriedKey,
                                 const char* ctxID );
double GIC_API_get_generalTrustValueC( const char* key,
                                 const char* ctxID );
double GIC_API_get_generalTrustValueP( const char* key,
                                 const char* ctxID );
int GIC_API_request_generalTrustValueP( const char* key,
          const char* ctxID, void* ptr_obj, listener lst );
int GIC_API_request_generalTrustValueC( const char* key,
          const char* ctxID, void* ptr_obj, listener lst );
int GIC_API_request_update_reputationC( const char* targetKey,
          const char* ctxID,void* ptr_obj, listener lst );
int GIC_API_request_update_reputationP( const char* targetKey,
          const char* ctxID,void* ptr_obj, listener lst );
              ...
typedef void (*listener)( const char* key,const char* context,
                  double value, void* ptr_obj );
```

6.1.4 Cooperação

Através do módulo de cooperação, entidades podem tomar decisões baseadas na confiança. Pelo fato do limiar de cooperação depender do valor de confiança geral, a requisição da decisão baseada na confiança pode ser feita assincronamente, assim como a requisição da reputação apresentada anteriormente. Na listagem de código 6.4 apresenta-se as versões síncronas e assíncronas para a requisição de decisão baseada na confiança. A decisão vem na forma da `struct` $TrustDecision$, que possui, além da decisão em cooperar ou não, os valores nos quais a decisão foi baseada, isto é, *nível de risco* e o *nível de cooperação* definidos na Seção 5.1.3. Além das funções que fornecem decisão baseada na confiança, ainda há a possibilidade de se consultar os valores do *nível de cooperação* e do *nível de risco* de qualquer entidade.

Código 6.4: Funções relacionadas a cooperação

```
void GIC_API_request_trustBasedDecisionP( const char* key,
        const char* ctxID,void* ptr_obj, tbdListener lst );
void GIC_API_request_trustBasedDecisionC( const char * key,
        const char* ctxID,void* ptr_obj, tbdListener lst );
TrustDecision GIC_API_get_trustBasedDecisionP( const char* key,
                                        const char* ctxID );
TrustDecision GIC_API_get_trustBasedDecisionC( const char* key,
                                        const char * ctxID );
double GIC_API_get_risk_levelC( const char* key,
                                        const char* ctxID );
double GIC_API_get_risk_levelP( const char* key,
                                        const char* ctxID );
double GIC_API_get_cooperation_levelC( const char* key,
                                        const char* ctxID );
double GIC_API_get_cooperation_levelP( const char* key,
                                        const char* ctxID );
                                ...
typedef void (*tbdListener)( const char* key,const char* context,
                TrustDecision td, void* ptr_obj );
```

6.1.5 Configuração

Algumas configurações são necessárias para o funcionamento adequado do GIC. Estas configurações estão relacionadas à base de dados e ao módulo de comunicação. As funções de configuração são apresentadas na listagem de código 6.5.

Código 6.5: Funções de configuração

```
void GIC_API_Init();
void GIC_API_Finish();
 void setRequestFunction( issue_RE_RQST rqst_func, void* ptr_obj );
```

As funções GIC_API_Init e GIC_API_Finish estão relacionadas à configuração da base de dados. A primeira que deve ser chamada na inicialização do GIC, pois carrega as informações de confiança presentes na base de dados e as torna disponíveis para utilização. A segunda, que deve ser chamada ao fim da utilização do GIC, pois armazena as novas informações obtidas durante a utilização do mesmo, possibilitando que elas possam ser recuperadas na próxima utilização.

O módulo de comunicação, como mencionado anteriormente, necessita de uma ligação direta com a camada de comunicação associada ao GIC para enviar de pedidos de recomendações

às entidades. Esta ligação é feita através da implementação, por parte da camada de comunicação, de uma função que siga a definição de $issue\_RE\_RQST$ como apresentado na listagem de código 6.2. No entanto, o GIC necessita acessar a função que implementa essa definição na camada de comunicação para que ela possa ser chamada quando necessário. Isso é feito a partir da função $setRequestFunction$ que registra junto ao GIC, a função da camada de comunicação que implementa a definição da função $issue\_RE\_RQST$.

6.2 O *middleware Wings* e o suporte à confiança

Para adicionar suporte à confiança ao *middleware Wings* é necessário que o mesmo forneça acesso às aplicações ao Gerenciador de Informação de Confiança. Para que isso seja possível o conhecimento da arquitetura do *middleware* é fundamental. A seguir, apresenta-se uma visão geral sobre o *middleware Wings* bem como uma descrição de sua arquitetura.

6.2.1 Visão Geral sobre o *Wings*

O *middleware Wings* foi desenvolvido para facilitar o desenvolvimento de aplicações em ambientes pervasivos. A partir dele, aplicações podem descobrir/anunciar serviços nestes ambientes utilizando diferentes protocolos de descoberta de serviços. Com uma arquitetura modular e com suporte a *plug-ins*, o *Wings* possibilita que suas funcionalidade sejam alteradas em tempo de execução. A seguir é apresentada a arquitetura do *middleware Wings* juntamente com a descrição de cada um de seus módulos.

Arquitetura

A arquitetura do *Wings* é ilustrada na Figura 6.2 e consiste em quatro módulos: Evolução dinâmica, Redes pervasivas, Ciência do contexto e Fachada. Estes módulos são descritos a seguir:

- O módulo de Evolução dinâmica é responsável por prover mecanismos para permitir que o *middleware* seja atualizado em tempo de execução através da adição e remoção de *plug-ins*. Isso permite que o *middleware* tenha suas funcionalidades alteradas durante a execução.

- O módulo de Redes pervasivas é responsável por possibilitar que as aplicações façam a descoberta de nós e serviços nos ambientes pervasivos. Essas duas funcionalidades são encapsuladas em *plug-ins* o que possibilita que as aplicações acessem serviços e nós de através de diferentes soluções de comunicação (e.g. Bluetooth SDP, UPnP).

- O módulo de Ciência do contexto disponibiliza informações de contexto que sejam de

interesse das aplicações como, por exemplo, quantidade de dispositivos nas proximidades, etc.

- O módulo da Fachada tem como principal função prover um ponto de acesso único para as aplicações possibilitando que elas acessem as funcionalidades providas pelos *plug-ins* do Wings. Além disso, a Fachada é responsável por realizar o gerenciamento dos *plug-ins* possibilitando que estes sejam adicionados ou removidos.

Figura 6.2: Arquitetura do *middleware Wings*

6.2.2 Implementação do *Wings*

O *middleware Wings* possui implementações em Symbian/C++ e Java. Ambas implementam o módulo de Redes pervasivas utilizando Bluetooth como tecnologia de comunicação. Isto permite a descoberta de nós capacitados com este tipo de tecnologia além de possibilitar a descoberta/anúncio de serviços utilizando o protocolo Bluetooth SDP. Neste trabalho optou-se por criar uma nova implementação do *middleware Wings* utilizando Python como linguagem de programação. Nesta implementação, o módulo de Redes pervasivas foi desenvolvido utilizando o UPnP como solução de descoberta de serviços. A arquitetura do UPnP permite comunicação ponto-a-ponto entre computadores em redes locais e dispositivos portáteis em redes sem fio. A implementação do *middleware* foi feita para *Internet Tablets* e se baseia no *framework* de UPnP BRisa desenvolvido no Laboratório de Sistemas Embarcados e Computação Pervasiva.

6.2.3 Suporte à confiança

A utilização de um *middleware* como o *Wings* em ambientes pervasivos reduz a complexidade no desenvolvimento de aplicações, pois fornece um nível de abstração para lidar com a diversidade de protocolos de descoberta/anúncio de serviços. Ele possibilita que a descoberta e o anúncio de serviços sejam feitos utilizando diferentes soluções de comunicação.

As aplicações desenvolvidas utilizando o *Wings* devem ter acesso às informações de confiança sobre serviços que utilizam e sobre os clientes de seus serviços de modo que possam tomar decisões. O GIC é utilizado com este propósito. Ele é integrado ao *middleware* através de um serviço nativo de confiança, que permite que as aplicações acessem suas funcionalidades. Como

o GIC é implementado em C, o acesso a sua interface, apresentada anteriormente, é realizado através de *bidings* implementados utilizando o Pyrex[1]. Estes *bidings* fazem a ligação direta entre a interface do GIC implementada em C e a interface implementada em Python, denominada Py_GIC. Além de utilizar o Py_GIC para implementar um serviço nativo de confiança, se faz necessário criar um serviço que possibilite a troca de recomendações entre as entidades de modo que a reputação possa ser determinada.

Na Figura 6.3 apresenta-se como os serviços foram adicionados ao *Wings*. A partir dela pode-se observar a utilização dos módulos da Fachada e de Redes pervasivas. O módulo de Redes pervasivas, que oferece as funcionalidades de descoberta de nós e descoberta de serviços, é utilizado pelas aplicações através do módulo da Fachada e pelos serviços nativos do *Wings*, sendo acessados diretamente. São adicionados dois serviços nativos ao *Wings*: Serviço de confiança e o Serviço de recomendação. O primeiro é utilizado pelas aplicações locais para acessar as informações de confiança, estando disponível através da Fachada, da mesma forma que um serviço remoto. O propósito do segundo é disponibilizar recomendações para aplicações executando em outros dispositivos. A seguir, estes serviços são descritos com mais detalhes.

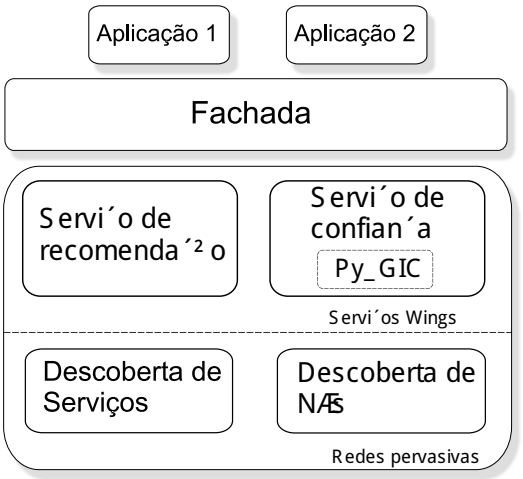


Figura 6.3: Organização dos serviços no *middleware Wings*

Serviço de recomendações

O Serviço de recomendação é anunciado pelas entidades nos ambientes pervasivos possibilitando que as mesmas troquem informações entre si. Como pode ser observado a partir da Figura 6.3 ele é implementado como um serviço *Wings*. Ao receber uma requisição de uma entidade do ambiente peravasivo, o Serviço de recomendação busca na base de dados do GIC a informação de recomendação e envia a resposta à entidade que solicitou a recomendação.

[1] *http://www.cosc.canterbury.ac.nz/greg.ewing/python/Pyrex/*

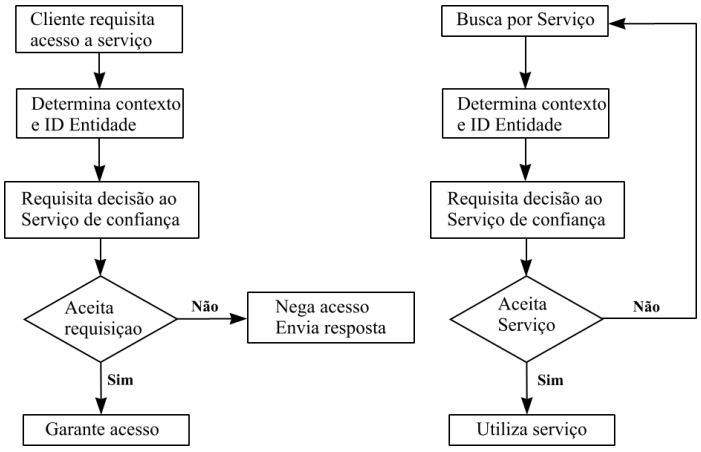
Serviço de confiança

O serviço de confiança utiliza a implementação do GIC para possibilitar que as aplicações acessem as informações de confiança e forneçam ao GIC informações sobre experiências. Além disso, o Serviço de confiança é responsável por gerenciar a emissão e recebimento das recomendações. Isso é necessário, pois o Serviço de confiança necessita realizar a determinação da reputação. Para identificar quais entidades estão disponíveis no ambiente pervasivo e poder utilizar os Serviço de recomendação anunciado por elas, o Serviço de confiança faz uso da funcionalidade de Descoberta de nós do módulo de Redes pervasivas. Ao realizar a Descoberta de nós a entidade é capaz de identificar os nós que anunciam o Serviço de recomendação e adicioná-los à uma lista de recomendadores que é atualizada periodicamente. Deste modo, a lista de recomendadores é acessada e os pedidos de recomendação são emitidos sempre que for necessário se determinar a reputação de alguma entidade.

Do ponto de vista das aplicações, o serviço de confiança é necessário principalmente em dois momentos: na seleção de provedores de serviço e no controle de acesso de clientes dos serviços anunciados. Na Figura 6.4 são apresentados dois diagramas de fluxo que podem ser utilizados nestes casos. No diagrama da Figura 6.4(a) trata-se do controle de acesso a um serviço anunciado. Ao receber a requisição de um cliente interessado, a entidade determina o contexto associado à requisição e o identificador da entidade requisitante. O nome do serviço pode ser utilizado na formação do contexto e o ID da Entidade pode ser gerado a partir da utilização de uma função *hash* segura sobre a chave pública da entidade[2]. Com base nestas informações, a aplicação requisita ao Serviço de confiança uma decisão utilizando o limiar de cooperação (ver Seção 6.1.4 para maiores detalhes). Após receber a resposta, a entidade decide se deve ou não permitir o acesso ao serviço. Caso o acesso não seja permitido, a entidade envia uma resposta com um código de erro informando que o pedido foi negado.

No diagrama da Figura 6.4(b) apresenta-se uma sugestão dos procedimentos necessários para realizar uma busca de serviços confiáveis em ambientes pervasivos. Este procedimento tem início na busca pelo serviço de interesse feita através do *middleware*. Ao encontrar um serviço que corresponda aos critérios da busca, a entidade determina o ID da Entidade e o contexto, que são obtidos da mesma forma que no caso anterior. Em seguida a entidade requisita uma decisão baseada no limiar de cooperação ao Serviço de confiança. Caso a decisão seja positiva o serviço é utilizado, caso contrário, a entidade continua a busca.

[2]*Assume-se que cada dispositivo possui uma chave pública e que as mensagens trocadas entre eles são assinadas*



(a) Seleção de um cliente (b) Seleção de um provedor

Figura 6.4: Diagrama de fluxo da tomada de decisão baseada na confiança.

6.2.4 Aplicação de teste

A aplicação de teste desenvolvida permite o compartilhamento, busca e obtenção automática de arquivos de interesse do usuário em um ambiente pervasivo. Esta aplicação, denominada *FileShare*, foi desenvolvida utilizando a linguagem Python e o *middleware Wings* e pode ser executada em dispositivos portáteis como o Nokia N800[3]. A aplicação anuncia o serviço de compartilhamento de arquivos possibilitando que outras aplicações procurem automaticamente por arquivos de interesse do usuário.

Os dispositivos utilizados estão capacitados com Wi-Fi e estabelecem MANETS para descobrir/anunciar serviços. Utilizam-se dois critérios para avaliar provedores e clientes do serviço de compartilhamento de arquivos. O critério de avaliação do provedor leva em consideração a integridade dos arquivos obtidos. O critério de avaliação dos clientes leva em consideração a quantidade de requisições em um determinado intervalo de tempo.

Na Figura 6.5 é apresentada a configuração de um ambiente pervasivo utilizado para avaliar a integração do GIC ao *middleware Wings*. Este ambiente é constituído de três dispositivos que executam a aplicação *FileShare*. Para localizar e obter arquivos, a aplicação utiliza o *Wings* para procurar por serviço de compartilhamento de arquivos anunciados por outras aplicações executando em outros dispositivos. Ao encontrar um serviço de compartilhamento de arquivos, a aplicação utiliza o Serviço de confiança do *middleware* para determinar se deve ou não interagir com o serviço em questão, seguindo a sugestão descrita no diagrama da Figura 6.4(b). Caso a interação possa ocorrer a aplicação procura, através do serviço, por arquivos de interesse do usuário. Caso contrário, a aplicação segue buscando pelo serviço de compartilhamento de arquivos em outros dispositivos disponíveis no ambiente. Ao receber uma requisição para uti-

[3] *http://www.nseries.com/n800*

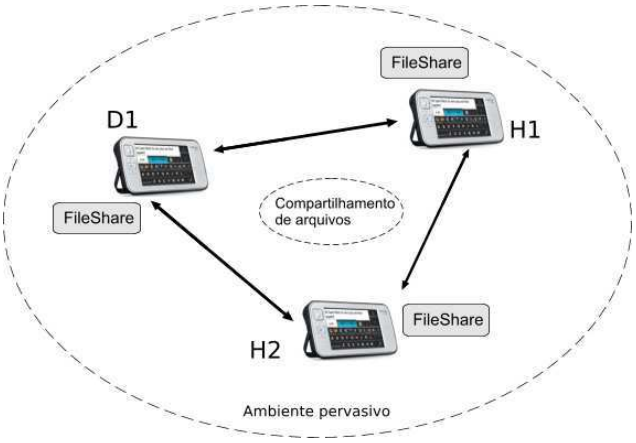


Figura 6.5: Ambiente pervasivo utilizando a aplicação de teste

lização do serviço de compartilhamento de arquivos, a aplicação utiliza o Serviço de confiança para determinar se deve interagir com a entidade requisitante seguindo o procedimento descrito no diagrama da Figura 6.4(a).

Neste cenário, baseado na Figura 6.5, considera-se que a entidade, $D1$ não segue os critérios de utilização de serviço, isto é, ela faz mais requisições do que os limites estipulados, mas compartilha arquivos íntegros. Tem-se ainda que as outras duas entidades, $H1$ e $H2$, seguem todos os critérios. O ambiente foi configurado de modo que as entidades eram desconhecidas umas das outras a princípio. Durante a avaliação, observou-se que os dispositivos $H1$ e $H2$ foram capazes de detectar o comportamento desonesto na utilização de serviços por parte de $D1$, cessando apenas o fornecimento do serviço para o mesmo, sem deixar de utilizar o serviço disponibilizado por ele. Observou-se que as entidades $H1$ e $H2$ foram capazes de detectar o comportamento desonesto associado ao contexto de utilização do serviço de compartilhamento de arquivos. $H1$ e $H2$ deixaram de interagir com $D1$ apenas neste contexto específico mantendo as relações nos outros contextos agindo deste modo com eficiência contra o comportamento desonesto no ambiente pervasivo.

6.3 Sumário

Neste Capítulo apresentou-se a implementação do GIC que é dividida nos módulos de cooperação, experiências, reputação e comunicação. A interface de cada um destes módulos foi descrita bem como as relações de dependência entre os módulos. Apresentou-se ainda a integração do GIC ao *middleware Wings*. Foram descritos os serviços adicionados ao *middleware* para dar suporte à análise da confiança. Foi desenvolvida uma aplicação simples de compartilhamento de arquivos para testar a implementação do *middleware* e sua integração ao GIC.

Capítulo 7

Considerações Finais

Em ambientes dinâmicos e heterogêneos como os ambientes pervasivos, devem ser implementados mecanismos para a disponibilização e/ou utilização de serviços. Pelo fato desses ambientes poderem ser formados em qualquer lugar a qualquer momento, os dispositivos podem ficar isolados e sem acesso à Internet, por exemplo. Nestes casos, as únicas informações disponíveis para os dispositivos são as experiências obtidas por ele e as compartilhadas por outros. Mesmo nestes casos, as aplicações devem ser capazes de selecionar com quais outros dispositivos deseja cooperar através da disponibilização/utilização de serviços, evitando abusos e realizando a escolha adequada dos serviços a serem utilizados.

Para que isso seja possível, foi desenvolvido um modelo de confiança capaz de, a partir das informações disponíveis, determinar a confiança nos diversos contextos estabelecidos entre os dispositivos nos ambientes pervasivos. Como observado pelos resultados apresentados no Capítulo 5, o modelo de confiança desenvolvido é eficiente contra comportamento variável dos dispositivos, emissão de recomendações falsas, e ações em colusão. Isso faz com que ele seja indicado para o uso em ambientes pervasivos que exigem robustez contra este tipo de ação.

Este modelo de confiança foi implementado pelo Gerenciador de Informação de Confiança. O GIC implementa ainda um mecanismo de tomada de decisão baseado em informações de confiança. Utilizando este mecanismo, aplicações em ambientes pervasivos, podem tomar decisões baseadas na confiança determinada pelo modelo de confiança reduzindo a necessidade de intervenção do usuário. Como observado a partir dos resultados apresentados no Capítulo 5, o mecanismo de tomada de decisão se mostrou adequado, pois reduziu significativamente a cooperação nos contextos cujas entidades observadas apresentavam comportamento desonesto.

Por fim, a adição de suporte a análise da confiança ao *middleware Wings* possibilitou que as aplicações fossem capazes de lidar com a heterogeneidade dos ambientes pervasivos garantindo a cooperação com entidades honestas elevando a eficiência da comunicação nestes ambientes.

7.1 Discussão

A utilização da credibilidade dos recomendadores se mostrou bastante eficaz para descartar recomendações falsas e lidar com ações em colusão. No entanto, outro tipo de ação denominada discriminação pode ser empregada. Nesta ação, entidades discriminam outras entidades agindo de forma diferente com cada uma. Isto pode reduzir a similaridade entre recomendações e experiências diretas reduzindo o desempenho do modelo de evolução da credibilidade. Em alguns casos, o estabelecimento de novas relações de confiança pode ser afetado, pois quando for necessário determinar a reputação de novas entidades pode não haver recomendadores com credibilidade suficiente. No entanto, a redução do desempenho na determinação da credibilidade não afeta ativamente o modelo de confiança, pois entidades agindo de forma discriminatória não podem levar à determinação incorreta da reputação nem da confiança, mantendo assim, relações de confiança previamente estabelecidas inalteradas.

7.2 Trabalhos Futuros

Pode-se adicionar ao GIC o suporte a configuração através de perfis de confiança. Cada perfil definiria diferentes valores para os parâmetros do GIC e do modelo de confiança de forma a obter diferentes modos de operação que variem, por exemplo, de mais conservador, onde o estabelecimento de novas relações de confiança seria mais difícil, a mais arrojado, onde novas relações de confiança seriam estabelecidas rapidamente.

Pode-se considerar a utilização de informações de contexto para realizar a configuração automática do perfil do GIC. Um exemplo de informação de contexto que pode ser utilizada é a localização. Através dela, o perfil do GIC pode ser alterado de acordo com a localização da entidade permitindo, por exemplo, que um perfil mais conservador seja escolhido quando a entidade está localizada em ambientes públicos, onde mais entidades desconhecidas estão presentes.

O mecanismo de tomada de decisão utilizado pelo GIC leva em consideração apenas informações de confiança. Um possível trabalho futuro consiste em estender os tipos de informações utilizadas por este mecanismo para considerar, além de informações de confiança, a utilização de credenciais para realizar o controle de acesso baseado em papéis.

Referências Bibliográficas

[1] Alfarez Abdul-Rahman and Stephen Hailes. A distributed trust model. In *NSPW '97: Proceedings of the 1997 workshop on New security paradigms*, pages 48–60, New York, NY, USA, 1997. ACM.

[2] F. Almenarez, A. Marin, D. Diaz, and J. Sanchez. Developing a model for trust management in pervasive devices. *Pervasive Computing and Communications Workshops, 2006. PerCom Workshops 2006. Fourth Annual IEEE International Conference on*, Março 2006. ISBN: 0-7695-2520-2.

[3] Florina Almenarez, Andres Marn, Celeste Campo, and Carlos Garcia. Ptm: A pervasive trust management model for dynamic open environments. *In Proceedings of Workshop on Pervasive Security, Privacy and Trust (PSPT)*, 2004.

[4] Ingmar Baumgart, Bernhard Heep, and Stephan Krause. Oversim: A flexible overlay network simulation framework. In *Proceedings of 10th IEEE Global Internet Symposium GI*, pages 79–84, Maio 2007.

[5] L. Capra and M. Musolesi. Autonomic trust prediction for pervasive systems. *Advanced Information Networking and Applications, 2006. AINA 2006. 20th International Conference on*, 2, Abril 2006.

[6] Yong Chen, Christian Damsgaard Jensen, Elizabeth Gray, Vinny Cahill, and Jean-Marc Seigneur. A general risk assessment of security in pervasive computing. Technical Report TCD-CS-2003-45, Department of Computer Science, Trinity College Dublin, Novembro 2003.

[7] Fabrizio Cornelli, Ernesto Damiani, Sabrina De Capitani di Vimercati, Stefano Paraboschi, and Pierangela Samarati. Choosing reputable servents in a p2p network. In *WWW '02: Proceedings of the 11th international conference on World Wide Web*, pages 376–386, New York, NY, USA, 2002. ACM.

[8] Elizabeth Chang Farookh Khadeer Hussain. Trustworthiness and ccci metrics in p2p

communication. *International Journal of Computer System Science and Engineering*, 19, 2004.

[9] Emerson Cavalcante Loureiro Filho. Um middleware extensível para disponibilização de serviços em ambientes pervasivos. Master's thesis, Universidade Federal de Campina Grande, Junho 2006.

[10] The Internet Engineering Task Force. Public-key infrastructure (x.509) (pkix).

[11] Diego Gambetta. *Can We Trust Trust?* Basil Blackwell, 1988. Reprinted in electronic edition from Department of Sociology, University of Oxford, chapter 13, pp. 213-237.

[12] Uwe Hansmann, T. Stober, L. Merk, and M. Nicklous. *Pervasive Computing*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2003.

[13] Farookh Khadeer Hussain, Elizabeth Chang, and Tharam S. Dillon. Trustworthiness and ccci metrics in p2p communication. *Comput. Syst. Sci. Eng.*, 19(3), 2004.

[14] Omar Khadeer Hussain, Elizabeth Chang, Farookh Khadeer Hussain, Tharam S. Dillon, and Ben Soh. Risk in trusted decentralized communications. In *ICDE Workshops*, page 1198, 2005.

[15] Audun Jøsang and Stéphane Lo Presti. Analysing the relationship between risk and trust. In *Second International Conference on Trust Management*, pages 135–145, 2004.

[16] A. Jsang. An algebra for assessing trust in certification chains. *Proceedings of the Network and Distributed Systems Security (NDSS'99) Symposium*, pages 52–61, Julho 1999.

[17] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82:35–45, 1960.

[18] Jinshan Liu and Valérie Issarny. Enhanced reputation mechanism for mobile ad hoc networks. pages 48–62. 2004.

[19] S. Marsh. *Formalising Trust as a Computational Concept*. PhD thesis, University of Sterling, 1994.

[20] Ghita Kouadri Mostefaoui, Jacques Pasquier-Rocha, and Patrick Brezillon. Context-aware computing: A guide for the pervasive computing community. In *ICPS '04: Proceedings of the The IEEE/ACS International Conference on Pervasive Services (ICPS'04)*, pages 39–48, Washington, DC, USA, 2004. IEEE Computer Society.

[21] Asad A. Pirzada and Chris Mcdonald. Establishing trust in pure ad-hoc networks. In *ACSC '04: Proceedings of the 27th Australasian conference on Computer science*, pages 47–54, Darlinghurst, Australia, Australia, 2004. Australian Computer Society, Inc.

[22] G. Pitsilis and L. Marshall. Trust as a key to improving recommendation systems. Technical Report 875, Newcastle University, School of Computing Science, Nov 2004.

[23] F. J. Richard. A flexible growth function for empirical use. *J. Exp. Bot.*, 10:290–300, 1959.

[24] Debashis Saha and Amitava Mukherjee. Pervasive computing: A paradigm for the 21st century. *Computer*, 36(3):25–31, 2003.

[25] M. Satyanarayanan. Pervasive computing: vision and challenges. *Personal Communications, IEEE [see also IEEE Wireless Communications]*, 8(4):10–17, 2001.

[26] A. SCHEIN, A. POPESCUL, L. UNGAR, and D. PENNOCK. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2002)*, pages 253–260, 2002.

[27] Stefan Schmidt, Robert Steele, Tharam S. Dillon, and Elizabeth Chang. Fuzzy trust evaluation and credibility development in multi-agent systems. *Appl. Soft Comput.*, 7(2):492–505, 2007.

[28] Moushumi Sharmin, Shameem Ahmed, and Sheikh I. Ahamed. An adaptive lightweight trust reliant secure resource discovery for pervasive computing environments. In *PERCOM '06: Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications (PERCOM'06)*, pages 258–263, Washington, DC, USA, 2006. IEEE Computer Society.

[29] Moushumi Sharmin, Shameem Ahmed, and Sheikh I. Ahamed. Marks (middleware adaptability for resource discovery, knowledge usability and self-healing) for mobile devices of pervasive computing environments. In *ITNG '06: Proceedings of the Third International Conference on Information Technology: New Generations (ITNG'06)*, pages 306–313, Washington, DC, USA, 2006. IEEE Computer Society.

[30] Andras Varga. The omnet++ discrete event simulation system. In *Proceedings of the European Simulation Multiconference*, pages 319–324, Prague, Czech Republic, Junho 2001. SCS – European Publishing House.

[31] Yao Wang and Julita Vassileva. Trust and reputation model in peer-to-peer networks. In *P2P '03: Proceedings of the 3rd International Conference on Peer-to-Peer Computing*, page 150, Washington, DC, USA, 2003. IEEE Computer Society.

[32] M. Weiser. The computer for the 21st century. *Scientific America*, 42:94–104, Setembro 1991.

[33] S.T. Wolfe, S.I. Ahamed, and M. Zulkernine. A trust framework for pervasive computing environments. *Computer Systems and Applications, 2006. IEEE International Conference on.*, pages 312–319, 2006.

[34] L. Xiong and L. Liu. Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities. *IEEE Transaction On Knowledge and Data Engineering*, 16(7):843–857, Julho 2004.

[35] D. Xiu and Z. Liu. A dynamic trust model for pervasive computing environments. *The Fourth Annual Security Conference*, pages 80–85, Março 2005.

[36] Ilan Yaniv and Eli Kleinberger. Advice taking in decision making: Egocentric discounting and reputation formation. *Organizational Behavior and Human Decision Processes*, 83(2):260–281, November 2000.

[37] Fen Zhu, Matt W. Mutka, and Lionel M. Ni. Service discovery in pervasive computing environments. *IEEE Pervasive Computing*, 4(4):81–90, 2005.