



Unioeste - Universidade Estadual do Oeste do Paraná
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
Colegiado de Informática
Curso de Bacharelado em Informática

**PROPOSTA DE UM SISTEMA AUTOMÁTICO INTELIGENTE PARA O
GERENCIAMENTO DE CONSUMO ENERGÉTICO RESIDENCIAL**

Gustavo Rezende Krüger

CASCAVEL
2009

GUSTAVO REZENDE KRÜGER

**PROPOSTA DE UM SISTEMA AUTOMÁTICO INTELIGENTE PARA O
GERENCIAMENTO DE CONSUMO ENERGÉTICO RESIDENCIAL**

Monografia apresentada como requisito parcial para obtenção do grau de Bacharel em Informática, do Centro de Ciências Exatas e Tecnológicas da Universidade Estadual do Oeste do Paraná - Campus de Cascavel.

Orientador: Prof. Dr. Jorge Bidarra

CASCADEL
2009

GUSTAVO REZENDE KRÜGER

**PROPOSTA DE UM SISTEMA AUTOMÁTICO INTELIGENTE PARA O
GERENCIAMENTO DE CONSUMO ENERGÉTICO RESIDENCIAL**

Monografia apresentada como requisito parcial para obtenção do Título de *Bacharel em Informática*, pela Universidade Estadual do Oeste do Paraná, Campus de Cascavel, aprovada pela Comissão formada pelos professores:

Prof. Dr. Jorge Bidarra (Orientador)
Colegiado de Informática, UNIOESTE

Prof. Dr. Clodis Boscarioli
Colegiado de Informática, UNIOESTE

Prof^a. Dr^a. Cláudia Brandelero Rizzi
Colegiado de Informática, UNIOESTE

Cascavel, 9 de dezembro de 2009

“Viva sem pressa, como se fosse viver para sempre, mas não perca os detalhes que a vida lhe proporciona, como se fosse seu último dia.”

Gustavo Rezende Krüger

AGRADECIMENTOS

Primeiramente quero agradecer a Deus pela oportunidade concedida e por estar sempre ao meu lado, auxiliando-me a superar os obstáculos da vida. Com muito amor, sou eternamente grato a meus pais por me apresentarem ao mundo e por estarem, também, sempre ao meu lado, orientando-me e apoiando-me em qualquer que seja o momento. Ao amor que a mim dedicaram, não possuo palavras para agradecer.

Aos meus amigos, gostaria de dar-lhes o mundo em troca de suas amizades. Sem os momentos que passei com vocês, talvez eu não estivesse aqui hoje, nesta etapa da vida. Também sou grato às pessoas que entraram na minha vida recentemente, pois cultivar novas amizades é um dos motivos que fazem de mim uma pessoa feliz.

Quanto ao meu trabalho, agradeço meu orientador, Jorge Bidarra, por me ensinar a aprender. Agradeço meu grande amigo, recém formado, Jorge Henrique Bueno Gomes, por me apresentar a ferramenta Elipse SCADA, que me economizou 3 meses de implementação.

Lista de Figuras

2.1	Organização dos Componentes Envolvidos nas Funcionalidades do SCADA, adaptada de Terezinho e Carvalho [31]	9
3.1	Interação entre Agente e Ambiente, adaptada de Wooldridge [35]	14
3.2	Exemplo de um Caso Simplificado de um Sistema de Conserto de Impressoras, Wangenheim e Wangenheim [33]	26
3.3	Ciclo Genérico de Processamento de um Sistema RBC, adaptada de Aamodt e Plaza [1]	28
4.1	Modelagem da Arquitetura Proposta	35
4.2	Estrutura de um Agente no Sistema Proposto	38
4.3	Exemplo de uma Mensagem no Formato da FIPA	40
4.4	Formato de Regras de Comportamento de um Agente, adaptada de Moll [25]	40
5.1	Fluxo de atividades do Agente de Perfil	43
5.2	Fluxo de atividades do Agente Administrador	44
5.3	Fluxo de atividades do Agente de Sensores	45
5.4	Fluxo de atividades do Agente de Dispositivos	46
5.5	Fluxo de atividades do Agente de Consumo Energético	47
5.6	Formulário de Casos do Sistema	51
5.7	Lista dos cômodos, equipamentos e suas respectivas potências	58
5.8	Lista das variáveis do ambiente simulado	58
6.1	Tela do Sistema em Execução	61
6.2	Tela do Ambiente Simulado em Execução	61

Lista de Tabelas

5.1	Similaridade local do atributo (<i>Cômodo</i>).	53
5.2	Similaridade local do atributo (<i>Nível de Interação</i>).	53
5.3	Similaridade local dos atributos que representam os estados dos dispositivos. . .	53
5.4	Similaridade local dos atributos que representam as preferências dos dispositivos.	53

Lista de Abreviaturas e Siglas

AC	<i>Agentes Computacionais</i>
ACI	<i>Agentes Computacionais Inteligentes</i>
CA	<i>Controle e Automação</i>
CLP	<i>Controlador Lógico Programável</i>
IA	<i>Inteligência Artificial</i>
IAC	<i>Inteligência Artificial Conexionista</i>
IAD	<i>Inteligência Artificial Distribuída</i>
IAE	<i>Inteligência Artificial Evolucionária</i>
IAH	<i>Inteligência Artificial Híbrida</i>
IAM	<i>Inteligência Artificial Monolítica</i>
IAS	<i>Inteligência Artificial Simbólica</i>
OPC	<i>Ole for Process Control (OLE para Controle de Processos)</i>
POO	<i>Paradigma Orientado a Objeto</i>
RBC	<i>Raciocínio Baseado em Casos</i>
RNA	<i>Redes Neurais Artificiais</i>
SDP	<i>Solução Distribuída de Problemas</i>
SAR	<i>Sistema de Automação Residencial</i>
SARI	<i>Sistema de Automação Residencial Inteligente</i>
SCA	<i>Sistemas de Controle Automático</i>
SCADA	<i>Sistemas de Controle e Aquisição de Dados</i>
SMA	<i>Sistemas Multiagente</i>

Sumário

Lista de Figuras	vi
Lista de Tabelas	vii
Lista de Abreviaturas e Siglas	viii
Lista de Símbolos	ix
Sumário	ix
Resumo	xi
1 Introdução	1
1.1 Trabalhos Correlatos na Área de Automação Residencial	2
1.2 Proposta	4
1.3 Organização do Trabalho	5
2 Automação Residencial	6
2.1 Sistemas de Automação Residencial - SAR	7
2.1.1 Sistemas de Supervisão e Aquisição de Dados - SCADA	8
2.2 Desenvolvimento de Sistemas de Automação Residencial Inteligentes - SARI .	10
3 Inteligência Artificial e Sistemas de Automação Residencial	12
3.1 Agentes Computacionais	14
3.1.1 Arquitetura dos Agentes	15
3.1.2 Ambientes	17
3.2 Inteligência Artificial Distribuída - IAD	19
3.3 Sistemas Multiagente - SMA	20
3.3.1 Comunicação em SMA	21
3.4 Raciocínio Baseado em Casos	24
3.4.1 Representação do Conhecimento	25

3.4.2	O Ciclo do RBC	26
3.4.3	Recuperação de Casos	28
3.4.4	Reutilização de Casos	31
3.4.5	Revisão de Casos	31
3.4.6	Retenção de Casos	31
3.4.7	Aplicação do RBC	32
4	Modelagem do Sistema Proposto	33
4.1	Dados de Entrada	33
4.2	Topologia do Sistema	35
4.3	A Estrutura de Comunicação dos Agentes	37
4.4	A Linguagem de Comunicação entre os Agentes	39
5	Aspectos de Implementação	42
5.1	Implementação dos Agentes	42
5.1.1	Implementando o Agente de Perfil	43
5.1.2	Implementando o Agente Administrador	43
5.1.3	Implementando o Agente de Sensores	44
5.1.4	Implemetando o Agente de Dispositivos	45
5.1.5	Implementando o Agente de Consumo Energético	46
5.2	Implementando o Mecanismo de Controle do Sistema	47
5.2.1	Identificando um Problema	47
5.2.2	Resolução dos Problemas do Tipo 1	49
5.2.3	Implementação do RBC para a Resolução dos Problemas do Tipo 2	50
5.3	Alimentação do Sistema	57
6	Conclusão	60
6.1	Resultados Obtidos	60
6.2	Considerações Finais e Perspectivas Futuras	62
	Referências Bibliográficas	64

Resumo

A aplicação das mais variadas técnicas de Inteligência Artificial (IA), em sistemas de Controle e Automação (CA), está atingindo um estágio de maturidade cada vez mais avançado e ao longo dos últimos anos, novas contribuições para a área de CA têm sido embasadas em fundamentos claros e sólidos da IA. As técnicas de IA não vêm substituir as técnicas clássicas de CA, como os controles ótimo, robusto, adaptativo, estocástico, dentre outros. Antes, elas podem ser utilizadas como complemento, oferecendo alternativas ainda não disponíveis. Seguindo essa tendência, este trabalho busca verificar a aplicabilidade de algumas técnicas surgidas na IA, na construção de Sistemas de Controle Automático (SCA), para o ambiente residencial. Entre as técnicas de IA abordadas, encontra-se a técnica de Agentes Computacionais Inteligentes (ACI), concebidos como uma sociedade de agentes em um Sistema Multiagente (SMA), onde cada agente possui uma arquitetura diferenciada e executa diferentes tarefas relacionadas com o controle e a gerência de recursos existentes no ambiente. O sistema desenvolvido neste trabalho, para gerenciar o consumo energético de uma residência, aplica a técnica de Raciocínio Baseado em Casos (RBC) como mecanismo de aprendizagem de alguns agentes, buscando tornar o sistema adaptativo às exigências do morador e às condições do ambiente controlado.

Palavras-chave: Inteligência Artificial, Sistemas Multiagente, Automação Residencial, Raciocínio Baseado em Casos, Sistemas de Controle Automático.

Capítulo 1

Introdução

O uso consciente da energia elétrica, seja nas indústrias ou residências, é um dos grandes desafios não apenas para o governo brasileiro, mas também para o mundo.

No Brasil, criou-se uma cultura irracional de consumo da energia elétrica pois esta, a princípio, era fornecida de forma abundante, com baixo custo e pequena demanda. Com o passar do tempo, o setor industrial, bem como os setores comerciais e residenciais, aumentaram seu consumo e conseqüentemente sua demanda. Com a crise do petróleo, na década de 70, o governo brasileiro se viu diante da necessidade de aumentar o custo e pesquisar novas fontes de energia. Entretanto, a sociedade ainda repousa na cultura do desperdício energético.

De acordo com as informações do balanço energético nacional, fornecido pelo Ministério de Minas e Energia [24], para cada 100 unidades de energia elétrica ofertada, em 1984, 10 unidades foram perdidas em centros de transformação e distribuição e 48 foram perdidas nos equipamentos de uso final. Assim, restaram apenas 42 unidades para realização de efetivo trabalho. Já em 2004, das mesmas 100 unidades de energia restaram 53 unidades para realização de efetivo trabalho, havendo, portanto, ganhos de 11 unidades de energia, por aumento de eficiência geral. Ainda que se note um aumento na eficiência energética, o desperdício é de quase 50%. O setor residencial, cujo problema é foco deste trabalho, ainda segundo o Ministério de Minas e Energia, consumiu, em 2008, no Brasil, 22% da energia elétrica produzida, ficando atrás apenas do setor industrial. Isto faz deste, um setor influente na estrutura de energia elétrica brasileira.

Ao longo dos últimos anos, tem sido crescente a preocupação em relação ao desperdício energético. A título de exemplo, destaca-se o trabalho desenvolvido pelo PROCEL (Programa Nacional de Conservação de Energia), cuja política é promover a racionalização da produção e do consumo de energia elétrica, para que se eliminem os desperdícios e se reduzam os custos e

investimentos setoriais. Além dos programas do governo, meios alternativos para a redução do desperdício energético vêm evoluindo rapidamente. Dentre os que apresentam bons resultados, podem-se citar os Sistemas de Controle Automático (SCA).

Segundo Nascimento e Yoneyama [26], SCA's podem ser divididos, de forma geral, em duas categorias. De um lado, os sistemas controlados manualmente e de outro, os de controle automático; sistemas capazes de executar tarefas sem a intervenção direta do homem.

A aplicação de SCA's em ambientes residenciais, quer para o controle de consumo energético, quer para outras finalidades, vêm despertando interesse de pesquisadores. No caso específico do controle de gastos de energia, as motivações são várias. Dentre elas, citam-se duas. A primeira diz respeito ao consumo elevado de alguns equipamentos aliado ao alto custo da energia elétrica e a outra, refere-se ao conforto do morador. Racionalizar o consumo, não necessariamente significa que o morador não possa desfrutar de nenhum conforto. Antes, essa racionalização significa utilizar a energia eficientemente, minimizando desperdícios. Devido ao custo elevado da energia elétrica e também da necessidade de se utilizarem os recursos energéticos de maneira eficiente, as pesquisas realizadas nesta área, que seguem com o objetivo de oferecer soluções que possibilitem reduções neste consumo, tem contribuído para a integração das mais diversas áreas da computação, dentre elas destaca-se a Inteligência Artificial (IA).

A automação residencial, por meio da aplicação de técnicas da IA, está evoluindo para o conceito de Domótica Inteligente. SCA's, que antes caracterizavam-se por reagirem, através de regras pré-definidas, a eventos como a captura de presença, mudanças no ambiente, entre outros, agora, se preocupam em interagir com o morador e adaptar-se ao seu comportamento. Estas aplicações, ainda que pouco difundidas, tem apresentado bons resultados. É importante ressaltar que as técnicas da IA não surgem como um meio de substituir SCA's mas como um complemento, oferecendo alternativas antes não disponíveis.

1.1 Trabalhos Correlatos na Área de Automação Residencial

Muitos dos trabalhos que envolvem a Automação Residencial têm sido desenvolvidos pela comunidade europeia e norte americana, onde a escassez de recursos e os problemas ambientais são elevados. Esses países vêm apresentando avanços significativos, tais como ferramentas para controle de consumo energético, monitoramento de demanda, inserção automática de energia

alternativa, transmissão de baixa voltagem pelo ar, entre outros. O uso dessas tecnologias em SCA's residenciais, no Brasil, ainda estão numa fase inicial e, portanto, pouco difundidas. A AURESIDE [2], cujo objetivo é a de fomentar a adoção de tecnologias de automação residencial no país, a fim de difundí-las, vêm realizando cursos de capacitação, formação e certificação profissional em automação residencial.

O principal foco destas tecnologias está no desenvolvimento de Sistemas de Automação Residencial (SAR), cuja ênfase está no gerenciamento de recursos, conforto, segurança e entretenimento. Alguns destes sistemas consideram-se inteligentes. No entanto, em sua maioria, apresentam apenas mecanismos automáticos.

SAR Inteligentes devem considerar características como capacidade de aprendizagem e habilidade proativa. Entende-se por habilidade proativa a capacidade que um indivíduo tem de não apenas reagir a um determinado evento no ambiente mas também de tomar iniciativas e decisões baseadas em seus objetivos e experiências. Neste contexto, Urzêda em [32], propõe um controlador Fuzzy cujo objetivo é gerenciar a temperatura de ambientes que utilizam aparelhos de condicionamento de ar. Uma característica interessante deste trabalho é o fato de que controladores Fuzzy têm como princípio modelar um especialista, o qual é capaz de controlar bem um processo. Se utilizam de regras pré-definidas, mas ao mesmo tempo, são capazes de tratar valores incertos e imprecisos.

Alguns trabalhos consideram a mudança do comportamento humano e aplicam técnicas de aprendizagem. É o caso de Sgarbi e Tonidandel [29], cujo trabalho se caracteriza em criar regras em função do aprendizado com o comportamento dos habitantes de uma casa, dando ênfase à obtenção, tratamento e manutenção das regras. O trabalho consiste, basicamente, em armazenar um determinado número de eventos em uma base de dados. Quando o número de eventos chegar ao limite determinado, um algoritmo de aprendizagem é aplicado, adaptando e modificando as regras a partir dos eventos com maior ocorrência.

Outro trabalho que realiza esse tipo de consideração é o de Moll [25], onde é implementado um protótipo que utiliza as preferências do morador para controlar a temperatura ambiente de uma residência e para economizar energia através da aplicação das técnicas de Sistemas Multigente e Raciocínio Baseado em Casos. Moll acrescenta: "A motivação principal de um sistema de RBC é tentar resolver um novo problema lembrando uma situação anterior similar e, então,

reutilizar as informações e conhecimento armazenados daquela situação”. Segundo Lee [21], o RBC funciona de maneira similar à atividade humana de utilizar experiências passadas para a resolução de novos problemas.

Ainda que mecanismos de aprendizagem resolvam o problema, implementar e modelar tal mecanismo não é trivial, uma vez que a utilização destes, na Domótica Inteligente, remete a uma série de discussões e questões de âmbito multidisciplinar onde o comportamento humano é avaliado. Ou seja, o desenvolvimento de SCA’s inteligentes em ambientes residenciais reúne esforços das mais variadas áreas de conhecimento, tais como a Engenharia, Ciência da Computação, Inteligência Artificial, Psicologia, Sociologia, Filosofia, dentre outras. Desta forma, garantir segurança e conforto torna-se uma tarefa complexa e demorada.

A solução proposta por Bolzani em [6] é a aplicação de SCA’s em ambientes residenciais simulados. A utilização de um simulador, segundo Bolzani, permite antecipar e analisar os resultados das interações humanas com um ambiente computacionalmente ativo, bem como prever o comportamento dos dispositivos frente à essa forma de interação. Seguindo esta linha de raciocínio, Bolzani apresenta um simulador inteligente de uma residência capaz de avaliar aspectos técnicos, bem como aspectos inerentes à interação dos equipamentos com o morador/usuário.

1.2 Proposta

A partir dessa idéia, a proposta deste trabalho é modelar e implementar um SCA inteligente de gerenciamento de consumo energético residencial simulado. Dentre as técnicas de IA, mais especificamente, pretende-se a implementação do paradigma orientado a Agentes Computacionais Inteligentes (ACI). Um agente, de acordo com Russel e Norvig [27], é tudo o que pode ser considerado capaz de perceber seu ambiente por meio de sensores e agir sobre esse ambiente por intermédio de atuadores. Neste caso, o sistema será formado por uma série de agentes que se comunicam entre si onde cada um possui uma arquitetura diferenciada e executa diferentes tarefas relacionadas ao controle e gerência de recursos existentes no ambiente. Para controlar e gerenciar tais recursos, o sistema deverá implementar, como mecanismo de inferência e aprendizagem, a técnica de Raciocínio Baseado em Casos (RBC) que, conforme Wangenheim e Wangenheim [33], é uma abordagem para a solução de problemas e para o aprendizado com

base em experiências passadas. Desta forma, pretende-se aplicar essas técnicas no desenvolvimento de um sistema que seja ao mesmo tempo inteligente e adaptável às exigências do morador e às condições do ambiente controlado.

1.3 Organização do Trabalho

O Capítulo 2 apresenta uma breve descrição das principais características da Automação Residencial, considerando os Sistemas de Automação Residencial, bem como o que devem apresentar para monitorar e controlar o ambiente em que se encontra inserido, a fim de atuar de maneira autônoma sobre o mesmo.

No Capítulo 3 é apresentada uma visão geral da área de Inteligência Artificial e das principais abordagens existentes nesta área, frequentemente aplicadas na área de automação residencial.

No Capítulo 4 é apresentada a modelagem de um SAR implementado a partir do paradigma de SMA, que utiliza, como mecanismo de controle e aprendizado, o RBC.

No Capítulo 5 é apresentada a implementação dos principais elementos considerados neste trabalho. Dentre eles estão o comportamento e funcionalidades dos agentes computacionais considerados e, por fim, o mecanismo de controle adotado.

No Capítulo 6 é discutido os resultados obtidos e as dificuldades encontradas. Neste capítulo, apresentam-se, também, algumas considerações finais sobre o trabalho, bem como perspectivas futuras relacionadas ao mesmo.

Capítulo 2

Automação Residencial

As tecnologias de computação, desde sua origem, vêm auxiliando o ser humano em suas atividades cotidianas, de forma a torná-las mais eficazes. A princípio, surgiram da necessidade do ser humano de automatizar cálculos. No decorrer do tempo, tais tecnologias foram estendidas para os diversos contextos vividos no cotidiano, tais como profissionais, pessoais, acadêmicos, entre outros. Desta forma, duas situações podem ocorrer, sejam elas: (i) a humanidade pode deixar de se preocupar com problemas que podem ser resolvidos através da computação e passar a se focar na solução de problemas ainda não resolvidos ou, (ii) problemas que são de conhecimento de alguns poucos, podem ser automatizados a fim de difundir suas soluções, de forma a permitir que os demais tenham conhecimento.

Tecnicamente, a domótica pode ser definida, de acordo com a Aureside [2], como um conjunto de serviços assistidos por um serviço central capaz de realizar várias funções de gerenciamento e atuação, tais como uma simples tarefa de ligar ou desligar um equipamento qualquer, como uma lâmpada ou um televisor; ou uma tarefa mais complexa, como se adaptar ao comportamento dos habitantes, de forma a interagir proativamente com o ambiente residencial. Tais conjuntos podem estar conectados entre si por meio de uma rede de comunicação interna ou externa e fornecem um conjunto de aplicações responsáveis pela segurança, conforto ambiental, gestão energética, automação de tarefas domésticas, entre outros.

De uma forma mais geral, segundo Nascimento e Yoneyama [26], automatizar significa dotar sistemas de mecanismos e dispositivos capazes de realizar, com o mínimo de intervenção humana, tarefas que satisfaçam restrições tais como segurança, conforto, produtividade, entre outras. A utilização conjunta de dispositivos como sensores e atuadores, em uma residência, permite que se obtenha informações sobre o funcionamento do ambiente residencial, de forma

que seja possível, por meio de mecanismos de controle, gerenciá-lo. Este processo torna possível a automação residencial. A implementação da comunicação e integração destes mecanismos e dispositivos é o que caracteriza um sistema de automação residencial (SAR) [6].

2.1 Sistemas de Automação Residencial - SAR

De modo geral, SAR são compostos por uma rede que possibilita a comunicação com os dispositivos de uma residência a fim de obter dados do ambiente, processar esses dados e gerenciar o ambiente. Para que isso seja possível, é necessário que existam dispositivos (sensores, detectores, etc.) que capturem os dados do ambiente (valor do consumo geral, estado dos equipamentos, etc.), de forma que estes sejam transmitidos a uma ou mais unidades inteligentes capazes de processar esses dados e gerenciar o ambiente por meio de ações de acionamento, ajustes, mensagens de alerta, entre outras [9].

Um SAR pode ser estruturado com base em um ou mais subsistemas, cada um deles responsável por fornecer serviços específicos para tipos específicos de processamento. Controlar gastos energéticos e capturar dados referentes à quantidade de energia consumida pelo morador são exemplos de diferentes serviços fornecidos pelos subsistemas.

Geralmente, um SAR pode utilizar uma tecnologia centralizada ou distribuída. Os sistemas centralizados, são aqueles que possuem apenas uma unidade central cuja função é receber os dados coletados pelos dispositivos de captura, processar esses dados e, por meio dos resultados desse processamento, gerenciar o ambiente. Já os sistemas distribuídos ou descentralizados apresentam diversas unidades com processamento próprio. Cada unidade é responsável por diferentes papéis dentro das necessidades do sistema doméstico e são distribuídas ao longo da instalação, interligadas por uma rede. Assim, os dados são recebidos pelas unidades de acordo com seus papéis [9].

É importante considerar que os subsistemas não necessariamente precisam estar em uma unidade diferente de controle. Isto é, o fato de subsistemas existirem, não implica no fato do sistema ser centralizado ou distribuído.

De acordo com Bolzani [6], a centralização dos sistemas domésticos, diminui o tempo gasto com o projeto e com equipamentos e cabeamentos para a implantação, uma vez que é necessário apenas uma unidade central. Ainda por esse motivo, tratar possíveis erros se torna uma tarefa

mais fácil e rápida. Desta forma, é possível alcançar os objetivos de conforto e redução dos gastos energéticos com custos relativamente baixos.

O desenvolvimento de um SAR, seja ele centralizado ou distribuído, requer um projeto que considere: as necessidades do habitante, os subsistemas responsáveis por satisfazerem tais necessidades, os dispositivos e mecanismos utilizados para cada subsistema, de que forma esses subsistemas serão integrados, como os dados serão capturados e processados e como as informações obtidas possibilitarão o gerenciamento do ambiente de acordo com as necessidades do habitante. Para avaliar se esses sistemas satisfazem ou não as necessidades de um morador, existem basicamente duas formas.

De um lado, a constante interação entre o morador e o sistema permite que, ao longo do tempo, ajustes sejam feitos a fim de corrigir os mecanismos de controle, de forma que estes, da próxima vez, forneçam resultados mais eficazes.

Por outro lado, esta abordagem é um tanto quanto custosa do ponto de vista econômico e operacional, uma vez que os ajustes refletem, muitas vezes, na modificação de sensores, atuadores, circuitos, entre outros dispositivos presentes no sistema. Neste sentido, Bolzani [6] afirma que a utilização de ferramentas computacionais permitem avaliar a interação antes do sistema ser implantado. Deste modo, por meio de simulações, é possível verificar a aplicabilidade de técnicas ou mecanismos de controle, bem como identificar e corrigir os erros detectados, de forma que nenhum custo esteja associado ao processo, assim como nenhum inconveniente ao morador. Entretanto, essa abordagem será viável desde que a simulação corresponda ao ambiente real.

2.1.1 Sistemas de Supervisão e Aquisição de Dados - SCADA

Sistemas de Supervisão e Aquisição de Dados (SCADA) têm como principal objetivo o monitoramento do comportamento de um sistema de automação, seja ele residencial ou industrial, cujo funcionamento é geralmente comandado por controladores lógicos programáveis (CLP).

Inicialmente, os sistemas SCADA permitiam informar periodicamente o estado do processo industrial, monitorando sinais representativos de medidas e estados de dispositivos, através de um painel de lâmpadas e indicadores, sem qualquer interface gráfica com o operador.

Com a evolução tecnológica, os computadores assumiram um papel de gestão na captura e tratamento de dados, tornando possível a sua visualização num ambiente gráfico e a geração de comandos de programação para execução de funções de controle complexas.

Além de fornecer uma interface gráfica para visualização da dinâmica do processo, esses sistemas podem incluir várias funcionalidades como geração de histórico de dados, envio de comandos aos CLPs, gerenciamento de alarmes, comunicação com outros sistemas, entre outras. Para realizar essas funcionalidades, os sistemas SCADA precisam de informações provenientes dos processos e, para tanto, utilizam *tags*. As *tags* são nomes que associam um endereço ou registrador de um dispositivo utilizado em algum processo. Isto é, todos os valores dos dispositivos (sensores, detectores, etc.) que compõe um sistema de automação, sejam estes valores oriundos de um CLP ou de uma base de dados industrial (OPC), são tratados como *tags*. Desta forma é possível armazenar dados, monitorá-los, exibí-los das mais variadas formas, processá-los, entre outras funcionalidades fornecidas pelo SCADA. Para que a comunicação com um CLP seja realizada, é necessário a utilização de um driver de comunicação. Neste sentido, sistemas SCADA possuem um ambiente integrado de desenvolvimento que fornece recursos para a criação de gráficos, banco de dados, API's, drivers de comunicação, entre outros. A Figura 2.1 ilustra essa relação entre: o Sistema SCADA, o CLP e os dispositivos.

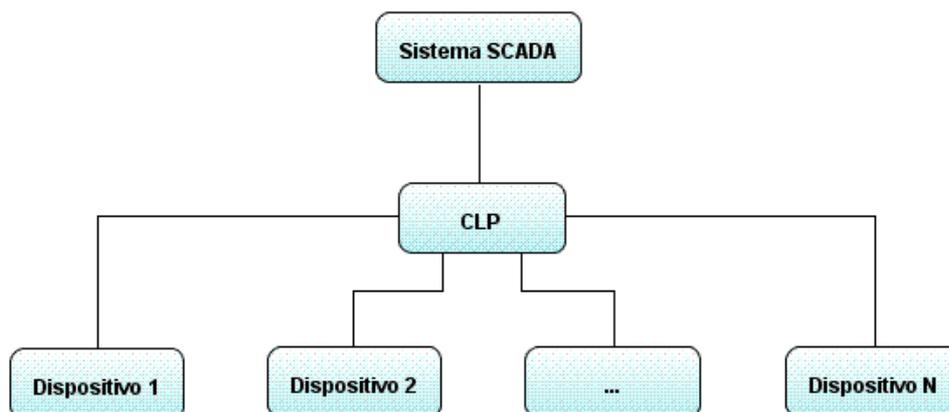


Figura 2.1: Organização dos Componentes Envolvidos nas Funcionalidades do SCADA, adaptada de Terezinho e Carvalho [31]

Um sistema SCADA sozinho, conforme a Figura 2.1, não realiza o controle em tempo real dos dispositivos. Este papel é de responsabilidade de um CLP. Portanto, um SCADA geralmente

se comunica com um CLP a fim de receber, exibir/monitorar, processar e enviar dados [31].

Em alguns casos, quando um novo mecanismo de controle está sendo avaliado, sistemas SCADA fornecem recursos para representar um processo industrial ou residencial sem a necessidade de dispositivos reais. Antes, esses processos e dispositivos podem ser simulados. Ou seja, nestes casos não existe nenhuma captura de valores de dispositivos mas sim, a geração destes valores pelo próprio ambiente de desenvolvimento fornecido pelo SCADA. No presente trabalho, o objetivo é avaliar um novo mecanismo de controle para redução de gastos energético. Desta forma, o software SCADA Eclipse [10] foi utilizado para construir um simulador a fim de testar e avaliar os resultados da técnica proposta.

2.2 Desenvolvimento de Sistemas de Automação Residencial Inteligentes - SARI

As tecnologias utilizadas em automação vêm avançando rapidamente. Um dos grandes marcos foi a substituição dos sistemas supervisórios que utilizavam hardware pelos que utilizam computadores digitais. A partir disso, não somente supervisão e monitoramento, mas mecanismos de controle passaram a ser implementados em computadores digitais. Os motivos pelos quais mecanismos de hardware foram substituídos por softwares são vários. Dentre eles, destacam-se: (i) possibilidade de executar tarefas complexas antes inviáveis por meio de hardwares; (ii) facilidade de manutenção dos mecanismos de controle; (iii) maior padronização nas interfaces de comunicação software-hardware e (iv) melhoria na confiabilidade no controle de processos [26].

Sob a ótica da domótica, satisfazer as necessidades de um morador não é uma tarefa trivial. Por esse motivo, foram introduzidos mecanismos de controle que utilizam softwares pois permitem a execução de tarefas complexas. Muitos dos trabalhos que abordam temas sobre domótica apresentam soluções que garantem conforto, segurança, gerenciamento de recursos, entre outros. Entretanto, a maioria deles apresentam, de acordo com [Bolzani, 2004], apenas mecanismos automáticos. Nos casos onde é considerado o comportamento humano, é necessário considerar características como capacidade de aprendizagem, habilidade proativa, multidisciplinaridade. Neste sentido, técnicas da Inteligência Artificial surgiram como forma de satisfazer tais considerações.

Ao se projetar um sistema domótico, faz-se necessário considerar a natureza do problema a ser tratado, de forma que seja possível avaliar se uma, mais de uma ou nenhuma técnica(s) da IA poderá(ão) ser aplicada(s). Em muitos casos onde se tem um modelo matemático, preciso e definido, é muito mais interessante aplicar técnicas convencionais. Por outro lado, se o problema envolve, por exemplo, o comportamento humano como critério para garantir conforto e economia de energia, então técnicas de IA são fortemente recomendadas [26].

Mesmo que técnicas de IA possam ser aplicadas, ainda existem alguns riscos. É o caso dos sistemas especialistas onde muitas vezes não é possível ter certeza de que a base de conhecimento estará correta ou completa. Na utilização de redes neurais também encontram-se dificuldades referentes ao tempo de resposta e convergência dos valores nos neurônios da camada de saída. Outro grande problema encontrado é a aplicação de técnicas que se adaptem ao comportamento humano. [25] e [29] apresentam soluções para esse problema. Contudo, é importante investigar a aplicação de técnicas da IA, ainda que a natureza do problema as recomende.

O presente trabalho propõe a utilização das técnicas utilizadas em Moll [25]. No entanto, diferentemente dos demais, este trabalho busca aplicar tais técnicas no gerenciamento do consumo energético residencial, de forma a reduzir o desperdício energético. Deste modo, faz-se necessário um novo estudo investigativo.

Capítulo 3

Inteligência Artificial e Sistemas de Automação Residencial

A IA, de acordo com McCarthy [22], é a ciência que estuda a forma de se construir máquinas inteligentes, principalmente softwares inteligentes. Trata-se da compreensão do comportamento humano por meio da utilização de máquinas. Russel e Norvig [27] dividem as linhas de pesquisa da IA em quatro categorias, a saber: (i) Sistemas que pensam como seres humanos, cujo objetivo é o de implementar máquinas capazes de pensar como um ser humano; (ii) Sistemas que pensam racionalmente, que buscam pensar e raciocinar de forma correta, sem se basear no pensamento humano, visto que este possui imperfeições; (iii) Sistemas que atuam como seres humanos, que se referem ao modo com que um ser humano age (comportamento) e não à sua forma de pensar; e (iv) Sistemas que atuam reacionalmente, que referenciam a forma correta de agir.

Esta última categoria traz o conceito, que será utilizado aqui, de agentes computacionais. Um agente é simplesmente algo que age. Entretanto, um agente computacional racional possui atributos que os distinguem de meros “programas”, tais como percepção do ambiente, adaptação a mudanças no ambiente, capacidade de assumir metas de outros agentes. Em outras palavras, um agente racional é aquele que age para alcançar o melhor resultado ou, em casos de incerteza, o melhor resultado esperado [27].

Com relação às formas de resolução de problemas, a IA se divide em duas grandes categorias: Os métodos utilizados para resolver problemas e o modo como as soluções são distribuídas espacialmente. De forma geral, a IA, quanto aos métodos de solução de problemas, é classificada em: IA Simbólica (IAS), IA Conexionista (IAC), IA Evolucionária (IAE) e IA Híbrida (IAH). Quanto a localização espacial tem-se a IA Monolítica (IAM) e a IA Distribuída (IAD)

[3].

A IAS, para manipular o conhecimento, utiliza as regras de inferência inspiradas nos silogismos propostos por Aristóteles, há mais de 2.000 anos. A IAC aplica-se a problemas mal definidos, mas que são conhecidos através de exemplos. Entre os campos de aplicações de técnicas conexionistas, das quais destacam-se as Redes Neurais Artificiais (RNA), citam-se, como exemplo: reconhecimento de padrões; controle de processos industriais; robótica; dentre outros. Outras técnicas de resolução de problemas podem ser encontradas na IAC, como é o caso do Raciocínio Baseado em Casos (RBC). Na IAE, os mecanismos utilizados são os mesmos encontrados na evolução biológica. É um exemplo de solução de problema bem definido de sobrevivência de uma espécie em ambiente variável. Pode ser encarada como um método de otimização com restrições variáveis e muitas vezes desconhecidas. A IAH reúne vantagens de mais de um tipo de método de abordagem para a resolução de problemas. A IAM é o modo de implementar soluções de IA como uma unidade indivisível. Muitas vezes são sistemas simples sem modularidade, como é o caso de muitos sistemas especialistas que utilizam uma base de conhecimento única e podem funcionar em um computador. O funcionamento da IAD depende de um determinado conjunto de partes (ou módulos), que funcionam de modo relativamente independente, para resolver de modo cooperativo um determinado problema. A IAD, de acordo com Bittencourt [5], surge como uma abordagem alternativa, onde as técnicas tradicionais da IA, combinadas com as funcionalidades dos sistemas distribuídos, dão origem à noção de sociedade de agentes.

No contexto de SAR's, é impossível definir precisamente o comportamento humano, o que caracteriza este trabalho como um problema mal definido, onde o objetivo é representar, computacionalmente, os módulos de um SAR. Desta forma, é de interesse deste trabalho, quanto à localização espacial, a IAD e, quanto ao método de resolução de problemas, a IAC.

Com o objetivo de tratar todos os aspectos que interessam a este trabalho, este capítulo foi dividido em quatro seções. A Seção 3.1 trata dos conceitos de Agentes Computacionais. A Seção 3.2 apresenta conceitos, sob a ótica da IAD, sobre a distribuição desses agentes. Nele, serão apresentadas duas maneiras de distribuir, espacialmente, sistemas inteligentes. Devido ao sistema, que aqui se propõe, tratar-se da implementação de Agentes Computacionais, a distribuição será realizada por meio da técnica de Sistemas Multiagente, cujos conceitos e a aplicabilidade serão

abordados na Seção 3.3. Por fim, a Seção 3.4 apresenta o funcionamento e a aplicabilidade do algoritmo RBC, método da IAC.

3.1 Agentes Computacionais

Atualmente, não existe uma definição universalmente aceita do termo agente. Permanece uma idéia comum de que autonomia é uma noção central para agentes, mas não muito além disso. Parte da dificuldade da noção de agentes deve-se às várias propriedades de agentes terem diferentes importâncias em diferentes domínios. Assim, para algumas aplicações, a habilidade dos agentes de aprender a partir de suas experiências é muito importante; para outras aplicações, aprendizagem não é somente desimportante, é indesejável. Para agentes, Wooldridge [35] utiliza a seguinte definição: “Um agente é um sistema de computador que é situado em algum ambiente, e que é capaz de realizar ações autônomas neste, para alcançar seus objetivos de projeto”. Já Russel e Norvig [27] definem agente como qualquer coisa que pode perceber seu ambiente através de sensores e agir nesse ambiente por meio de atuadores. Essas duas definições levam à representação da Figura 3.1.

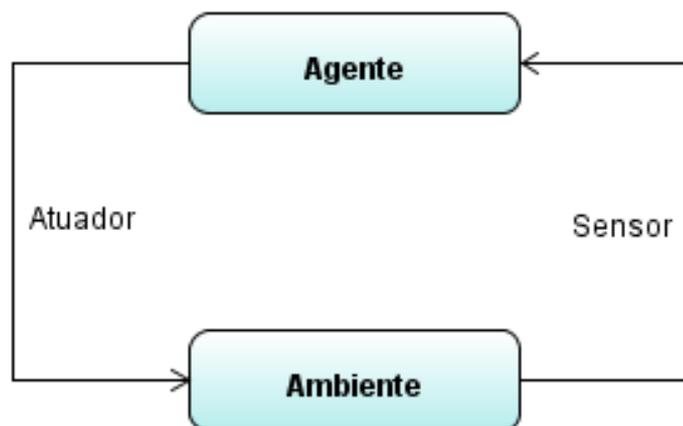


Figura 3.1: Interação entre Agente e Ambiente, adaptada de Wooldridge [35]

De forma geral, é possível compreender que um agente possui três conceitos essenciais. O primeiro se refere ao ambiente em que o agente está inserido. Entende-se por ambiente um lugar ou espaço que o agente ocupa a fim de interagir sobre ele por meio de sensores e atuadores.

O segundo conceito refere-se à autonomia. Neste contexto, autonomia trata-se da capaci-

dade de um agente atuar em um determinado ambiente sem a intervenção humana. Isto é, só o agente possui o controle de suas ações e de seu estado interno. Conceitualmente, a autonomia pode ser dividida em: (i) Autonomia Absoluta, quanto menos previsível o agente, mais autônomo ele parece ser; (ii) Autonomia Social, um agente sabe sobre os seus colegas e é sociável; (iii) Autonomia de Interface: onde a autonomia absoluta não é viável, pode-se ter autonomia com relação ao projeto interno das funções do agente; (iv) Autonomia de Execução: é a liberdade que um agente possui enquanto está executando tarefas em um ambiente; (v) Autonomia de Projeto (design): é a autonomia do ponto de vista dos projetistas [17].

O terceiro conceito, envolve a flexibilidade na execução de ações. A flexibilidade, de acordo com Jennings [17], implica em algumas propriedades que um agente pode possuir. Se um agente for reativo, ele deverá perceber seu ambiente e responder oportunamente às mudanças que nele ocorram. Por outro lado, se um agente for pró-ativo, ou seja possui meta, ele não deverá simplesmente atuar em resposta ao ambiente. Antes, deverá exibir um comportamento oportunista e direcionado ao seu objetivo tomando a iniciativa quando apropriado.

Para o caso de SAR, agentes possuem sensores para capturar dados do ambiente (residência) e produzem, como saída, um sinal que indica que, por exemplo, se a temperatura está baixa ou se o consumo está elevado. As ações a serem executadas podem ser a de ligar ou desligar um determinado dispositivo. A ação de ligar um dispositivo de condicionamento de ar, por exemplo, geralmente terá o efeito de diminuir a temperatura da sala, mas isso pode elevar o consumo energético da residência.

Para esclarecer os conceitos relacionados com os tipos de agentes e de ambientes, as próximas Seções abordam exclusivamente a arquitetura dos agentes e os tipos de ambientes existentes.

3.1.1 Arquitetura dos Agentes

Existem diversas classificações possíveis para o universo dos agentes. Esta seção apresentará algumas arquiteturas de agentes com base em duas classificações abordadas por Müller em [23]. Segundo Müller, o ponto central de um agente autônomo é a sua arquitetura de controle, isto é, a descrição de seus módulos e de como eles trabalham juntos. Segundo ele, todas as inúmeras arquiteturas propostas possuem a influência de três correntes: agentes reativos, agentes

deliberativos e agentes interativos. A classificação da arquitetura de software que implementa o paradigma de agentes computacionais depende em grande parte do ambiente em que este(s) agente(s) será(ão) executado(s). A seguir temos uma breve descrição destas três arquiteturas principais de agentes:

- **Agentes puramente reativos:** um agente é puramente reativo, se ele decide que ação executar com base no estado atual do ambiente. Agentes desse tipo são construídos com base em diferentes comportamentos. Não possuem uma representação interna do mundo, ou no máximo uma representação muito simples, e provém forte acoplamento entre percepção e ação. Estes agentes tomam suas decisões em tempo de execução, normalmente baseadas em uma quantidade de informações limitada e com regras simples de situação-ação.
- **Agentes deliberativos:** são agentes com estado interno. Possuem uma representação simbólica do mundo em termos de categorias tais como crenças, objetivos ou intenções. Também possuem mecanismos de inferência lógica para tomar decisões baseadas em seu modelo de mundo. Apresentam como foco a propriedade de pró-atividade, isto é, a habilidade de agir de forma direcionada a objetivos. Tais agentes mantêm uma representação interna do seu mundo e existe um estado mental explícito que pode ser modificado por alguma forma de raciocínio simbólico. Sistemas de planejamento podem ser vistos como predecessores deste tipo de agente. Agentes deliberativos podem ser modelados em função de crenças, desejos e intenções.
- **Agentes interativos:** são capazes de coordenar suas atividades com as de outros agentes através de comunicação e negociação. Este tipo de agente possui alguma representação explícita de outros agentes e pode ser capaz de raciocinar sobre eles. A propriedade principal de um agente desse tipo é a habilidade de comportamento social cooperativo. A construção de agentes interativos deve tratar de aspectos de coordenação e cooperação entre os agentes inteligentes distribuídos.

A segunda classificação apresenta uma taxonomia para agentes inteligentes com base em diferentes áreas de aplicação. Esta classificação toma por base o Estado Material dos Agentes,

ou seja, se são implementados por software ou hardware, e o Modo de Interação do Agente com seu ambiente.

Quanto ao Estado Material dos Agentes, os implementados por software são programas que interagem com ambientes reais ou simulados. Já os agentes implementados por hardware são agentes físicos que percebem o ambiente através de sensores e alteram o mesmo por meio de atuadores. Um robô é um exemplo típico de implementação física de um agente, que, obviamente, também utilizam componentes de software.

Em relação ao Modo de Interação do Agente, tem-se os Agentes Autônomos, Multiagente e Agentes Assistentes.

Os Agentes Autônomos realizam seu trabalho independentemente de outros agentes. Já em uma interação Multiagente, um agente usa seu conhecimento sobre outros agentes para coordenar e planejar suas ações, ou para alcançar seus objetivos colaborativamente. Por fim, Agentes Assistentes possuem a capacidade de interação com outros agentes, sejam eles artificiais ou humanos. Tais agentes podem servir de auxílio em diversos domínios de aplicação.

É importante citar que algumas aplicações utilizam mais de um tipo de arquitetura de agentes. Se aplicado, por exemplo, ao contexto de automação residencial, alguns agentes são reativos pois apenas capturam dados. Esses dados são informados aos agentes deliberativos que se encarregam de tomar a decisão. Em relação ao modo de interação, os agentes que capturam os dados são considerados os Agentes Assistentes e, como um todo, a interação é considerada Multiagente.

3.1.2 Ambientes

Como mencionado anteriormente, um agente está sempre relacionado a um ambiente, seja para a captura de dados ou execução de tarefas. Portanto, esta seção descreve os diferentes tipos de ambientes propostos por Russel e Norvig em [27], a saber:

- **Acessível x Inacessível:** Um ambiente acessível é aquele em que o agente pode obter informações completas, acuradas e atualizadas sobre o estado do mesmo. Ambientes do mundo real (como o nosso mundo cotidiano e a Internet) não são acessíveis neste sentido.
- **Determinístico x Não Determinístico** Um ambiente determinístico é aquele em que cada ação do agente tem um único efeito garantido - não há incerteza sobre o estado resultante

da ação. Em ambientes não determinístico, as ações do agente podem falhar em alcançar o resultado esperado. Ambientes determinísticos são mais fáceis de programar, pois o agente não precisa avaliar se uma determinada ação terá o resultado esperado. Em particular, em ambientes determinísticos, o projetista do agente pode assumir que as ações do agente sempre terão sucesso.

- **Estático x Dinâmico x Semi-dinâmico:** Um ambiente estático é aquele que se mantém inalterado, exceto pelas ações do agente. Um ambiente dinâmico muda sem o controle do agente. O mundo real e a Internet são altamente dinâmicos. Um ambiente semi-dinâmico não muda, exceto pelas ações do agente, mas a performance do agente cai ao longo do tempo.
- **Discreto x Contínuo:** Um ambiente é discreto se existe um conjunto fixo e finito de ações, percepções e estados nele. Um jogo de xadrez é discreto, pois existe um conjunto finito (apesar de extenso) de estados.
- **Episódico x Não Episódico:** Em um ambiente episódico a experiência do agente é dividida em episódios atômicos. Cada episódio consiste da percepção do agente e realização de uma única ação. Crucialmente, o próximo episódio não depende das ações tomadas em episódios anteriores, e assim, as escolhas das ações dependem somente do próprio episódio. Por exemplo, um agente que deve apontar peças defeituosas em uma linha de montagem, baseia cada decisão somente na peça atual, sem se preocupar com decisões anteriores; além do mais, a decisão atual não afeta se a próxima peça será defeituosa ou não. Em ambientes não episódicos, a decisão atual pode afetar todas as futuras decisões. Ambientes episódicos são mais simples pois o agente não tem que pensar no futuro.
- **Único agente x Multiagente:** Ambientes que possuem apenas um agente ou mais de um agente.

O tipo do ambiente está diretamente relacionado com o domínio da aplicação de um agente. Para agentes reativos que apenas capturam dados, um ambiente pode ser considerado Acessível, Determinístico, Estático, Discreto e Episódico. Por outro lado, agentes responsáveis por tomar decisões precisam avaliar os resultados de sua solução. Isso caracteriza o ambiente como Não

Determinístico. Além disso, alguns desses agentes podem utilizar experiências passadas como critério de tomada de decisão, o que torna o ambiente, também, Não Episódico.

Internamente, os agentes podem estar distribuídos em um ambiente Multiagente onde todos interagem entre si. Neste contexto, a próxima seção apresenta as formas como se distribuem sistemas inteligentes baseados em agentes computacionais.

3.2 Inteligência Artificial Distribuída - IAD

A IAD é uma das áreas da IA que mais se desenvolveram nos últimos anos e apresenta um enorme potencial de aplicações. A IAD estuda o conhecimento e os métodos de raciocínio que podem ser necessários ou úteis para que agentes computacionais participem de sociedades de agentes. Em termos de técnicas computacionais, a IAD reúne as da área de sistemas distribuídos às de IA [5].

Segundo Bittencourt em [5], as razões para distribuir sistemas inteligentes, que implementam alguns domínios de aplicação, tais como controle de tráfego aéreo, distribuição de energia elétrica, controle ambiental, neste caso, residencial, são várias. As principais são:

- Melhorar a adaptabilidade, a confiabilidade e a autonomia do sistema;
- Reduzir os custos de desenvolvimento e manutenção;
- Aumentar a eficiência e a velocidade;
- Permitir a integração de sistemas inteligentes existentes de maneira a aumentar a capacidade de processamento e, principalmente, a eficiência na solução de problemas;
- Permitir a integração dos computadores nas redes de atividades humanas.

A literatura costuma dividir a IAD em duas grandes áreas: a Solução Distribuída de Problemas (SDP) e os Sistemas Multiagente (SMA).

A SDP tem como foco principal o problema, conforme a tradição na IAS. Seus objetivos são utilizar a capacidade de processamento e a robustez oferecidas pela tecnologia de redes para atacar problemas naturalmente distribuídos ou excessivamente complexos, como a decomposição e alocação de tarefas em uma rede de computadores ou o controle de tráfego aéreo. Para a SDP,

os agentes são pré-programados para cooperar, e seus métodos visam garantir que esta cooperação ocorra de maneira coerente, robusta e eficiente. A qualidade de um sistema de SDP é medida pelo desempenho global do sistema na solução de um problema específico.

Os agentes envolvidos em SDP são programados para cooperar, dividir tarefas, comunicar-se de maneira confiável, dentre outros. Entretanto, experiências em ciências sociais mostram que não é simples estabelecer tais propriedades em uma coleção de indivíduos. O estudo das pressuposições básicas sobre agentes que garantam a possibilidade de ação cooperativa em sociedade é o foco de estudo dos SMA, isto é, neste caso o foco das pesquisas é o agente. O estudo de SMA é naturalmente multidisciplinar e envolve conceitos provenientes de diversas disciplinas, por exemplo, economia, teoria de jogos, ciências sociais, etologia, etc. Dentre estes conceitos, o de agente racional, isto é, aquele que age no sentido de maximizar seus benefícios e minimizar suas perdas, apresenta especial interesse para a área de SMA.

Como o presente trabalho envolve aspectos multidisciplinares, optou-se pelo enfoque na implementação de Agentes Computacionais, bem como a distribuição destes por meio dos SMA.

3.3 Sistemas Multiagente - SMA

O paradigma de SMA, segundo Wooldridge [35], adota o conceito de agente para caracterizar uma unidade autônoma de resolução de problemas. Desta forma, pode-se desenvolver uma estratégia de solução por meio do agrupamento de agentes que trabalham cooperativamente, cada um deles resolvendo parte do problema.

Para Jennings [17], SMA também se referem à subárea da IAD que investiga o comportamento de um conjunto de agentes autônomos designados para resolver problemas que estão além das capacidades de um único agente resolver.

Conceitualmente, SMA são sistemas compostos por mais de um agente, que se comportam de forma autônoma, mas que, ao mesmo tempo, interagem com os outros agentes presentes no sistema. A estes agentes, são atribuídas duas características essenciais: (i) agir de forma autônoma, tomando decisões capazes de satisfazer seus objetivos; (ii) interagir com outros agentes utilizando protocolos de interação social inspirados nos humanos e incluindo pelo menos algumas das seguintes funcionalidades: coordenação, cooperação, competição e negociação.

Para que tais funcionalidades de interação sejam implementadas, é necessário que haja comunicação entre os agentes. A próxima seção aborda os conceitos relacionados com a comunicação existente em um SMA.

3.3.1 Comunicação em SMA

A comunicação entre entidades computacionais foi desde sempre considerada um dos problemas mais importantes das ciências da computação. No entanto, na área dos SMA, a comunicação é tratada a um nível mais alto do que nas outras áreas das ciências da computação. Neste caso, a comunicação utiliza linguagens de comunicação próximas das linguagens utilizadas por humanos [35].

Para compreender como a comunicação é realizada em uma sociedade de agentes, é necessário entender como a comunicação é tratada no Paradigma Orientado a Objetos (POO), onde ela é realizada por invocação de métodos. Assim, um objeto *o1* pode invocar um método *m1* de um objeto *o2*. Neste cenário, o objeto *o1* é o responsável por decidir se o método será executado ou não e quando executar. Em um cenário orientado a agentes, não existe o conceito de um agente *j* “invocando um método” em um agente *i*. Isto é devido ao agente *i* ser um agente autônomo, ou seja, ele tem controle tanto sobre seu estado quanto sobre seu comportamento. Não é garantido que o agente *i* irá executar uma ação a só porque o agente *j* quer. O agente *i* pode não estar interessado em executar a ação.

De modo geral, agentes não podem forçar outros agentes a executar uma ação, nem alterar o estado interno de outros agentes. O que eles podem fazer são ações comunicativas na tentativa de influenciar outros agentes. Por exemplo, um agente pode dizer “Está frio lá fora” de uma forma sincera. Esta ação pode ser uma tentativa de modificar as crenças de um outro agente, mas isso não é suficiente pois cada agente, neste caso, assim como um ser humano, tem controle sobre suas próprias crenças e desejos [35].

A estas ações comunicativas, dá-se o nome de atos da fala. A teoria de atos da fala, definida por Searle em [28], trata a comunicação como ação. É baseada na suposição de que frases são ditas por agentes apenas como ações, no amparo de suas intenções. A partir desta teoria, uma série de linguagens de comunicação foram desenvolvidas.

Linguagens de Comunicação em SMA

Existem basicamente dois problemas a serem enfrentados quando se trata da definição de uma linguagem de comunicação. O primeiro se refere à definição da sintaxe e ao formato da linguagem que deverá representar todas as ações de falas de um agente. O segundo problema está relacionado com a semântica destas representações. Isto é, não basta apenas representar os atos da fala, mas é crucial que as duas partes compartilhem um vocabulário de palavras e seus significados. A este vocabulário dá-se o nome de ontologia [4]. Brevemente, apresentam-se, a seguir, 3 das mais conhecidas linguagens de comunicação em SMA:

- ***Knowledge Query and Manipulation Language (KQML)***: é uma linguagem “externa” para comunicação entre agentes. Ela define um formato de “envelope” para mensagens, no qual um agente pode explicitamente tratar a ilocucionária força desejada para a mensagem. KQML se preocupou com a parte de conteúdo das mensagens [11].
- ***Knowledge Interchange Format (KIF)***: é uma linguagem explícita que tem como objetivo de permitir representações do conhecimento sobre algum domínio particular. Ela foi planejada primeiramente (apesar de não unicamente) para formar as partes de conteúdo de mensagens KQML [14].
- **FIPA ACL**: em 1995, a Foundation for Intelligent Physical Agents (FIPA) iniciou o desenvolvimento de padrões para sistemas de agentes, definindo a FIPA ACL - *Agent Communication Language*, superficialmente similar à KQML: definida como uma linguagem “externa” para mensagens, com 20 performativas para definição de interpretação desejada de mensagens, e não obriga qualquer linguagem específica para o conteúdo das mensagens [12].

De acordo com Wooldridge [35], a FIPA ACL possui uma semântica formal abrangente, algo que faltava na KQML, permitindo representar crenças e desejos incertos de agentes, como as ações que os agentes realizam. Por este motivo, o presente trabalho utilizará a FIPA ACL como linguagem de comunicação.

Não basta apenas uma linguagem para prover a comunicação, é necessário adotar alguma arquitetura de comunicação. A próxima seção aborda conceitos relacionados com os tipos de arquiteturas para SMA.

Arquiteturas de Comunicação

Esta seção toma por base as arquiteturas propostas por Huhns e Stephens em [16]. Para Huhns e Stephens, a comunicação entre os agentes pode assumir uma de duas arquiteturas básicas, a saber:

- **Comunicação Direta:** Os agentes tratam da sua própria comunicação sem intervenção de qualquer outro agente. Para tal, partilham especificações, enviando aos outros agentes as suas capacidades e/ou necessidades de forma a cada agente poder tomar individualmente as suas decisões relativas à comunicação. Neste tipo de arquitetura cada agente comunica diretamente com qualquer outro agente, sem qualquer intermediário. Um dos principais problemas que se coloca nesta arquitetura está relacionado com a inexistência de um elemento coordenador da comunicação, o que pode originar o bloqueio do sistema se, por exemplo, todos os agentes decidirem enviar mensagens ao mesmo tempo.
- **Comunicação Assistida:** Os agentes apóiam-se em agentes especiais designados, aqui, como agentes intermediadores, de forma a efetuarem a comunicação com os outros agentes. Nesta arquitetura a organização de agentes é do tipo sistema federado. Nestes casos, se um dado agente i desejar enviar uma mensagem a um outro agente j , terá primeiro de a enviar para o agente intermediador, que se encarregará de a reencaminhar ao seu destinatário. Esta arquitetura resolve parcialmente o problema da coordenação da comunicação e diminui consideravelmente a complexidade necessária aos agentes individuais na realização de comunicação. Os agentes não necessitam de armazenar informações detalhadas sobre todos os outros agentes e nem sequer necessitam de saber o seu endereço de forma a comunicarem com eles. Basta comunicar com o agente intermediador. No entanto, a existência do agente intermediador pode introduzir uma certa centralização no sistema e um estrangulamento (bottleneck) no sistema de comunicações. Se este agente deixar de funcionar, o sistema de comunicações deixa também de funcionar.

Entendidos os conceitos que envolvem a distribuição dos agentes em um sistema inteligente, a próxima seção parte para o mecanismo de controle de sistemas inteligentes, apresentando os conceitos do RBC, técnica da IAC.

3.4 Raciocínio Baseado em Casos

Em um SAR, onde existem exigências relacionadas ao conforto, a segurança, a comunicação e ao consumo energético, o ideal seria que o sistema pudesse se adaptar ao perfil do morador e às condições existentes no ambiente controlado [19]. Diferentemente de um controlador simples baseado em regras e pré-programado, com ações bem definidas, que não consideram as condições pré-existentes no ambiente, um sistema adaptativo, além de considerar as condições prevalentes no ambiente, também deve levar em consideração as exigências de seus usuários. Essas exigências podem ser explícitas ou implícitas.

Exigências explícitas podem ser entendidas como sendo aquelas em que o usuário especifica, através de uma espécie de conjunto de preferências. Por outro lado, as exigências implícitas podem referir-se à performance, estabilidade, rejeição de distúrbios (ruídos), robustez a incertezas no modelo, forma da resposta do sistema a entradas padrão, simplicidade de implementação, custo de operação, dentre outras [26].

Para atender as exigências de projeto, um SAR adaptativo precisa ser capaz de aprender com o ambiente e operar segundo as preferências do usuário. Assim, uma vez iniciada a operação do sistema, espera-se que ele seja robusto ao ponto de controlar o ambiente eficientemente.

De acordo com Wangenheim e Wangenheim [33], o RBC, nos últimos anos, surgiu como uma técnica poderosa para solução automática de problemas. É aplicável de forma simples e direta a um amplo espectro de tarefas, todas tipicamente relacionadas à IA.

Conceitualmente, o RBC é um enfoque para a solução de problemas e para o aprendizado baseado em experiência passada. A resolução de problemas ocorre no momento em que o RBC recupera e adapta experiências passadas, denominadas casos, armazenadas em uma base de casos. Desta forma, um novo problema é resolvido com base na adaptação de soluções de problemas similares já conhecidas [33].

Os elementos básicos de um sistema RBC, de acordo com Wangenheim e Wangenheim [33], são:

- **Representação do Conhecimento:** Em um sistema de RBC, o conhecimento é representado principalmente em forma de casos que descrevem experiências concretas. Entretanto, se for necessário, também outros tipos de conhecimento sobre o domínio de

aplicação podem ser armazenados em um sistema RBC (por exemplo, casos abstratos e generalizados, tipos de dados, modelos de objetos usados como informação).

- **Medida de Similaridade:** Um sistema RBC tem de ser capaz de encontrar um caso potencialmente similar para o problema atual na base de casos e reutilizar sua solução para a situação atual. É a medida de similaridade que viabiliza este processo.
- **Adaptação:** Situações passadas representadas como casos dificilmente serão idênticas às do problema atual. Sistemas RBC avançados têm mecanismos e conhecimento para adaptar os casos recuperados completamente, para verificar se satisfazem às características da situação presente.
- **Aprendizado:** Para que um sistema mantenha-se atualizado, sempre que um problema for resolvido com sucesso, este deve ser armazenado em uma base de casos para que seja utilizado, futuramente, como possível solução para novos problemas.

Assim, quando um novo problema é encontrado, o RBC relembra casos similares e adapta as soluções que funcionaram no passado ao problema corrente.

3.4.1 Representação do Conhecimento

A forma principal de representação de conhecimento no RBC são os casos. Um caso é a representação de um problema que foi total ou parcialmente solucionado. O caso representa, portanto, o conhecimento utilizado para solucionar um determinado problema.

Tecnicamente, um caso representa a descrição de um problema, bem como a solução utilizada durante a sua resolução, sendo visto como essa associação dos dois conjuntos de informações: descrição do problema e respectiva solução. Cada conjunto estrutura-se por meio de um ou mais ítems ou atributos. A Figura 3.2 mostra um exemplo de um caso.

Casos podem, por exemplo, representar:

- o conjunto dos sintomas de um paciente e os passos do tratamento médico aplicado;
- a descrição dos sintomas do defeito técnico apresentado por um equipamento (Exemplo da Figura 3.2) e da estratégia de conserto aplicada;

- os objetivos de um processo legal e sua respectiva jurisprudência;
- os requisitos para um prédio e sua respectiva planta de construção;
- a descrição de um pacote de viagem.

	Caso 1	Caso N	
P R O B L E M A	Problema:	Impressora não funciona	S I N T O M A S
	Modelo:	Matrix 200	
	Luz de Estado do Papel:	Apagada	
	Luz de Estado da Tinta Colorida:	Apagada	
	Luz de Estado da Tinta Preta:	Apagada	
A	Luz do Interruptor:	Acesa	
S O L U Ç Ã O	Diagnóstico:	Curto-circuito	
	Ação:	Trocar Fonte de Alimentação	

Figura 3.2: Exemplo de um Caso Simplificado de um Sistema de Conserto de Impressoras, Wangenheim e Wangenheim [33]

Um caso também pode conter outros itens, como os efeitos da aplicação da solução ou a justificativa para aquela solução e sua respectiva explicação. Pode ainda ser enriquecido por dados administrativos, como o número do caso, a data de sua criação ou o nome do engenheiro de conhecimento que o incorporou à base.

Para que estejam à disposição para serem reutilizados, casos são organizados e armazenados em uma base de casos (BC). Trata-se de um conjunto de casos apropriadamente organizados. Geralmente, uma base de casos contém experiências positivas descrevendo estratégias de solução que contribuíram com sucesso para resolver o problema descrito, de forma que possam ser reutilizadas. Experiências negativas, expressando tentativas frustradas de solução de um problema podem também ser armazenadas, com o objetivo de indicar problemas potenciais e prevenir a repetição de erros passados.

3.4.2 O Ciclo do RBC

O algoritmo RBC tem como objetivo buscar um caso já resolvido cuja descrição seja similar à descrição do problema atual e reutilizar a solução deste caso para resolver o problema atual. Esta similaridade é determinada matematicamente pelo uso de uma medida de similaridade.

Não existe um padrão que define as etapas a serem executadas pelo RBC. No entanto, o modelo mais aceito para o processo de RBC é o Ciclo de RBC proposto por [1], representado na Figura 3.3. Este ciclo engloba um ciclo de raciocínio contínuo composto por quatro tarefas principais:

- **Recuperação:** onde recupera-se o caso mais similar ou um conjunto de casos da biblioteca ou base de casos;
- **Reutilização:** em que reutilizam-se as informações e o conhecimento que resolveram um problema anteriormente, associando-se sua solução e adaptando-a se necessário, ao contexto do problema atual.
- **Revisão:** onde revisa-se (quando necessário) e testa-se a aplicabilidade da antiga solução para resolver o novo problema;
- **Retenção:** esta é a etapa em que o sistema aprende, por armazenar todas as partes consideradas úteis do novo problema, na biblioteca de casos.

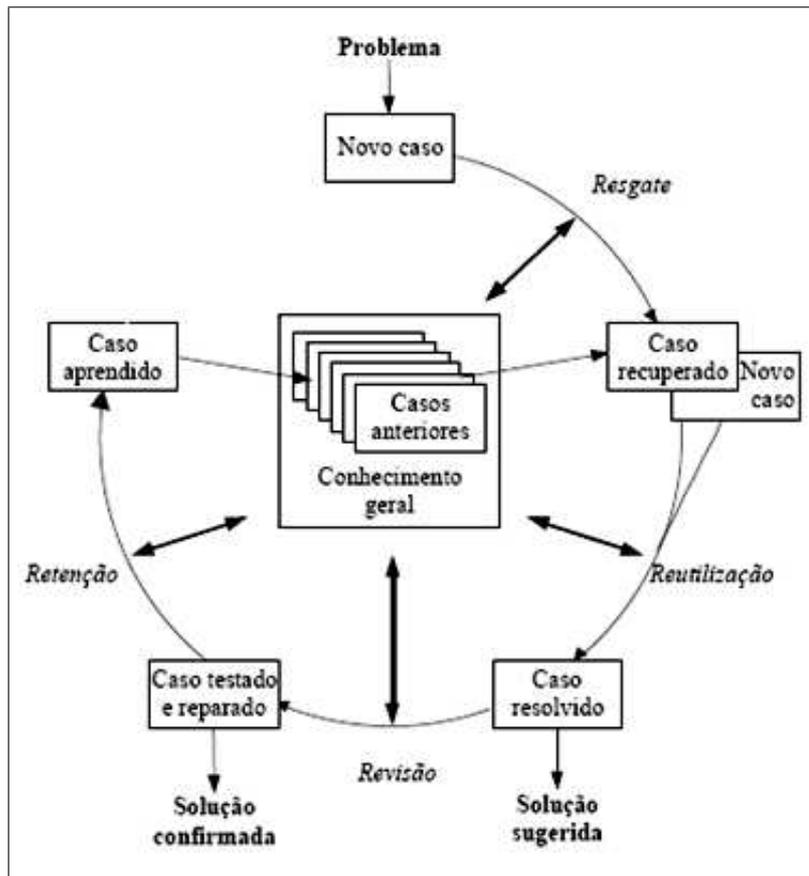


Figura 3.3: Ciclo Genérico de Processamento de um Sistema RBC, adaptada de Aamodt e Plaza [1]

3.4.3 Recuperação de Casos

O objetivo da recuperação de casos é encontrar um caso ou um pequeno conjunto de casos armazenados na base, que contenha uma solução útil para o problema ou situação atual. Portanto, a recuperação de casos adequados é uma das fases mais importantes e ao mesmo tempo mais custosas do RBC. Os algoritmos mais simples baseiam-se no exame exaustivo das características principais (atributos) dos casos e recuperam o caso com o maior número de características idênticas. Outros buscam o melhor caso, utilizando-se de heurísticas para reduzir e dirigir a busca. Entre esses pode-se citar: *busca serial*, *busca hierárquica* e *busca por simulação paralela* [18].

Durante a recuperação, é preciso realizar a comparação entre os casos contidos na base. Assim, para cada caso da base, um valor de similaridade é calculado, com a utilização de uma medida de similaridade. Este valor de similaridade indica o grau de semelhança entre o pro-

blema presente e um caso específico da base de casos. O valor de similaridade é expresso como número real entre 0.0 (nenhuma semelhança) e 1.0 (igualdade) e é calculado para cada caso na base conforme os valores dos atributos.

Os casos são eventualmente ordenados de acordo com a métrica ou algum critério de classificação, desta forma o caso que possui a mais forte sustentação de similaridade ao novo problema é escolhido. Dentre os principais métodos utilizados para realizar a medida de similaridade entre os casos, o presente trabalho considera: o método do vizinho mais próximo e o método da contagem de característica.

Métricas de Similaridade entre Casos

O conceito de similaridade, neste contexto, representa o quão útil é um caso específico para a resolução de um problema atual. Portanto, um caso é útil ou similar para a solução de um problema, quando for necessário modificar pouco ou nada daquele caso para adaptá-lo ao problema atual [33].

Similaridade entre dois objetos recebe o nome de similaridade global em RBC. Para determinar a utilidade de um caso em relação a um determinado problema, a similaridade global entre o caso e o problema deve ser determinada.

Método do vizinho mais próximo (*Nearest neighbour*)

O método do vizinho mais próximo determina a similaridade global entre um caso e um problema com base na similaridade local entre valores de um mesmo atributo e nos pesos dos diferentes atributos de um caso.

Como já mencionado, um caso possui uma série de atributos que representam o problema e a solução obtida para a resolução do problema. Assim, entende-se por similaridade local a proximidade entre os valores de um mesmo atributo. Se, por exemplo, um atributo de um caso armazena valores térmicos, então um atributo cujo valor seja 13° C é muito mais similar a um atributo que contenha o valor 14° C do que um atributo que contenha o valor 23° C. Já os pesos dos atributos representam a importância daquele atributo para um determinado problema. Portanto, a similaridade global, representada pela proximidade espacial entre um problema e um caso específico na base de casos é determinada com a utilização de uma medida de similaridade local entre os valores dos atributos do problema e do caso multiplicado pelo peso daquele de-

terminado atributo. Ao final, o resultado, para cada atributo, é somado. Matematicamente, o método descreve-se através da seguinte equação:

$$\text{Similaridade}(P, C) = \sum_{i=1}^n f(P_i, C_i) \times W_i$$

Onde:

P é o caso de entrada (problema)

C é o caso da base

n é o número de atributos de cada caso

i é o i ésimo atributo do caso

f é a função de similaridade local para o atributo i nos casos P e C

W é o peso dado ao atributo i

Similaridades são geralmente normalizadas em uma faixa de 0 a 1, onde 0 é a dissimilaridade total e 1 a coincidência absoluta, ou através de porcentagens, onde 100% é um casamento exato. A normalização é realizada com a divisão do valor de similaridade pela soma total dos pesos dos índices:

$$\text{Similaridade}(P, C) = \frac{\sum_{i=1}^n f(P_i, C_i) \times W_i}{\sum_{i=1}^n W_i}$$

Método da Contagem de Características (*Features Count*)

Neste método, o cálculo da similaridade consiste da contagem dos atributos do caso armazenado na base que coincidem com os atributos do caso de entrada. A cada atributo i do caso armazenado na base igual ao atributo i do caso de entrada (problema), soma-se 1 ao número de atributos coincidentes entre os dois casos. O grau de similaridade entre o caso de entrada e o caso armazenado na base é dado pelo percentual do número de atributos coincidentes entre os dois casos em função do número de atributos considerados para a comparação destes.

3.4.4 Reutilização de Casos

Após a recuperação de um caso considerado o mais adequado e similar, a solução utilizada por este caso é reutilizada para tentar resolver o problema atual. Durante este passo, a reutilização ocorre pela transferência de conhecimento (a descrição da solução) do caso anterior para o caso atual, ainda não solucionado [33].

A reutilização consiste principalmente da adaptação da solução do caso anterior ao caso atual, e as técnicas tratadas na reutilização de casos tentam resolver os problemas envolvidos na adaptação de casos, que são: quais aspectos da situação devem ser adaptados, quais modificações devem ser realizadas para esta adaptação, que método aplicar para realizar a adaptação e como controlar este processo.

3.4.5 Revisão de Casos

A revisão representa a confirmação de uma solução plausível para um problema. Esta fase envolve uma avaliação da qualidade da solução construída. Reparos na nova solução podem ser feitos, utilizando-se conhecimento específico do domínio. Com a aplicação do processo de revisão, a solução que foi adaptada é validada e, corrigida, caso seja necessário. A revisão, de acordo com Wangenheim e Wangenheim [33], consiste em duas tarefas:

1. Avaliar criteriosamente a solução gerada pelo reuso. Se for considerada como correta, aprenda com o sucesso e continue com a **retenção** do novo caso na base de casos.
2. Caso contrário, reparar a solução para o caso, utilizando conhecimento específico sobre o domínio de aplicação ou informações fornecidas pelo usuário.

3.4.6 Retenção de Casos

Retenção de casos é o processo de incorporação de uma nova experiência ou solução de um novo problema. O objetivo é o de constantemente atualizar e estender a base de casos. Isto permite a um sistema de RBC continuamente incrementar seu conhecimento e tornar-se um solucionador de problemas mais poderoso, com o passar do tempo e sua atualização [33].

Após resolver um problema, um novo caso é armazenado na base de casos (ou memória de casos) para futuramente ser utilizado. Nesta fase, é preciso adotar algum critério de seletivi-

dade dos casos a serem armazenados. Podem-se armazenar, por exemplo, somente casos que resultaram em sucesso, somente os que resultaram em falha, ou ainda, casos com ambos os resultados poderão ser acrescentados à base de casos.

Portanto, sempre que um problema é resolvido, a nova solução pode ser retida e integrada na base de casos, tornando-a imediatamente disponível para situações futuras. Desta maneira, o conhecimento presente em um sistema de RBC é continuamente atualizado, à medida que novas experiências dão origem a novos casos armazenados (aprendidos).

3.4.7 Aplicação do RBC

Sistemas de RBC já estão presentes em diversos tipos de aplicações como suporte ao consumidor, controle de qualidade, manutenção de aeronaves, planejamento de processos, apoio à decisão, roteamento de mensagens, help desks computadorizados, previsões, pesquisa jurisprudencial, dentre outros [8] [34] [21].

Quanto à área de automação residencial, foco deste trabalho, Moll realizou, em [25], um estudo referente a essa técnica, implementando um protótipo para o gerenciamento térmico de uma residência. Seguindo a mesma linha de raciocínio, um dos objetivos do presente trabalho é investigar a aplicabilidade do RBC no gerenciamento de consumo energético de uma residência. Esta técnica será utilizada como mecanismo de inferência e raciocínio de alguns agentes integrantes do sistema proposto, mais especificamente, os agentes de tarefas do sistema, apresentados no Capítulo 4 .

Capítulo 4

Modelagem do Sistema Proposto

O sistema que aqui se apresenta implementa uma arquitetura que toma por base o modelo de organização orientada a agentes computacionais onde cada agente é capaz de realizar funções de um subsistema de um SAR convencional. É importante considerar que este trabalho propõe, também, uma arquitetura escalável, onde mais módulos poderão ser agregados. O modelo ora proposto é a base para um modelo maior e mais complexo. Por este motivo é que se propõe a utilização de agente computacionais, uma vez que estes são flexíveis e recomendados para a realização de tarefas complexas. Como mecanismo de inferência e aprendizagem, o sistema utiliza o RBC.

Em termos estruturais, este capítulo se apresenta da seguinte forma: na Seção 4.1 apresenta-se a forma como o sistema captura os dados de entrada; a Seção 4.2 apresenta a organização dos agentes no modelo, descrevendo suas funções e a forma como cada um processa os dados de entrada; a Seção 4.3 trata das estruturas de dados utilizadas na comunicação dos agentes; por fim, a Seção 4.4 apresenta a arquitetura da linguagem de comunicação utilizada nas mensagens, bem como as regras de comportamento dos agentes, utilizadas para o tratamento das mensagens e tomadas de decisão.

4.1 Dados de Entrada

Os agentes envolvidos no sistema lidam com três tipos de dados diferentes, cujas origens são: (i) dados capturados de um simulador residencial; (ii) dados externos, no caso específico de controle de consumo energético, necessários para a atualização das tarifas de consumo e (iii) dados informados pelo morador, que vão permitir ao sistema traçar um perfil de suas preferências,

e a partir do uso RBC, ajustar o ambiente às condições que ele julga confortáveis.

No caso específico da redução de gastos energéticos, os dados do tipo (i) referem-se às informações coletadas dos equipamentos em funcionamento no ambiente simulado (televisão, geladeira, lâmpadas, ar condicionado, etc.), as taxas de consumo, apuradas em kWh (Quilowatt-hora), e o tempo de uso (acumulado) desses equipamentos. As tarifas estabelecidas pela COPEL (Companhia Paranaense de Energia) são dados do tipo (ii).

Quanto aos dados do tipo (iii), o sistema conta com as seguintes informações:

- valor (em R\$) que representa a quantidade máxima que o morador poderá despender;
- tempo (em dias) que representa o tempo máximo que o morador pensa em concentrar seu consumo;
- nível de interação do sistema:
 - nível baixo: o sistema envia apenas mensagens na tela, apontando problemas no consumo energético;
 - nível médio: o sistema desativa os equipamentos, respeitando as preferências do morador;
 - nível alto: o sistema desativa os equipamentos quando julgar necessário, ignorando, em casos críticos, as preferências do morador;
- preferência de uso dos equipamentos: para cada equipamento da casa, o sistema permite que o morador informe um valor, em um intervalo de 0 a 1, onde 0 representa que o morador não possui preferência por aquele equipamento e 1, representando que o equipamento é sempre utilizado, como é o caso de uma geladeira.

Referente ao processamento dos dados, com a utilização de um software simulador, os dados, que deveriam ser capturados por meio de sensores instalados em uma residência, podem ser simulados e capturados, sem a necessidade de um CLP. Entretanto, os valores gerados, diferentemente de uma situação real, são capturados sem nenhum tipo de ruído (interferência na rede elétrica). Contudo, estas características não farão diferença drástica no controle, uma vez

que ruídos normalmente causam diferenças de tensão por tempos muito curto e raramente prejudicam o funcionamento de um equipamento. Quanto aos dados dos tipos (ii) e (iii), o processamento é realizado de forma convencional; ou seja, uma vez capturados, estes são armazenados em bases de dados específicas, cujos acessos podem estar restritos ou compartilhados entre os agentes, conforme a situação.

4.2 Topologia do Sistema

O sistema conta com quatro categorias diferentes de agentes: (i) Agente Intermediário; (ii) Agente de Interface; (iii) Agente de Tarefas; (iv) Agentes de Informação. A Figura 4.1 ilustra o modo como os agentes estão interligados.

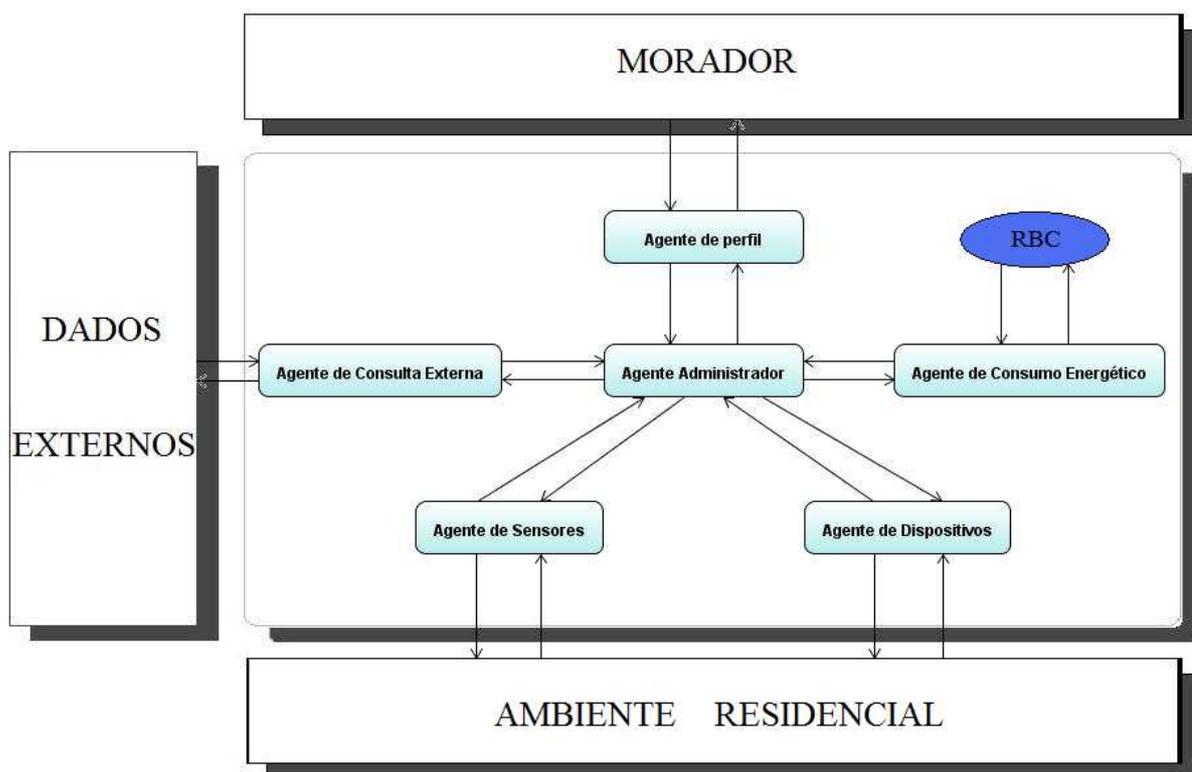


Figura 4.1: Modelagem da Arquitetura Proposta

O Agente do tipo Intermediário é identificado pelo Agente Administrador. É o agente responsável pelo controle geral do sistema. Tem como funções: estabelecer a comunicação entre todos os agentes; monitorar o ambiente e verificar as tarefas em execução.

O Agente de Perfil é um agente de Interface, cabe a ele repassar para o sistema as informações obtidas do exterior, bem como dar apoio ao usuário quanto ao funcionamento da aplicação [15]. O Agente de Perfil ficará encarregado de gerenciar as informações passadas pelo morador da casa. Tais informações englobam os dados do tipo (iii), descritos na Seção 4.1. Por meio destas informações, o sistema é capaz de traçar um perfil, que representa as preferências do morador. Estas preferências são utilizadas pelo sistema como critério decisivo para o controle do ambiente.

Como Agentes de Informação, encontram-se os de Consulta Externa e de Sensores. Segundo Brenner *et al.* [7], são os responsáveis tanto pelo acesso e gerenciamento de todas as informações disponíveis no ambiente, quanto pelo fornecimento de acesso inteligente a uma coleção heterogênea de fontes de informação e/ou conhecimento, organizando-as de modo que facilite a extração do conhecimento necessário para o processamento.

O Agente de Consulta Externa, neste caso, tem por função identificar e recuperar, periodicamente, em sites de concessionárias de energia elétrica, informações referentes às tarifas de consumo. O Agente de Sensores, por sua vez, é responsável por capturar os dados referentes aos estados dos equipamentos em funcionamento, as taxas de consumo e o tempo de uso acumulado desses equipamentos, bem como o local onde o morador se encontra. Todas essas informações são necessárias para o planejamento futuro e permitem a tomada de decisão dos agentes de Tarefas.

Os Agentes de Tarefas são os responsáveis pela execução de tarefas específicas no ambiente. Nesta categoria, encontram-se o Agente de Dispositivos, responsável por tarefas que vão desde ligar ou desligar determinados equipamentos da residência, até monitorar seu tempo de funcionamento, e o Agente de Consumo Energético, responsável pelo controle do ambiente.

Para controlar o ambiente, o Agente de Consumo Energético utiliza um controlador híbrido onde parte dos problemas são resolvidos com regras pré-definidas e a outra parte, resolvidos pelo algoritmo RBC. Para indentificar qual das técnicas será utilizada na resolução de um problema, o agente faz uma comparação entre os dados do ambiente, fornecidos pelos Agente de Sensores, e as preferências do morador, fornecidas pelo Agente de Perfil.

Os agentes do sistema trabalharão conjuntamente para alcançar objetivos comuns. O objetivo principal é que todos possam realizar atividades para a redução do consumo de energia, ao

mesmo tempo em que não comprometam o conforto e a segurança dos moradores da residência. Este objetivo é alcançado com base na interação entre os agentes.

Tal interação ocorrerá através da comunicação com o Agente Administrador. Em outras palavras, toda a comunicação será mediada pelo Agente Administrador, de modo que somente ele terá pleno conhecimento as capacidades dos demais agentes do sistema. Isto caracteriza uma comunicação assistida, mencionada na Seção 3.3.1. As próximas seções apresentam as estruturas utilizadas na comunicação, bem como o funcionamento do seu protocolo.

4.3 A Estrutura de Comunicação dos Agentes

Para que subsistemas de um SAR, dependentes um do outro, troquem dados ou informações, é necessário que haja algum tipo de comunicação, seja ela por rede sem fio, cabos, etc. Em um Sistema Multiagente, vários agentes podem ser definidos, de forma a representar os subsistemas de um SAR e, de forma similar, também necessitam de algum tipo de comunicação.

A comunicação fornece a base necessária para a realização da cooperação entre múltiplos agentes e permite a exploração em comum dos seus recursos e conhecimentos próprios, tornando possível o trabalho em paralelo de diferentes partes do problema, e a obtenção mais rápida da resolução do problema.

Para o sistema proposto, optou-se utilizar uma comunicação baseada em troca de mensagens entre os agentes, uma vez que estes agentes já são de conhecimento geral. Desta forma, evita-se que os agentes tenham de buscar mensagens em uma central de informações. Antes, eles podem comunicar-se de forma direta, de modo que cada um tenha um servidor de mensagens. A Figura 4.2 ilustra a estrutura de um agente, considerando aspectos referentes à comunicação utilizada no sistema.

Um agente, como já mencionado, para interagir com um ambiente, necessita de sensores e atuadores. Na Figura 4.2, o Sensor é responsável pela captura dos dados de entrada descritos anteriormente, bem como das mensagens provenientes da comunicação de outros agentes. Após essa captura é que o Núcleo de Processamento realiza a análise dos dados. A decisão deste processamento será efetivada com a utilização do Atuador que poderá tanto atuar diretamente com dispositivos da casa (Dados de Saída da Figura 4.2) quanto utilizar o Cliente de Comunicação para realizar a comunicação com um determinado agente. Esta comunicação, por exemplo,

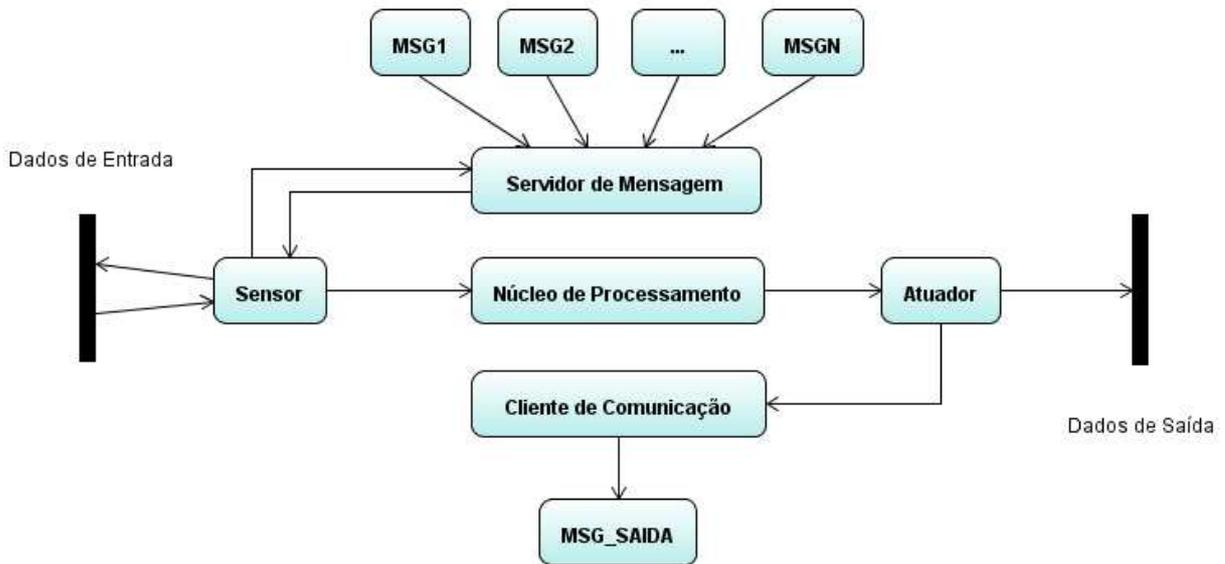


Figura 4.2: Estrutura de um Agente no Sistema Proposto

pode ser uma mensagem de resposta à uma mensagem anteriormente capturada (pelo Sensor) e processada (pelo Núcleo de Processamento).

Referente à troca de mensagens entre os agentes, a comunicação é realizada por meio do Protocolo TCP/IP com a utilização de *Sockets*. Esta abordagem, além de permitir que os agentes trabalhem em máquinas diferentes, fornece uma conexão segura e, portanto, essencial para um sistema de domínio crítico como o de controle de consumo energético residencial [20].

Para cada agente, é utilizado um *Socket* Servidor para receber as conexões de um ou mais agentes e um *Socket* Cliente responsável por criar a conexão com outro agente. Tais conexões permitem que mensagens sejam trocadas. Esta representação pode ser vista na Figura 4.2 através do Servidor de Mensagens e do Cliente de Comunicação.

O Servidor de Mensagens é capaz de receber mais de uma conexão ao mesmo tempo. Em outras palavras, o *Socket* Servidor, quando recebe uma conexão, cria um processo fraco (*Thread*) para tratá-la. Cada uma dessas *Threads* retira a mensagem, enfileira em uma estrutura de dados que será utilizada para posterior processamento e, por fim, finaliza a conexão. Paralelamente a isso, o *Socket* Servidor está esperando por novas conexões. Desta maneira, o fluxo e o tráfego da rede é reduzido, de maneira que a comunicação seja realizada de forma eficaz.

As mensagens enfileiradas são processadas utilizando a política PCPS (Primeiro a Chegar

é o Primeiro a Sair). Portanto, o Sensor captura as mensagens e envia ao Núcleo de Processamento à medida em que vão chegando. Como a comunicação envolve conexões em paralelo, neste caso, o Servidor de Mensagens utiliza um semáforo para controlar o acesso concorrente à estrutura de dados utilizada como fila das mensagens, uma vez que esta estrutura é compartilhada entre as *Threads* [30].

A próxima seção apresenta a linguagem utilizadas entre os agentes, bem como a forma de tratamento das mesmas.

4.4 A Linguagem de Comunicação entre os Agentes

Para que seja possível realizar a comunicação entre agentes em um Sistema Multiagente, além de utilizar uma estrutura que permita a comunicação, é necessário que os agentes utilizem uma linguagem para expressar suas intenções.

Como visto na Seção 3.3.1, alguns problemas são enfrentados na definição de uma linguagem de comunicação entre agentes. Devido a estes problemas, a FIPA [12], entidade internacional que define padrões relacionados à área de agentes computacionais, especificou uma linguagem de comunicação entre agentes (ACL), utilizada no sistema ora proposto, denominada FIPA ACL. A semântica desta ACL baseia-se na teoria dos atos da fala, definidas em [Searle, 1970], onde as mensagens estão associadas a atos que representam a vontade do agente sobre a informação contida na mensagem¹.

Para os agentes do sistema, são utilizados os atos performativos *inform*, *request* e *reply*. Por meio destes atos, por exemplo, o Agente de Sensores é capaz de, depois de capturar os dados, informá-los ao Agente Administrador que os informará ao Agente de Consumo Energético. Este último, por meio de algum mecanismo de controle, processa os dados recebidos e ao final pode requisitar ao Agente Administrador que desative um dispositivo específico ou informá-lo de que o ambiente continua sem alterações. O Agente Administrador, por fim, repassa esses informes ou requisições ao agente responsável e informa, por meio do *reply*, ao Agente de Consumo Energético, os resultados da requisição. A Figura 4.3 apresenta um exemplo de uma mensagem de requisição por meio do protocolo definido pela FIPA.

A Figura 4.3 representa o caso em que o Agente de Consumo Energético toma uma decisão

¹A lista destes atos podem ser encontrados em FIPA [13].

```

(Request
  :sender agenteConsumoEnergetico
  :receiver agenteAdministrador
  :ontology DesativarDispositivo
  :content (Desativar Ar Condicionado)
)

```

Figura 4.3: Exemplo de uma Mensagem no Formato da FIPA

e faz uma requisição ao Agente Administrador. Neste caso, o campo `sender` identifica o agente que está enviando a mensagem e `receiver`, o que está recebendo a mensagem. A semântica ou o significado da mensagem está relacionado com o desativamento de um dispositivo e é representado pelo campo `ontology`. O conteúdo que o agente receptor analisará está armazenado no campo `content`.

A forma como um agente trata uma mensagem é do tipo *WHEN-IF-THEN* [25]. O comportamento de um agente é implementado através de regras que seguem essa lógica. A cláusula *WHEN* representa uma situação ou um evento que aconteceu e o agente percebeu. Este evento pode ser uma mensagem recebida ou uma mudança no ambiente. A cláusula *IF*, por sua vez, compara este evento com os objetivos, intenções e desejos do agente. Esta comparação resulta em uma decisão tomada pelo agente. Tal decisão é representada pela cláusula *THEN*. A título de ilustração, a Figura 4.4 representa o formato destas regras.

```

NAME:
  Nome da Regra
WHEN:
  Evento ou Situação
IF:
  Estados Mentais (Desejos, Intenções, Crenças)
THEN:
  Ações Pessoais
  Atualizar Estados Mentais
  Ações de Comunicação

```

Figura 4.4: Formato de Regras de Comportamento de um Agente, adaptada de Moll [25]

Este formato, como explica Vinícius Moll em [25], facilita a especificação e a implementação das ações executadas por um agente. De fato, este formato, além de ser utilizado para o tratamento das mensagens, é utilizado para a construção das regras que resolverão parte dos problemas identificados pelo Agente de Consumo Energético. O próximo capítulo apresenta estes aspectos com mais detalhes.

Capítulo 5

Aspectos de Implementação

Como já mencionado no Capítulo 1, a proposta deste trabalho é modelar e implementar um sistema inteligente para o controle do consumo energético residencial. Portanto, este capítulo aborda as principais etapas do processo de implementação do sistema de controle, tais como a implementação do comportamento dos agentes do sistema, suas funcionalidades, o mecanismo de controle, o algoritmo RBC, utilizado como máquina de inferência do Agente de Consumo Energético e, por fim, a forma como o sistema de controle é alimentado.

5.1 Implementação dos Agentes

Conforme o modelo apresentado no Capítulo 4, temos na Figura 4.1, uma visão geral da arquitetura do sistema. Considerando o tempo e os esforços demandados para a implementação dos agentes do modelo, optou-se pela programação de apenas alguns agentes. Desde que o principal interesse consiste em verificar a aplicabilidade da técnica de RBC para a situação problema aqui trabalhada, essa redução não implica em prejuízos para o projeto. Desta maneira, são esses os agentes implementados:

Agente de Perfil - da categoria de agentes de Interface;

Agente Administrador - da categoria de agentes Intermediários;

Agente de Sensores - da categoria de agentes de Informação;

Agente de Consumo Energético - da categoria de agentes de Tarefas;

Agente de Dispositivos - da categoria de agentes de Tarefas;

A próxima seção descreve a implementação do comportamento e funcionalidades dos Agentes considerados neste trabalho. Neste contexto, em alguns trechos do texto será mencionado a captura de dados e a execução de tarefas no ambiente. Estes aspectos serão aprofundados na Seção 5.3.

5.1.1 Implementando o Agente de Perfil

A implementação do Agente de Perfil consistiu na criação de um gerenciador de informações do morador da residência. Internamente ao sistema, este agente está encarregado de manter o Agente Administrador informado das mudanças e da configuração inicial das informações do perfil do morador da residência. O diagrama de atividades mostrado na Figura abaixo representa o algoritmo que implementa este agente.

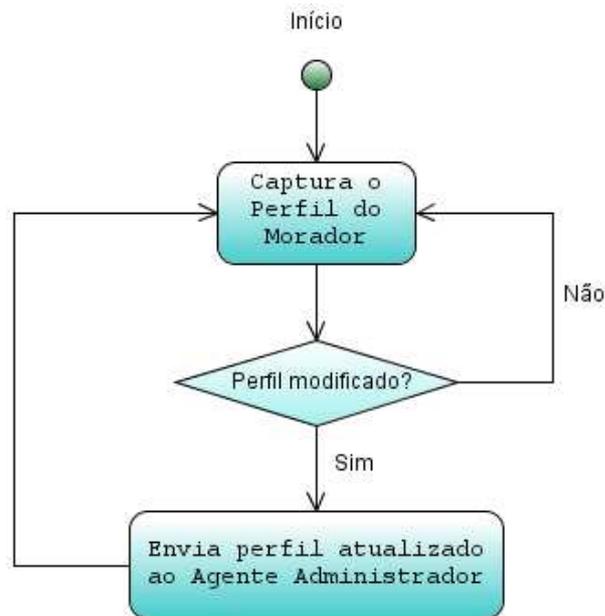


Figura 5.1: Fluxo de atividades do Agente de Perfil

5.1.2 Implementando o Agente Administrador

O Agente Administrador, como já mencionado no Capítulo 4, tem como objetivo a intermediação da comunicação entre os demais agentes do sistema. Basicamente, o algoritmo que implementa este agente pode ser visto no diagrama de atividades mostrado na Figura 5.2.

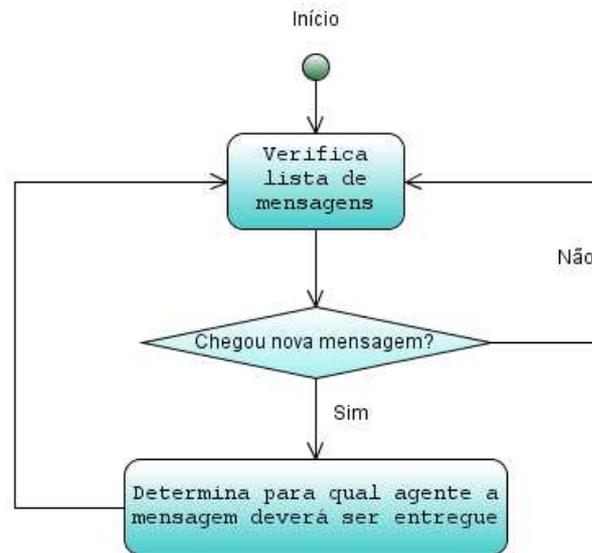


Figura 5.2: Fluxo de atividades do Agente Administrador

5.1.3 Implementando o Agente de Sensores

O Agente de Sensores é responsável por realizar a captura dos dados do ambiente, cuja função é manter o sistema de controle informado sobre as mudanças no ambiente simulado. Portanto, toda e qualquer mudança nos dados do ambiente é informada ao Agente Administrador. Estes dados são capturados por meio de arquivos que, como já mencionado, serão melhor entendidos na Seção 5.3. O diagrama de atividades mostrado na Figura 5.3 apresenta, de forma geral, o algoritmo que implementa este agente.



Figura 5.3: Fluxo de atividades do Agente de Sensores

5.1.4 Implementando o Agente de Dispositivos

O Agente de Dispositivos, quando em funcionamento, deverá realizar o desacionamento dos equipamentos controlados pelo sistema. Para tanto, deverá se comunicar diretamente com o ambiente por meio de arquivos. Esta comunicação será melhor compreendida na Seção 5.3. O diagrama de atividades da Figura 5.4 mostra, de modo geral, o algoritmo que implementa este agente.

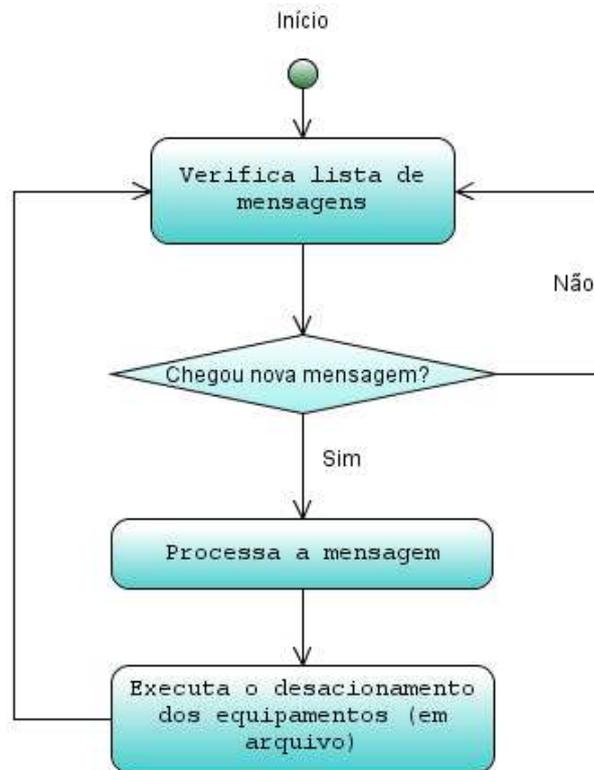


Figura 5.4: Fluxo de atividades do Agente de Dispositivos

5.1.5 Implementando o Agente de Consumo Energético

O Agente de Consumo Energético, como já sabido, é responsável pelo controle do ambiente. Para tanto, além de possuir um controlador híbrido com um módulo baseado em regras e outro que implementa o algoritmo de RBC, também é capaz de interagir com o agente administrador através da troca de mensagens. Este é considerado o agente com a lógica mais complexa. Esta característica pode ser observada no diagrama de atividades da Figura 5.5, que mostra o algoritmo que implementa este agente.

Através da Figura 5.5 é possível perceber que o Agente de Consumo Energético possui duas atividades principais. Primeiramente é verificado se existe alguma mensagem na fila de mensagens. Se existir, verifica-se quem foi o agente que a enviou e o conteúdo desta mensagem, de modo que este conteúdo, ao final, seja processado (Armazenar Perfil, Armazenar os Dados do Ambiente, etc.). Caso não exista uma mensagem na fila de mensagens ou se a mensagem existente já foi processada, o fluxo de atividades agora passa para o controle do ambiente. Como será visto, com mais detalhes, na Seção 5.2, o Agente de Consumo Energético trabalha com dois

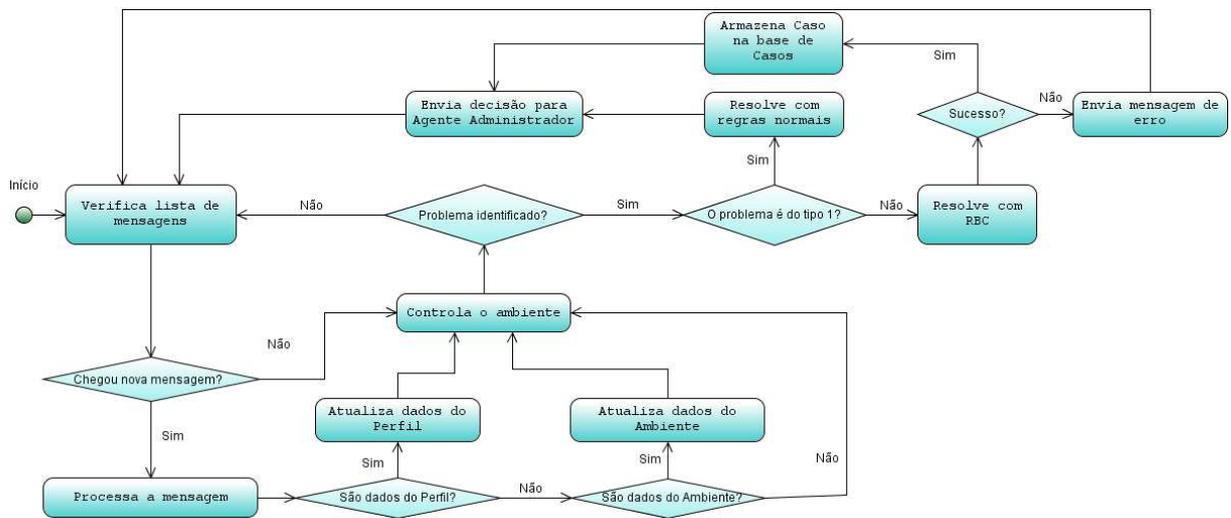


Figura 5.5: Fluxo de atividades do Agente de Consumo Energético

tipos de problemas, um deles é tratado com regras pré-definidas, o outro, pelo algoritmo RBC.

5.2 Implementando o Mecanismo de Controle do Sistema

Como já mencionado no Capítulo 4, o sistema lida com dois tipos de problema. O primeiro refere-se ao fato do morador esquecer algum dispositivo ligado, uma vez que ele não esteja no cômodo daquele determinado dispositivo. O segundo está relacionado com o consumo energético elevado. Respectivamente, o sistema rotula estes problemas como *Problema do Tipo 1* e *Problema do Tipo 2*. O *Problema do Tipo 1* é simples e não justifica a utilização de uma técnica de controle complexa. Portanto, a resolução deste tipo de problema é realizada por meio de regras pré-definidas. Já o *Problema do Tipo 2* envolve uma complexidade considerável e será tratado com a utilização do algoritmo RBC. Portanto, o mecanismo de controle adota uma abordagem híbrida, onde parte dos problemas serão resolvidos com regras pré-definidas e a outra parte com a utilização do algoritmo RBC.

5.2.1 Identificando um Problema

O Agente de Sensores estará, a cada instante, recolhendo dados do ambiente e informando-os para o Agente de Consumo Energético. Da mesma forma, o Agente de Perfil, a cada instante, verifica se os dados do morador foram modificados e, se houver alguma mudança, os envia

ao Agente de Consumo Energético. Desta maneira, o Agente de Consumo Energético realiza uma comparação entre as preferências do morador (dados fornecidos pelo Agente de Perfil) e os dados referente ao ambiente (fornecidos pelo Agente de Sensores). A identificação dos problemas ocorre nesta comparação.

Para identificar problemas do tipo 1, o sistema, a cada instante, verifica se o morador esqueceu algum dispositivo ligado em um cômodo onde o morador não esteja. Se isto for verdade, o problema será resolvido por regras pré-definidas que serão melhor compreendidas na Seção 5.2.2.

Para os problemas do tipo 2, o processo é um pouco diferente. Nestes casos, o sistema necessita de uma “função de gatilho”. Esta função representará o quão crítica é esta situação, se ela for problemática. Ela identifica se uma dada quantidade de energia, capturada pelo Agente de Sensores, consumida, em um determinado tempo, também capturado pelo Agente de Sensores, caracteriza um consumo elevado.

Tecnicamente, a função descreve-se da seguinte forma: o tempo máximo em que o morador pensa em consumir energia será 100% (normalizado para o valor 1) do tempo, bem como a quantidade máxima que ele pode consumir será 100% (normalizado para o valor 1) do consumo. Desta forma, a situação será identificada pela razão entre esses dois valores multiplicado, posteriormente, pela porcentagem de energia consumida. Sempre que o resultado da razão for menor ou igual a 1, a situação estará dentro do normal. Matematicamente, a função é descrita como:

$$f(x) = \frac{c^2}{t}$$

Onde:

c representa a porcentagem de energia consumida

t representa a porcentagem de tempo decorrido

A título de exemplo, considere que um morador tenha consumido 2% (0,02) do que poderia em 10% (0,1) do tempo em que pensou em consumir, portanto:

$$f(x) = \frac{0,02}{0,1} * 0,02 = 0,004$$

Se, por exemplo, o morador consumisse 50% em 20% do tempo, a função seria alterada para:

$$f(x) = \frac{0,5}{0,2} * 0,5 = 1,25$$

É importante considerar que o quadrado da quantidade de energia consumida (c^2) suaviza a identificação de um problema. Para melhor compreensão, considere um caso onde o morador mal começou a utilizar os dispositivos da casa e já acionou o chuveiro. Como o chuveiro consome muita energia, obviamente, dentro de 1min o consumo energético estará muito elevado em relação ao tempo decorrido. Um exemplo desta situação é quando o consumo atinge uma porcentagem de 0,01 e o tempo atinge uma de 0,001. Considere a função sem o quadrado do consumo, o resultado seria:

$$f(x) = \frac{0,01}{0,001} = 10$$

Isto caracteriza uma situação bastante crítica, quando, na verdade, não é, uma vez que o morador nem sequer atingiu 1% do que poderia consumir. Se o quadrado do consumo for considerado:

$$f(x) = \frac{0,01}{0,001} * 0,01 = 0,1$$

Desta forma, o sistema suaviza o controle de consumo energético.

Para situações onde o resultado for maior do que 1, um caso será criado e resolvido por meio do algoritmo RBC. Para efeitos de esclarecimento, o resultado desta função é chamado de “Grau da Problemática”. Esta nomenclatura será utilizada com bastante frequência nas próximas seções.

5.2.2 Resolução dos Problemas do Tipo 1

Como já citado, a resolução deste tipo de problema é feita por meio de regras pré-definidas. As regras são criadas no formato representado pela Figura 4.4 da Seção 4.4. A criação das regras toma por base as preferências definidas pelo morador. Mais especificamente, as regras respeitam as preferências de equipamentos, descritas, na Seção 4.1, como um dos atributos dos dados de entrada do tipo iii. Em outras palavras, o Agente de Consumo Energético, uma vez que

tenha identificado um problema do tipo 1, verifica, por meio dos dados do ambiente, fornecidos pelo Agente de Sensores, a existência de algum equipamento ligado em um cômodo em que o morador não se encontra. Isto é, se em um determinado cômodo existe algum equipamento ligado e o morador não se encontra neste cômodo, em seguida é verificado se algum desses equipamentos possuem preferência de uso. Se não tiverem, então são desligados.

5.2.3 Implementação do RBC para a Resolução dos Problemas do Tipo 2

Como visto na Seção 3.4, o RBC é um enfoque para a solução de problemas e para o aprendizado baseado em experiência passada. Como já mencionado, também, em várias seções deste texto, o principal foco deste trabalho é o gerenciamento do consumo energético de uma residência.

Em uma situação convencional, normalmente, uma pessoa utiliza suas experiências passadas para prever futuras situações. Isto vale também para o caso do consumo energético. O RBC, neste caso, será implementado com o objetivo de reduzir o desperdício energético sem abrir mão do conforto do morador e utilizará, para tanto, as experiências (casos) similares que já ocorreram.

Basicamente, os Agentes de Sensores e de Perfil, fornecerão os dados necessários para que o Agente de Consumo Energético seja capaz de traçar um perfil das preferências do morador e, com este perfil, identificar a ocorrência de um problema do tipo 2. Uma vez identificado, o Agente de Consumo Energético aciona o RBC, que por sua vez será responsável por resolver o problema e sugerir uma solução para o mesmo. Esta solução será executada pelo Agente de Dispositivo.

Antes de apresentar, com mais detalhes, o funcionamento do RBC, faz-se necessário uma descrição geral de como os problemas de tipo 2 serão representados.

Representação dos Casos

Os problemas são representados por uma série de atributos. Embora um caso seja a representação de um problema já solucionado, neste momento, uma vez que um problema do tipo 2 tenha sido identificado, este passará a ser denominado como “caso de entrada”. A Figura 5.6 mostra como os casos são representados.

CASO						PESO
Problemas/Sintomas	Diagnóstico/ Situação	Grau da Problemática	0 – 1000			5
		Cômodo	Cozinha	Sala	Quarto	Banheiro
		Nível de Interação	Baixo	Médio	Alto	5
	Estado dos Dispositivos	Luz – Sala	Ligado		Desligado	2
		Televisão – Sala	Ligado		Desligado	2
		Luz – Cozinha	Ligado		Desligado	2
		Geladeira – Cozinha	Ligado		Desligado	2
		Fogão – Cozinha	Ligado		Desligado	2
		Luz – Quarto	Ligado		Desligado	2
		Ar – Quarto	Ligado		Desligado	2
		Luz – Banheiro	Ligado		Desligado	2
		Chuveiro – Banheiro	Ligado		Desligado	2
	Preferência dos Dispositivos	Luz – Sala			0 ou 1	2
		Televisão – Sala			0 ou 1	2
		Luz – Cozinha			0 ou 1	2
		Geladeira – Cozinha			0 ou 1	2
		Fogão – Cozinha			0 ou 1	2
		Luz – Quarto			0 ou 1	2
		Ar – Quarto			0 ou 1	2
		Luz – Banheiro			0 ou 1	2
Chuveiro – Banheiro				0 ou 1	2	
Solução	Ação	Lista de dispositivos a serem desligados				
	Resultado	Fracasso		Sucesso		

Figura 5.6: Formulário de Casos do Sistema

Estruturalmente, na Figura 5.6, ao lado de cada atributo estão, de forma geral, os valores que os mesmos podem assumir. Alguns valores, como é o caso do atributo *ação*, não puderam ser descritos detalhadamente em função da quantidade de possibilidades existentes.

Os casos são divididos em duas seções. A primeira seção refere-se ao momento de criação de um caso. Dentro desta primeira seção encontram-se, subdivididas em 3 partes, informações que diferenciam um caso do outro. A primeira destas 3 partes representa a situação do caso, onde os atributos *Grau da Problemática*, *Cômodo* e *Nível de Interação* são informados. O atributo *Grau da Problemática* representa o quão crítica está a situação. Este atributo representa o resultado da “função de gatilho”, descrita na Seção 5.2.1. O atributo *Cômodo* representa o cômodo em que o morador se encontra no momento da ocorrência de um caso. Já o atributo *Nível de Interação* representa se, naquele momento, o morador gostaria que o sistema interagisse de maneira baixa, média ou alta. As outras duas partes representam, respectivamente, os estados dos equipamentos da residência e suas preferências.

A segunda seção representa a solução encontrada pelo RBC. Neste momento são informados a ação que o sistema deverá executar e o resultado da execução desta solução. O atributo *Ação* compreende a lista de dispositivos que o sistema desliga. Para garantir que o resultado seja satisfatório, o RBC passa por um processo de avaliação da decisão tomada. Tal processo será

descrito com mais detalhes na Seção 5.2.3.

Recuperando o Caso mais Similar

Uma vez que o Agente de Consumo Energético tenha identificado um problema do tipo 2, o RBC inicia a etapa de recuperação do caso mais similar. Trata-se de um processo cujo objetivo é encontrar um caso tão similar que seja possível reutilizar sua solução para resolver um problema (caso de entrada) novo.

Para que seja possível realizar o processo de recuperação, é necessário algum tipo de cálculo que compare a similaridade entre dois casos. Desta maneira, a abordagem adotada neste trabalho utiliza a técnica do vizinho mais próximo, descrita na Seção 3.4.3

Nesta técnica, para que um caso seja considerado similar, duas características são necessárias. A primeira requer a existência de similaridade entre os possíveis valores de um atributo, o que caracteriza a similaridade local entre dois atributos iguais. A segunda refere-se à distribuição de pesos entre os diferentes atributos do caso.

A similaridade local representa o quão similar é um valor de um atributo para outro valor do mesmo atributo. Para melhor compreensão, será descrito a seguir, as similaridades locais dos atributos considerados nos casos deste trabalho.

Primeiramente, apresenta-se a similaridade local do atributo *Grau da Problemática*. Tal similaridade é calculada a partir da seguinte equação:

$$SimLocal = 1 - \frac{Grau_j - Grau_i}{1000}$$

Onde:

$Grau_j$ representa o valor do atributo *Grau da Problemática* do caso j

$Grau_i$ representa o valor do atributo *Grau da Problemática* do caso i

Deste modo, quanto menor for a diferença entre os valores, maior será a similaridade entre os valores deste atributo. A título de exemplo, considere o valor do atributo *Grau da Problemática* de um caso como 10 e outro como 8. A similaridade local para este atributo será:

$$SimLocal = 1 - \frac{10 - 8}{1000} = 0,998$$

De forma parecida, os demais atributos também realizam seus cálculos de similaridade local. No entanto, para estes, não é necessário a utilização de uma equação. Todos eles possuem uma tabela pré-definida, apresentadas abaixo, que representa a similaridade local entre os valores destes atributos.

	<i>Cozinha</i>	<i>Sala</i>	<i>Banheiro</i>	<i>Quarto</i>	<i>Nenhum</i>
<i>Cozinha</i>	1.0	0	0	0	0
<i>Sala</i>	0	1.0	0	0	0
<i>Banheiro</i>	0	0	1.0	0	0
<i>Quarto</i>	0	0	0	1.0	0
<i>Nenhum</i>	0	0	0	0	1.0

Tabela 5.1: Similaridade local do atributo (*Cômodo*).

	<i>Baixo</i>	<i>Medio</i>	<i>Alto</i>
<i>Baixo</i>	1.0	0	0
<i>Medio</i>	0	1.0	0
<i>Alto</i>	0	0	1.0

Tabela 5.2: Similaridade local do atributo (*Nível de Interação*).

	<i>Ligado</i>	<i>Desligado</i>
<i>Ligado</i>	1.0	0
<i>Desligado</i>	0	1.0

Tabela 5.3: Similaridade local dos atributos que representam os estados dos dispositivos.

	<i>Com preferencia</i>	<i>Sem preferencia</i>
<i>Com preferencia</i>	1.0	0
<i>Sem preferencia</i>	0	1.0

Tabela 5.4: Similaridade local dos atributos que representam as preferências dos dispositivos.

Cada uma das tabelas representa um atributo. As linhas representam os possíveis valores daquele determinado atributo para um determinado caso. As colunas representam os possíveis valores deste atributo para um outro caso. É possível notar que só haverá similaridade, e esta será total, em situações onde os valores são os mesmos. Caso contrário, não haverá similaridade alguma.

Quanto aos pesos dos atributos, considerou-se, no presente trabalho, os representados na Figura 5.6. O atributo *Grau da Problemática* possui peso (importância) 5, assim como o atributo *Nível de Interação*. Os atributos que representam os estados de dispositivos e suas preferências, possuem peso 2. Como não existe um padrão de distribuição de pesos para o contexto de consumo energético, os valores citados foram estimados com base na responsabilidade dos atributos. Por exemplo, o atributo *Grau da Problemática* é considerado um importante critério que diferencia um caso do outro e, portanto, possui um peso maior. Já os atributos que representam os estados dos dispositivos, são utilizados, também, para diferenciar um caso do outro, mas são mais utilizados como informações para a tomada de decisão.

O motivo que justifica a inexistência de um peso para o atributo *Cômodo* é o fato dele ser considerado o elemento chave para decidir se um caso é minimamente similar a outro. Ou seja, um caso só pode ser similar a outro se, no mínimo, os dois tiverem o atributo *Cômodo* com o mesmo valor. É por meio deste atributo, também, que a recuperação se dá em duas etapas. A primeira etapa consiste em filtrar os casos onde o atributo *Cômodo* possui mesmo valor que o atributo *Cômodo* do novo caso. Na segunda etapa, utiliza-se a técnica do vizinho mais próximo, descrita na Seção 3.4.3.

Através deste cálculo é que se encontra a similaridade global entre dois casos. Resumindo, a similaridade global entre dois casos é calculada em função da similaridade local de cada atributo e seus respectivos pesos. O caso que possuir o maior valor de similaridade será o mais similar.

Por meio de testes no sistema, verificou-se que um caso é potencialmente similar a outro se, primeiro, possuírem valores iguais para o atributo *Cômodo* e, segundo, se o valor da similaridade global entre esses dois casos for maior que 0,9. É importante mencionar que existem três situações resultantes do processo de recuperação de casos, a saber: (i) o resultado pode ser um caso completamente similar (similaridade 1), (ii) potencialmente similar (similaridade maior que 0,9) ou (iii) a inexistência de um caso similar (similaridade menor que 0,9).

Após o processo de recuperação de casos, segue-se para a etapa de reutilização.

Reutilizando o Caso mais Similar

O processo de reutilização de um caso consiste em analisar as decisões tomadas pelo caso recuperado para resolver um caso de entrada. Além disso, as decisões tomadas dependem do atual nível de interação do sistema e das preferências do morador pelos equipamentos. Isto é, em situações onde o nível de interação é baixo e o caso de entrada é completamente similar ao caso recuperado, o RBC apenas informa ao morador, por meio de uma mensagem, os equipamentos que o caso recuperado desligou em uma situação passada. Em situações onde o nível de interação é médio ou alto, apenas executa-se a decisão tomada pelo caso recuperado, sem necessidade de realizar processamento algum.

Para os casos potencialmente similares, o RBC analisa o conjunto de ações executadas pelo caso recuperado. Para cada equipamento da lista (atributo *Ação*) é verificado se para o caso de entrada aquele equipamento pode ser desligado. Esta verificação é realizada por meio das preferências do morador em relação aos equipamentos. Em seguida, em situações onde o nível de interação é baixo, apenas envia-se uma mensagem ao morador sugerindo os equipamentos que devem ser desligados. Em situações onde o nível de interação é médio, desliga-se apenas os equipamentos que estejam na lista e que não possuam preferência. Nos casos onde o nível de interação é alto, além de desligar os dispositivos que estejam na lista e que não possuem preferência, o RBC, em um segundo momento, caso a decisão não tenha sido satisfatória e a situação esteja bastante crítica (*Grau da Problemática* maior que 500), ignora as preferências dos equipamentos, desligando os equipamentos que julgar necessário.

Em situações onde não foram encontrados casos similares, o Agente de Consumo Energético toma a decisão baseada apenas nas preferências do morador pelos equipamentos. Este processo pode ser, por vezes, demorado, pois um equipamento que o Agente de Consumo Energético julga ser desnecessário, depois de desligado, pode voltar a ser acionado pelo morador, fazendo com que o Agente de Consumo Energético tenha que reconsiderar as decisões até que nenhum equipamento seja acionado novamente ou até que a lista de equipamentos que podem ser desligados, chegue ao fim. Da mesma forma que os casos potencialmente similares, em situações onde o nível de interação é baixo, respeita-se as preferências do morador e apenas sugere-se, por meio de mensagens, equipamentos a serem desligados. Quando em nível de interação médio,

respeita-se as preferências e desliga-se os equipamentos sem preferência. Em nível de interação alto, ignora-se, para situações onde o *Grau da Problemática* é maior que 500, as preferências e desliga-se os equipamentos que o Agente de Consumo Energético julgar necessário.

Para que estas atividades sejam executadas satisfatoriamente, é necessário que o RBC esteja sempre revisando e avaliando as decisões tomadas. Este processo é descrito na próxima seção.

Revisando a Decisão Tomada

Depois de uma decisão ter sido tomada, independente se o caso reutilizado é completamente similar, potencialmente similar ou se não existiu algum caso similar, o Agente de Consumo Energético continua recebendo, por meio do Agente de Sensores, os dados do ambiente e, através destes dados, é capaz de julgar se a solução foi satisfatória ou não. Portanto, sempre que uma decisão é tomada, o Agente de Consumo Energético verifica se o *Grau da Problemática* está diminuindo. Se a tendência for diminuir, então significa que a solução está causando efeito positivo na redução do consumo energético.

Considerou-se duas formas para identificar se uma solução foi satisfatória ou não. Uma delas é perceber a primeira redução do *Grau da Problemática*. Isto é, se um problema foi identificado com o *Grau da Problemática* contendo o valor 10, mesmo que durante o controle este grau aumente para 12, a primeira redução termina o processo de controle, mesmo que a redução faça com que o *Grau da Problemática* caia para 11 (ainda maior do que o *Grau da Problemática* inicial). A outra abordagem é avaliar a solução até que o *Grau da Problemática* atinja seu valor inicial. Ou seja, se o problema foi identificado com o *Grau da Problemática* contendo o valor 10, a solução só será satisfatória se o valor final deste for igual ou menor que 10. A abordagem adotada neste trabalho foi a segunda, pois apresenta melhores resultados, ainda que consuma mais tempo de processamento.

Uma outra abordagem, considerada computacionalmente ideal, seria avaliar a solução até que o *Grau da Problemática* atingisse o valor 1. No entanto, esta abordagem interfere bastante no cotidiano do morador e, por vezes, pode causar insatisfação ao mesmo, pois força bastante o desligamento de equipamentos da residência.

Uma solução é considerada um fracasso quando o Agente de Consumo Energético percebe que o morador saiu do cômodo em que estava quando o caso foi identificado. Isto se justifica pois caracteriza um novo caso de entrada, uma vez que o atributo *Cômodo* é utilizado para

filtrar casos similares. Outra característica que faz da solução um fracasso é quando o *Grau da Problemática* não diminui ou a possibilidade de equipamentos a serem desligados se esgota.

Retenção de Casos

Uma vez que a solução de um caso tenha sido reutilizada e revisada, um caso de entrada só é retido na base de casos se sua solução for um sucesso. Caso contrário, o caso é desconsiderado. Desta forma, o RBC aprende apenas casos cujas soluções resultaram em sucesso.

5.3 Alimentação do Sistema

Para que o sistema de controle possa realizar o processamento dos dados, é necessário que este seja alimentado. Para tanto, contou-se com um ambiente simulado cujo objetivo é gerar os dados que deverão ser processados pelo sistema de controle a fim de investigar a aplicabilidade das técnicas consideradas para o desenvolvimento deste trabalho.

Este ambiente simulado consiste em uma série de equipamentos e sensores que serão monitorados e controlados pelos agentes do sistema. A Figura 5.7 apresenta a distribuição dos equipamentos por cômodos da residência, levando em consideração suas potências.

Quanto aos sensores, o Agente de Sensores necessita de 3:

- Sensor de Consumo: Representa a soma entre o consumo de cada equipamento.
- Sensor de Presença: Representa o local onde o morador se encontra em um determinado momento. Os possíveis locais são 5:
 - Cozinha
 - Sala
 - Quarto
 - Banheiro
 - Nenhum (Quando o morador está fora da casa ou em locais da casa onde não existem equipamentos, como um corredor).
- Sensor de Estados: Representa o estado de cada equipamento da casa. Isto é, se um determinado equipamento está ligado ou desligado.

Cômodo	Dispositivo	Potência (W)
Cozinha	Geladeira	400
	Forno Elétrico	2500
	Luz	60
Sala	Televisão	100
	Luz	60
Quarto	Condicionador de Ar	2500
	Luz	60
Banheiro	Chuveiro	3200
	Luz	60

Figura 5.7: Lista dos cômodos, equipamentos e suas respectivas potências

Mais especificamente, o ambiente simulado trabalha com variáveis que representam os cômodos e os sensores citados anteriormente. Segue, na Figura 5.8, a lista de todas as variáveis do ambiente simulado.

Classe	Nome	Tipo	Possíveis valores
Sensor de Estado	estadoLuzCozinha	Binário	0 ou 1
	estadoGeladeiraCozinha	Binário	0 ou 1
	estadoFornoCozinha	Binário	0 ou 1
	estadoLuzSala	Binário	0 ou 1
	estadoTvSala	Binário	0 ou 1
	estadoLuzQuarto	Binário	0 ou 1
	estadoArQuarto	Binário	0 ou 1
	estadoLuzBanheiro	Binário	0 ou 1
	estadoChuveiroBanheiro	Binário	0 ou 1
Sensor de Presença	presenca	Texto	Sala, Cozinha, Quarto, Banheiro, Nenhum
Sensor de Consumo	consumo	Real	0, ..., 1, ..., 2, ..., N

Figura 5.8: Lista das variáveis do ambiente simulado

É possível perceber que, de acordo com a Figura 5.8, os equipamentos são representados pelas variáveis da classe *Sensores de Estado*. As demais variáveis representam os sensores.

Para cada variável da lista da Figura 5.8, existe um arquivo que armazena o seu valor. Computacionalmente, o ideal seria armazenar todos os valores destas variáveis em apenas um arquivo. No entanto, o *software* utilizado para a implementação deste ambiente simulado, para o desenvolvimento deste trabalho, se encontra na versão demonstrativa e, por este motivo, não possui recursos suficientes que permitem o armazenamento de valores de mais de uma variável em um mesmo arquivo.

A comunicação entre os agentes do sistema e o ambiente simulado se dá através desses arquivos. Em outras palavras, o Agente de Sensores está sempre capturando os dados do ambiente. Estes dados compreendem toda a lista da Figura 5.8. Assim, o Agente de Sensores captura

os dados dos arquivos que representam o *consumo*, a *presença* e o estado de cada dispositivo. De forma similar, o ambiente simulado também captura as decisões do sistema de controle. Ou seja, o Agente de Dispositivos, quando recebe uma requisição, desliga um ou mais equipamentos. Para tanto, este agente faz uso dos arquivos que representam as variáveis da classe *Sensor de Estado*. Desta forma, o ambiente simulado captura os dados desses arquivos e atualiza o valor de suas variáveis, para que, posteriormente, o Agente de Sensores capture as modificações no ambiente.

Capítulo 6

Conclusão

Este capítulo discute os resultados obtidos ao longo do desenvolvimento deste trabalho, bem como apresenta algumas considerações finais, levando em consideração melhorias no protótipo atual e trabalhos futuros que podem ser desenvolvidos a partir deste.

6.1 Resultados Obtidos

Como resultados obtidos, pode-se mencionar a implementação de um protótipo que contempla alguns dos agentes. Dentre estes, o agente de Perfil, o Administrador, o de Dispositivos, o de Consumo Energético e o de Sensores. O protótipo de sistema, já é capaz, por exemplo, de armazenar perfis de diferentes usuários e já têm desenvolvida toda a estrutura de comunicação entre os agentes. O Agente de Consumo Energético possui um controlador baseado em regras de comportamento e possui também um módulo com o algoritmo de RBC (com as suas quatro etapas) para auxílio na tomada de decisões. A Figura 6.1 apresenta uma tela do protótipo em execução.

O ambiente simulado, utilizado para a geração dos dados a serem processados pelo sistema, foi implementado por meio do Software Elipse Scada, citado na Seção 2.1.1. A Figura 6.2 apresenta a tela principal do ambiente em execução.

Uma vez que o foco do trabalho foi o desenvolvimento de um sistema de gerenciamento de consumo energético, foi essencial dominar os problemas e soluções que envolvem o consumo energético no País. Portanto, é importante considerar que este trabalho contou com o apoio da COPEL (Companhia Paranaense de Energia) e do projeto Casa Inteligente, coordenado pelos professores do curso de Engenharia Agrícola da Universidade Estadual do Oeste do Paraná.

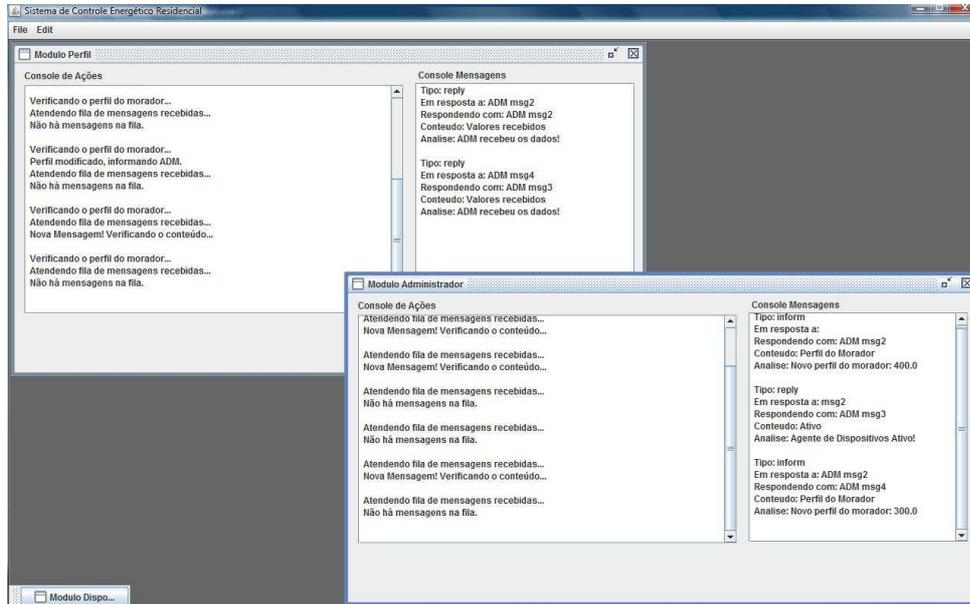


Figura 6.1: Tela do Sistema em Execução

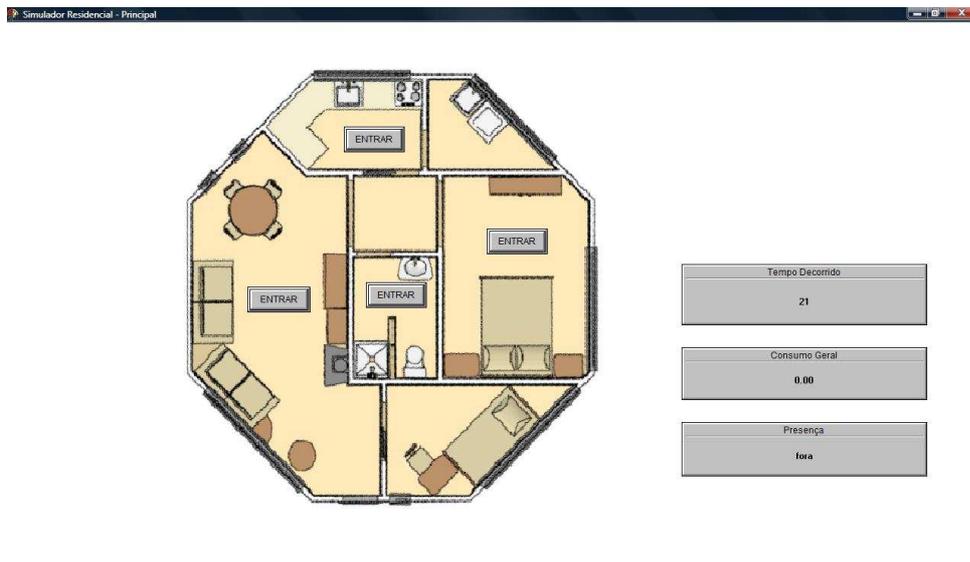


Figura 6.2: Tela do Ambiente Simulado em Execução

Os principais problemas enfrentados ao longo do desenvolvimento foram: (i) a falta de dados estatísticos disponibilizados pela COPEL. Desta forma, o mecanismo de controle não pôde ser enriquecido com muitas heurísticas; (ii) a inexistência de recursos capazes de fornecer a implantação do sistema de controle em um ambiente real, forçando a implementação de um ambiente simulado e; (iii) a limitação do software utilizado para a implementação do simulador

que, por estar em uma versão demonstrativa, não forneceu recursos para armazenar o valor de mais de uma variável em um mesmo arquivo. Desta forma, vários arquivos tiveram que ser utilizados.

6.2 Considerações Finais e Perspectivas Futuras

Com o estudo das técnicas de IA apresentadas neste trabalho, foi possível compreender melhor as características e vantagens dos Sistemas Multiagentes para o projeto e implementação do sistema de controle proposto. Embora o protótipo implementado possua apenas alguns agentes do modelo proposto inicialmente, pode-se perceber que o sistema engloba todos os objetivos previstos.

Como resultado final desta pesquisa, tem-se a definição de uma arquitetura, baseada em SMA, para o gerenciamento do ambiente residencial. O protótipo desenvolvido é capaz de gerenciar o consumo energético de um ambiente residencial. Além disso, o protótipo utiliza as preferências do morador para controlar o ambiente e economizar energia através da aplicação das técnicas de Sistemas Multiagentes e Raciocínio Baseado em Casos.

Quanto às contribuições deste trabalho e aos trabalhos futuros, além de propor a utilização de Sistemas Multiagente para o gerenciamento do consumo energético, a contribuição deste trabalho está, também, na possibilidade de organizar sistemas de automação residencial através de uma arquitetura capaz de garantir fácil modificação sem prejudicar subsistemas já existentes. Além disso, o modelo proposto facilita a agregação dos vários mecanismos de controle existentes, mesmo que, neste íterim, o foco tenha sido a aplicação algoritmo RBC para o gerenciamento do consumo energético residencial. Desta forma, trabalhos futuros podem aprofundar os estudos em Automação Residencial e agregarem mais módulos, como o de segurança e luminosidade. Além disso, mais heurísticas podem ser adicionadas ao modelo. Heurísticas para este caso específico podem ser: o horário de pico onde o consumo é mais elevado e a temperatura externa, para que seja possível identificar a necessidade de equipamentos condicionadores de ar.

Como possíveis melhorias no protótipo implementado pode-se citar: (i) a agregação das heurísticas citadas anteriormente, para o processo de tomada de decisão do Agente de Consumo Energético e o (ii) desenvolvimento de uma política de manutenção para a base de casos. Quanto

à política de manutenção da base de casos, uma estratégia válida é a utilização de um atributo que determina a frequência com que um determinado caso é utilizado. Desta maneira, os casos menos frequentemente utilizados são removidos da base de casos, reduzindo, assim, o tamanho desta base.

Referências Bibliográficas

- [1] AAMODT, A.; PLAZA, E. Case-based reasoning: Foundational issues, methodological variations, and system approaches. **AICom - Artificial Intelligence Communications**, [S.l.], v.7, n.1, 1994.
- [2] AURESIDE. **Associação Brasileira de Automação Residencial**. Disponível em <<http://www.aureside.org.br/quemsomos/default.asp?file=missao.asp&menu=quemsomos>>. Acesso em: 20 de Fevereiro.
- [3] BARRETO, J. M. **Inteligência Artificial No limiar do Século XXI Abordagem Híbrida Simbólica, Conexionista e Evolucionária**. 3. ed. Florianópolis - SC: Editora UFSC, 2001.
- [4] BIGUS, J. P.; BIGUS, J. **Constructing Intelligents Agents With Java: A Programmer's Guide to Smarter Applications**. Wiley Computer Publishing, 1998.
- [5] BITTENCOURT, G. **Inteligência Artificial: ferramentas e teorias**. 3. ed. Florianópolis - SC: Editora UFSC, 2001.
- [6] BOLZANI, C. A. M. **Desenvolvimento de um Simulador de Controle de Dispositivos Residenciais Inteligentes: uma Introdução ao Sistemas Domóticos**. São Paulo: Escola Politécnica da Universidade de São Paulo, 2004. “dissertação de mestrado em sistemas eletrônicos”.
- [7] BRENNER, W.; ZARNEKOW, R.; WITTIG, H. **Intelligent Software Agents: Foundations and Applications**. Springer-Verlag, 1998.
- [8] BUTA, P. Mining for financial knowledge with cbr. ai expert. **EUA**, [S.l.], v.9, n.2, p.34–38, Fevereiro, 1994.

- [9] DIAS, C. L. A. **Domótica: aplicabilidade às edificações residenciais**. Niterói: Universidade Federal Fluminense, 2004. “dissertação de mestrado”.
- [10] ELIPSE, M. **Elipse Software Manual - ElipseSCADA**. www.elipse.com.br.
- [11] FININ, T. et al. **DRAFT Specication of the KQML Agent-Communication Language. The DARPA Knowledge Sharing Initiative External Interfaces Working Group**. Disponível em <http://www.cs.umbc.edu/kqml/papers/kqmlspec.pdf>.
- [12] FIPA. **Foundations of Intelligent Physical Agents**. Disponível em <http://www.fipa.org/specs/fipa00037/>. Acesso em: 5 de Junho.
- [13] FIPA. **Foundations of Intelligent Physical Agents**. www.fipa.org/.
- [14] GENESERETH, M. R.; FIKES, R. E. **Knowledge Interchange Format. Version 3.0. Reference Manual**. Disponível em <http://ksl.stanford.edu/knowledge-sharing/kif/#manual>.
- [15] HERMANS, B. **Intelligent Software Agents on the Internet: an inventory of currently offered functionality in the information society and a prediction of (near-)future developments**. Tilburg, The Netherlands: Tilburg University, Julho, 1996. “tese de doutorado”.
- [16] HUHNS, M. N.; STEPHENS, L. M. **Multiagent Systems and Societies of Agents**.
- [17] JENNINGS, N. R. **Coordination Techniques for DAI.**, chapter6. John Wiley and Sons, 1996.
- [18] KOLODNER, J. L. **Case-Based Reasoning**. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.
- [19] KUNG, A.; RAITHER, B.; MCCONNELL, S. Eletronic commerce services expand home automation capabilities. **Trialog, EMMSEC'99 Conference**, [S.l.], v.1, 1999.

- [20] KUROSE, J. F.; ROSS, K. W. **Redes de Computadores e a Internet: Uma Abordagem Top-Down**, chapter3. Editora Pearson Education, São Paulo, 3. ed., 2006.
- [21] LEE, R. W. **Pesquisa Jurisprudencial Inteligente**. Florianópolis - SC: Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia de Produção PPGEP/UFSC, Maio, 1998. “tese de doutorado”.
- [22] MCCARTHY, J. **What Is Artificial Intelligence? Computer Science Department - Stanford University**. Disponível em <http://www-formal.stanford.edu/jmc/whatisai.pdf>.
- [23] MÜLLER, J. P. **Architectures and applications of intelligent agents: A survey.**, v.13, p.p. 1–24. In Knowledge Engineering Review, 1998.
- [24] MME. **Ministério de Minas e Energia, Balanço Energético 2008**. Disponível em <http://www.mme.gov.br/site/menu/select_main_menu_item.do?channelId=1432>. Acesso em: 12 de Fevereiro.
- [25] MOLL, V. Desenvolvimento de um sistema multiagente com rbc para controle automático de uma residência. Cascavel-PR: Universidade Estadual do Oeste do Paraná, 2006. “trabalho de conclusão de curso em bacharelado em informática”.
- [26] NASCIMENTO, C. L. J.; YONEYAMA, T. **Inteligência Artificial em Controle e Automação**. São Paulo: Editora Edgard Blücher, 2000.
- [27] RUSSELL, S.; NORVIG, P. **Artificial Intelligence: A Modern Approach**. 2nd edition. ed. Prentice-Hall, Englewood Cliffs, NJ, 2003.
- [28] SEARLE, J. R. **Speech Acts: An Essay in the Philosophy of Language**. Cambridge University Press, Janeiro, 1970.
- [29] SGARBI, J. A.; TONIDANDEL, F. Domótica inteligente: Automação residencial baseada em comportamento. In: WORKSHOP DE TESES E DISSERTAÇÕES EM INTELIGÊNCIA ARTIFICIAL, 2006. **Proceedings...** Ribeirão Preto - SP: [s.n.], 2006.

- [30] SILBERSCHATZ, A.; GALVIN, P. B. **Sistemas Operacionais - Conceitos e Aplicações**. São Paulo - SP: Editora Prentice Hall, 2000.
- [31] TEREZINHO, F.; CARVALHO, P. C. Ssc - sistemas de supervisão e controle. **Revista Mecatrônica Atual**, [S.l.], v.14, 2004.
- [32] URZÊDA, C. C. **Software SCADA como Plataforma para Racionalização Inteligente de Energia Elétrica em Automação Predial**. Brasília: Universidade de Brasília, 2006. “dissertação de mestrado em engenharia elétrica”.
- [33] WANGENHEIM, C. G.; WANGENHEIM, A. **Raciocínio Baseado em Casos**. 1. ed. Barueri - S: Editora Manole, 2003.
- [34] WATSON, I. **Applying Case-Based Reasoning: Techniques for Enterprise Systems**, p.p. 289. Morgan Kaufmann, San Francisco, 1997.
- [35] WOOLDRIDGE, M.; JENNINGS, N. R. Intelligent agents: Theory and practice. **The Knowledge Engineering Review**, [S.l.], v.v.10, n.n.2, 1995.