

Um Sistema de Apoio à Decisão baseado em Agentes para a Companhia Elétrica do Estado de Goiás

Ronneesley M. Teles¹, Marcos Ivamoto¹, Leonardo H. S. Mello¹,
Valdemar V. Graciano Neto¹ & Cedric Luiz de Carvalho¹

¹Instituto de Informática – Universidade Federal de Goiás (UFG)

Caixa Postal 131 – 74.001-970 – Goiânia – GO – Brasil

Fone: +55 (62) 3521 1182 (FAX)

{ronneesley | mivamoto | lhsmello | vgracianoneto | cedric.inf.ufg }@gmail.com

Abstract

Decision Support Systems (DSS) are a specific kind of Information System that has help skills in decision-making suggesting optimized solutions to the user while he is performing some task. So, DSSs must have a relevant autonomy level to be capable of deliberating about the problem is being solved and suggest a good solution to the user. Considering such need, intelligent agents seems like a suitable option to provide such capability. This paper proposes a Decision Support System based on Multi-agent Systems to support the Electric Company of the State of Goiás in its occurrences treatment job.

Keywords: Decision Support Systems, Multi-Agent, Behaviour Repository.

Sistemas de Apoio à Decisão (SAD) são um tipo específico de Sistema de Informação capaz de auxiliar no processo de tomada de decisão sugerindo ao usuário soluções otimizadas para a tarefa que está sendo desempenhada. Logo, SADs devem ter um grau de autonomia elevado, suficiente para conseguir deliberar sobre o problema a ser resolvido a fim de sugerir uma boa solução para o usuário. Considerando tal necessidade de autonomia, agentes inteligentes aparecem como uma opção adequada para prover tal capacidade ao sistema. Este artigo propõe a concepção de um Sistema de Apoio à Decisão baseado em Sistema Multi-Agente para apoiar a Companhia Elétrica do Estado de Goiás em sua tarefa de atendimento de ocorrências.

Keywords: Sistema de Apoio à Decisão, Multi-Agentes, Repositório de Comportamentos.

1. INTRODUÇÃO

A Companhia Elétrica do Estado de Goiás (CELG) é a empresa responsável pela distribuição de energia elétrica bem como pela manutenção da rede de distribuição em todo o Estado de Goiás.

A CELG é dividida em unidades gerenciais chamadas Centros de Operação e Distribuição (COD). Cada COD é responsável por uma região específica do Estado de Goiás. O COD utilizado como cenário deste trabalho é o COD correspondente à Região Metropolitana de Goiânia.

Cada COD é responsável pelo atendimento de reclamações referentes a falhas na rede elétrica na sua região de cobertura. Tais reclamações são chamadas de *ocorrências*. As ocorrências são registradas pelo serviço de atendimento ao cliente e são encaminhadas para os *despachantes* do COD.

Para tomar providências em relação às falhas ocorridas, a CELG dispõe de veículos especializados denominados *viaturas*. Um *atendimento* consiste na associação de uma viatura a uma ocorrência. As viaturas são posicionadas em locais estratégicos para otimizar o atendimento das ocorrências. Os despachantes são responsáveis por alocar as viaturas para atender as ocorrências registradas. Este posicionamento é realizado a cada início de turno das viaturas. O dia é dividido em três turnos de oito horas. Esta distribuição é feita manualmente e com base no conhecimento e experiência dos despachantes.

Para alocar viaturas às ocorrências, os despachantes

devem atentar-se a várias restrições que a CELG deve respeitar: o tempo que o cliente está sem fornecimento, a urgência da ocorrência, a distância da ocorrência às viaturas, o valor homem-hora (HH) dos técnicos que atendem às ocorrências, horas excedentes caso o turno da viatura tenha se encerrado e quilometragem excedente em relação ao limite estipulado por veículo.

A quantidade de despachantes é bastante reduzida em relação à quantidade de ocorrências registradas por dia. Para tomar decisões, a única ferramenta que os despachantes têm como auxílio é um sistema computacional que exhibe a região de atendimento, as ocorrências, a prioridade de cada uma, a localização das viaturas e a localização de cada ocorrência. Não existe uma ferramenta que auxilie na seleção da melhor viatura para atendimento de uma ocorrência, garantindo que o consumidor seja atendido no menor tempo possível e com o menor custo para a empresa.

O cenário descrito exhibe vários inconvenientes:

1. o conhecimento envolvido na tomada de decisão durante um atendimento não é documentado. Assim, apenas os despachantes humanos detêm tal conhecimento em suas mentes;
2. não há ferramentas que possibilitem uma análise histórica e/ou estatística das decisões tomadas e das ocorrências registradas. Sem estes dados, não se obtém uma otimização nas decisões;
3. as decisões são tomadas com base em conhecimento implícito, não verificável ou validável;
4. não se tem garantias da otimalidade das soluções concebidas e;
5. erros dispendiosos podem se repetir devido a não ser possível observar falhas de experiências anteriores [6].

Os inconvenientes apresentados revelam que a CELG possui um processo de tomada de decisão ineficiente e não auditável. Além disso, o conhecimento e a memória organizacional não são documentados ou armazenados. Como o mecanismo de tomada de decisão não é um processo bem definido, o capital intelectual não é preservado.

Este artigo propõe a concepção de um Sistema de Apoio à Decisão Inteligente baseado em Sistema Multi-Agentes. A intenção é otimizar a solução destes problemas e auxiliar os despachantes na tomada de decisão referente à distribuição das viaturas e a alocação das viaturas para atender às ocorrências de acordo com as restrições estipuladas.

As vantagens inicialmente observáveis são inerentes à utilização da técnica de Sistemas Multi-Agentes. Dentre as vantagens, pode-se citar processamento distribuído

de ocorrências, automatização, inteligência, troca de mensagens entre despachantes e viaturas, sugestão de soluções com base em conhecimento armazenado e não apenas com base na experiência dos despachantes.

Espera-se, como consequência, um aumento na velocidade de atendimento das ocorrências, minimização do tempo que o usuário fica sem fornecimento, redução dos custos com desperdício de recursos humanos e financeiros, e a preservação do capital intelectual e da memória organizacional da CELG, minimizando os problemas apresentados.

O restante deste trabalho está estruturado da seguinte forma: a Seção 2 trata da motivação de utilizar agentes para implementação deste trabalho; a Seção 3 apresenta o modelo proposto, discute a implementação e os conceitos utilizados; a Seção 4 discute os trabalhos correlatos; a Seção 5 traz as conclusões, trabalhos futuros e discussões finais sobre o tema e, por fim, são apresentadas as referências utilizadas neste trabalho.

2. MOTIVAÇÃO PARA O USO DE AGENTES

Para auxiliar a CELG, foi concebido um Sistema de Apoio à Decisão (SAD) baseado em Sistemas Multi-Agentes (SMA).

Várias iniciativas de utilizar agentes no desenvolvimento de SADs tem sido citadas na literatura. Harbouche e Djoudi [8] citam que o uso de agentes é adequado quando estes são uma metáfora natural dos atos humanos e que agentes possuem uma alta capacidade de representação de comportamento. Estas propriedades endossam a utilização de agentes neste trabalho, no qual pretende-se simular os comportamentos de operadores humanos responsáveis pela rede elétrica e dos despachantes.

Eom e Kim [7] reafirmam a adequação desta abordagem quando mencionam que o uso de agentes inteligentes está aumentando no desenvolvimento de SADs devido a um propósito especial dos agentes: reduzir o trabalho e a sobrecarga de informações sobre o usuário.

Bellifemine et al. [2] sugerem que a maneira mais fácil de obter comportamento coerente e resolução de conflitos é prover um grupo com um agente que tem uma perspectiva mais ampla do sistema, explorando uma estrutura hierárquica e/ou organizacional. Este agente controlador seria responsável por captar informação dos agentes em um grupo, criar planos e atribuir tarefas para agentes individuais para reforçar a coerência global.

Este artigo propõe uma adaptação das ideias sugeridas por Bellifemine et al [2]. Tal adaptação inclui a utilização de um sistema multi-agentes heterogêneo no qual os agentes possuem um caráter cooperativo, composto de três tipos de agentes: viatura, despachante e comunicador.

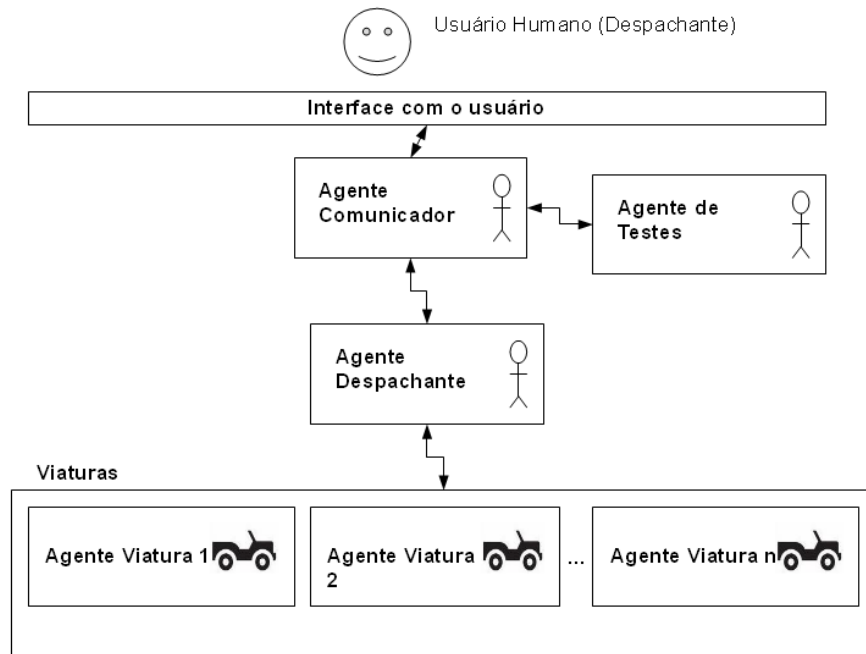


Figura 1. Arquitetura do Sistema

Para isso, a região metropolitana de Goiânia é dividida em regiões gerenciadas por um agente despachante que representa o conhecimento dos despachantes humanos.

As viaturas serão representadas por agentes reativos. A sugestão das viaturas mais adequadas para cada ocorrência acontecerá como resultado da troca de mensagens entre o agente do tipo despachante e agentes do tipo viatura sob sua responsabilidade.

3. MODELO PROPOSTO

Um protótipo foi desenvolvido utilizando o framework JADE, um arcabouço totalmente implementado na linguagem Java que simplifica a implementação de sistemas multi-agentes [9]. As seções seguintes descrevem o modelo implementado neste protótipo.

3.1. DESCRIÇÃO

A Figura 1 apresenta a arquitetura do modelo. Em concordância com vários trabalhos presentes na literatura ([1], [4], [8] e [14]), a arquitetura apresenta um agente especializado em receber dados do usuário e comunicar aos agentes do SMA as requisições realizadas pelo usuário humano. Tal agente é chamado Agente Comunicador. Este agente é responsável por efetuar a comunicação entre o usuário e o restante do sistema, em especial, os outros agentes.

Toda requisição feita pelo usuário é encaminhada pelo

agente comunicador para o agente despachante.

O modelo tenta estabelecer um processo para dar suporte ao trabalho desenvolvido na CELG, composto de duas fases principais a saber: distribuição de viaturas em cada início de turno e alocação das viaturas às ocorrências (despacho) tomando as restrições mencionadas como parâmetro para escolha e/ou sugestão.

Para resolver o problema da alocação de viaturas a cada início de turno, foi criada uma heurística chamada **Heurística da Diagonal**. Esta heurística processa os dados históricos de ocorrências na região metropolitana de Goiânia para distribuir as viaturas de modo igualitário nas várias regiões da cidade.

Primeiramente, verifica-se quantas ocorrências foram registradas em toda a região metropolitana durante um certo período na base histórica (por exemplo, um mês). Divide-se então a quantidade total de ocorrências pela quantidade total de viaturas obtendo-se um valor que representa a *média de ocorrências atendidas por viatura (moav)* no período considerado. Esta média é então utilizada como parâmetro na Heurística da Diagonal. A equação 1 ilustra um equivalente matemático a esta descrição, onde $qtdeOcorrencias$ representa a quantidade de ocorrências registradas no período considerado e $qtdeViaturas$ representa a quantidade de viaturas disponíveis para realizar o atendimento.

$$moav = \frac{qtdeOcorrencias}{qtdeViaturas} \quad (1)$$

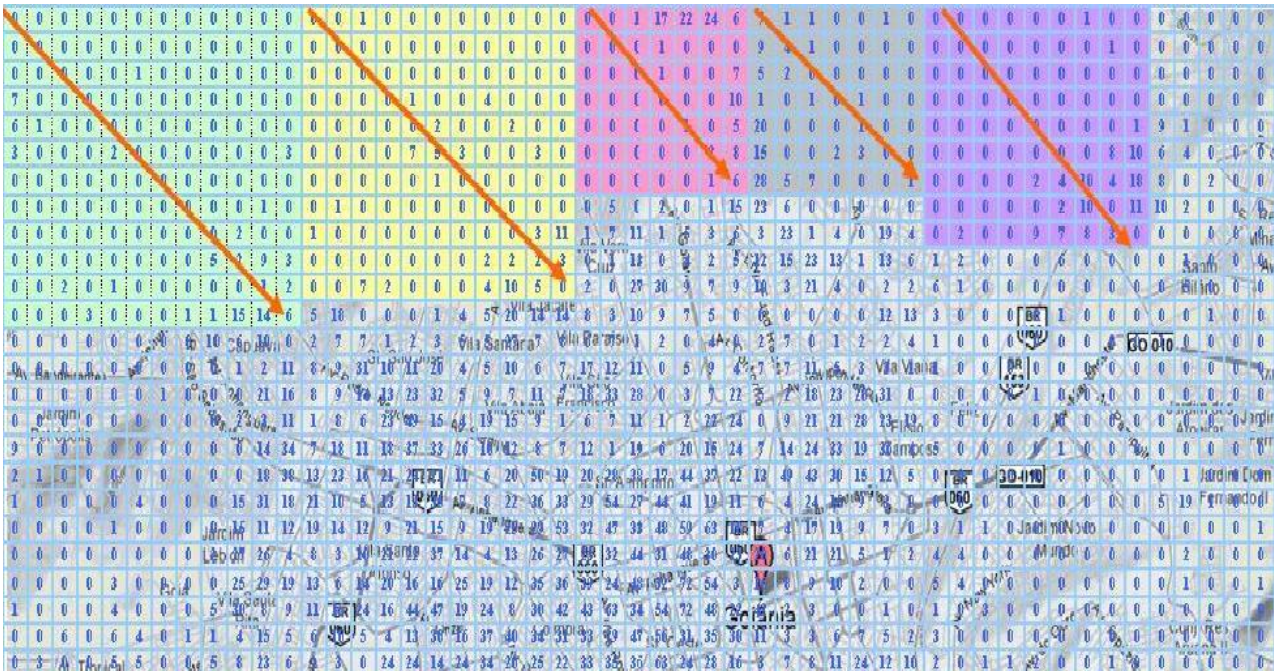


Figura 2. Início da Execução da Heurística da Diagonal

Em seguida, de posse do valor do *moav*, é possível criar sub-regiões gerenciáveis. Sub-regiões são regiões geográficas obtidas a partir da execução da heurística. Estas sub-regiões serão formadas por uma ou mais micro-regiões de 1 km².

Em cada início de turno, executa-se a Heurística da Diagonal. Se uma micro-região possui uma quantidade de ocorrências igual ou superior ao *moav* do período histórico para o qual este valor foi calculado, é associada uma viatura àquela micro-região que passa a ser considerada uma sub-região. Mas, se ao contrário, a micro-região de 1 km² não atingir o valor do *moav* calculado, então a Heurística da Diagonal expande a área considerada no sentido da diagonal principal (do canto superior esquerdo para o canto inferior direito) da micro-região sendo avaliada. A área candidata a sub-região agora possui 4 km². Se esta área possuir uma quantidade de ocorrências no período considerado igual ou superior ao *moav*, então marca-se esta área como sendo uma sub-região e ela recebe uma viatura. Caso ainda não atinja o valor *moav*, a heurística continua executando iterativamente, expandindo a área, até que finalmente o acumulado de ocorrências no período registrado iguale-se ou supere o *moav* calculado e aquela seja considerada uma sub-região.

Depois que uma sub-região é definida, a heurística continua sua execução até que toda a área de Goiânia esteja dividida nestas sub-regiões. As viaturas sobressalentes são divididas proporcionalmente em relação à quantidade de ocorrências que ultrapassaram o valor de

moav em cada sub-região.

A Figura 2 apresenta alguns blocos obtidos a partir da execução da Heurística da Diagonal. O primeiro bloco possui 144 km². É possível notar nesta figura que as micro-regiões de 1 km² possuem associadas a si um número. Este número corresponde à quantidade de ocorrências registradas na micro-região no último período considerado. A Figura 2 ilustra apenas uma porção da região metropolitana.

A Figura 3 ilustra a região metropolitana como uma matriz dividida em várias sub-regiões. Esta figura corresponde ao que seria a região metropolitana dividida em sub-regiões após a execução da Heurística da Diagonal. Para isso, a região metropolitana é considerada um retângulo de 168 km por 137 km.

A Heurística da Diagonal possui algumas peculiaridades. Um exemplo é o problema que foi chamado de **Problema da Fronteira**. Esta situação acontece quando, percorrendo uma linha e produzindo as sub-regiões, chega-se a uma situação na qual não é possível avançar no sentido da diagonal e a quantidade de ocorrências registradas é inferior ao *moav*. Nestes casos, convencionou-se aglutinar esta área sobressalente à última sub-região definida. O fato de a área desta última sub-região definida aumentar não é um problema porque as viaturas sobressalentes são distribuídas sempre proporcionalmente à quantidade de ocorrências que excedem o *moav* naquela área.

Depois que a Heurística da Diagonal é executada, a

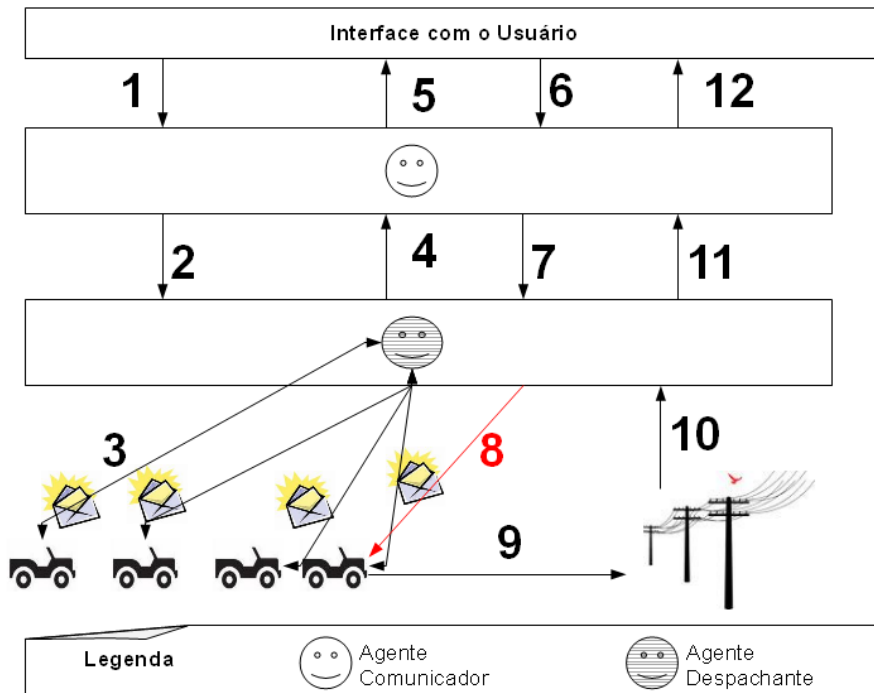


Figura 4. Processo Completo de Sugestão de Viaturas

uma das grandes vantagens em se utilizar SMA: a possibilidade de abstrair melhor a realidade. Afinal, SMA provê a capacidade de representação de várias viaturas trabalhando de forma paralela nas diversas sub-regiões da cidade, assim como acontece no cenário real.

A troca de mensagens entre agentes viatura e despachante dispara comportamentos como cálculo de distância nos agentes viatura e comportamento de despacho nos agentes despachantes. A seção seguinte descreve a forma como os comportamentos são armazenados dentro da aplicação desenvolvida.

3.2. REPOSITÓRIO DE COMPORTAMENTOS

Em SMAs convencionais desenvolvidos com JADE [9], os comportamentos são implementados como parte do código da aplicação. Os comportamentos são adicionados aos agentes através do método *addBehaviour()*. A Figura 5, retirada de Bellifemine et al. [2], mostra um exemplo de adição de um comportamento cíclico a um agente. Nesta figura, é possível observar como o código do comportamento fica acoplado ao código do agente. Com isso, algumas premissas clássicas da Engenharia de Software acabam por não ser atendidas, tais como alta coesão, baixo acoplamento e a consequência de ambas: boa manutenibilidade.

Boff e Oliveira [3] propõem uma solução que auxilia na portabilidade e manutenibilidade de Sistemas de Informação (SI). Nesta abordagem, eles isolam as Regras de

```
import jade.core.Agent;
import jade.core.behaviours.CyclicBehaviour;
import jade.lang.acl.ACLMessage;

public class HelloWorldAgent extends Agent {

    public void setup() {

        System.out.println("Hello. My name is "+getLocalName());

        addBehaviour(new CyclicBehaviour() {
            public void action() {
                ACLMessage msgRx = receive();
                if (msgRx != null) {
                    System.out.println(msgRx);
                    ACLMessage msgTx = msgRx.createReply();
                    msgTx.setContent("Hello!");
                    send(msgTx);
                } else {
                    block();
                }
            }
        });
    }
}
```

Figura 5. Exemplo de adição de comportamento retirado de Bellifemine et al. [2]

Negócio (RN) do SI em um repositório de modo que, se for necessária alguma atividade de manutenção, inserção ou remoção de regras no modelo de domínio especificado, basta realizar estas alterações no repositório e tais alterações serão propagadas por todo o sistema.

Criou-se neste trabalho uma variação do conceito apresentado por Boff e Oliveira [3]: o conceito de

Repositório de Comportamentos. O princípio de funcionamento é semelhante ao apresentado no trabalho de Boff e Oliveira. Os comportamentos dos agentes podem ser vistos como elementos do SMA que podem ser isolados em um repositório. Quando o sistema vai ser carregado, os comportamentos especificados no repositório são então adicionados aos agentes antes que entrem em execução. Assim, os comportamentos não são especificados juntamente aos agentes, mas sim no repositório. Além dos comportamentos, são registradas tuplas que relacionam os agentes aos quais cada comportamento deve ser adicionado.

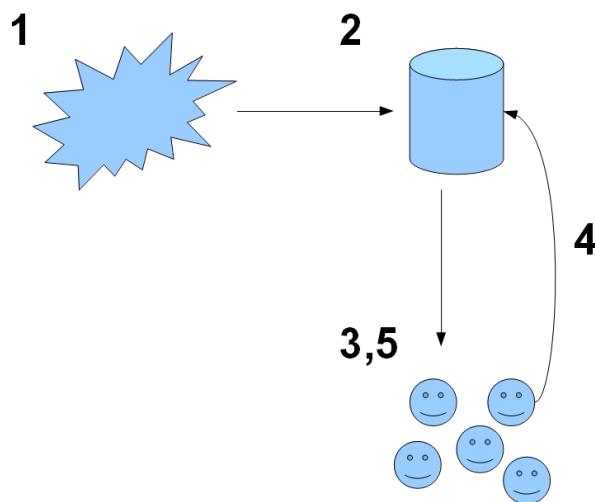


Figura 6. Ciclo de Manutenção de Comportamentos

Caso seja desejado modificar um comportamento, basta selecionar o comportamento do Repositório, efetuar a alteração, e requisitar a replicação no sistema.

A Figura 6 ilustra o ciclo de vida dos comportamentos no Repositório de Comportamentos:

1. A aplicação é inicializada;
2. os comportamentos são lidos do repositório, instanciados e adicionados aos agentes. Este passo é executado observando quais agentes estão associados a quais comportamentos dentro do repositório;
3. os agentes são criados efetivamente;
4. em virtude de uma requisição de mudança, são efetuadas mudanças em um ou mais comportamentos no repositório;
5. os comportamentos são propagados nos agentes, facilitando assim a manutenção e teste dos mesmos.

Tal conceito é muito importante pois traz para este trabalho as mesmas vantagens apresentadas no trabalho

de Boff e Oliveira [3]: portabilidade (não somente relacionada a plataformas distintas, mas até mesmo a domínios distintos, podendo aplicar tal conceito a qualquer SI que possa se utilizar de sistemas multi-agentes) e manutenibilidade (agilidade e corretude na modificação de comportamentos do SMA como um todo).

Este conceito aparece como uma boa prática de programação orientada a agentes. Criar classes que representam os comportamentos dos agentes e isolá-los em um repositório permite um melhor reaproveitamento de tais códigos, reduz o acoplamento, facilita a manutenção e melhora a legibilidade do código.

4. TRABALHOS RELACIONADOS

Foram encontrados alguns trabalhos que se assemelham ao trabalho desenvolvido em relação à tecnologia utilizada para implementar a inteligência do SAD.

Silva e Ito [5] apresentam um SAD baseado em agentes utilizado para auxiliar no tratamento de pacientes crônicos. Eles usam JADE para desenvolver tal sistema e criam agentes utilizando a arquitetura BDI. No que tange à tecnologia utilizada para implementação da inteligência do SAD, Silva e Ito [5] desenvolvem um trabalho estritamente correlato. A diferença é que a quantidade de agentes utilizada é menor e os agentes utilizados são do tipo BDI, possuindo capacidade cognitiva. Isso minimiza a necessidade de uma grande quantidade de agentes atuando no sistema.

Cizbula et al. [4] apresentam uma arquitetura genérica para um SAD Multi-agente que dá suporte à tomada de decisão baseado em técnicas de aprendizado supervisionado. A principal vantagem da arquitetura proposta é que ela pode ser facilmente usada em diferentes domínios nos quais Sistemas de Apoio à Decisão são necessários: saúde, psicologia, economia, seguros, etc. Para validar a proposta, Cizbula et al. [4] implementam dois estudos de caso para a arquitetura genérica aplicada a um Sistema de Apoio à Decisão Médica, que apóia o diagnóstico de doenças.

O trabalho de Cizbula et al. [4] possui semelhanças com o que está sendo aqui descrito com relação a algumas características do domínio de aplicação (essencialmente distribuído e paralelo). Além disso, este trabalho contribui no sentido de generalizar a arquitetura que eles descreveram para aplicá-la a vários domínios. O projeto aqui descrito também tem por intuito generalizar a aplicabilidade da solução concebida.

Yu et al. [14] apresentam um Sistema de Apoio à Decisão para Desenvolvimento de Produto baseado em Multi-Agente. Yu et al. [14] dizem que o desenvolvimento de um novo produto é uma tarefa muito complexa e que envolve muitos riscos, pois a falha na populariza-

ção de novos produtos ocorre em uma taxa elevada (66% a 90% dos produtos lançados não se popularizam).

Para capturar informações relevantes a respeito deste domínio, Yu et al. [14] criam agentes. Cada agente é responsável por receber informações de uma fonte diferente (usuário, mercado, departamentos), analisá-las, deliberar com outros agentes e, em seguida, responder ao usuário sobre a viabilidade de lançamento do produto. Segundo Yu et al. [14], a diversidade de fontes e de sub-problemas a serem resolvidos destacam o caráter distribuído deste problema.

Para ajudar na solução deste problema, Yu et al. [14] oferecem um SAD para clarear as vantagens e desvantagens de cada projeto, entender o impacto de várias incertezas sobre o projeto para reduzir riscos ou evitá-los e reduzir problemas advindos de um lançamento de produto mal-sucedido. Os agentes de Yu et al. [14] são do tipo cognitivo, enquanto que os agentes deste trabalho são reativos. Trata-se de um trabalho que também utiliza agentes para prover a inteligência do SAD.

Em relação à utilização de agentes para auxiliar empresas de energia elétrica, Nagata e Sasaki [10] apresentam um trabalho no qual agentes são utilizados para auxiliar no processo de encontrar o ponto de falha na rede elétrica de modo automático, sem a necessidade do contato do cliente notificando sobre a falha na rede. Tolbert et al. [13] mostra um cenário no qual agentes auxiliam no gerenciamento, otimização do consumo e geração de energia. Rehtanz [12] menciona a utilização de agentes na operação e controle de Sistemas de Energia Elétrica, auxiliando na gerência da rede, pontos de falha e fluxo de energia. Phadke e Thorp [11] sugerem a utilização de agentes para auxiliar no processo de retransmissão de energia.

Não foi encontrado nenhum outro trabalho na literatura que utilizasse agentes para resolver problemas de distribuição de veículos para manutenção de redes de energia elétrica, o que torna este trabalho inovador em sua abordagem de manutenção de redes elétricas.

5. CONCLUSÕES E TRABALHOS FUTUROS

Concluiu-se que a utilização de Sistemas Multi-Agentes para a implementação de Sistemas de Apoio à Decisão para resolver problemas como o descrito pode ser uma alternativa a se considerar. Sistemas de Apoio à Decisão Inteligentes parecem ser uma alternativa viável para auxiliar os despachantes a cumprir seu trabalho sem a necessidade de efetuar cálculos e análises que demandavam tempo, esforço e eram ainda sujeitos a erros.

Além disso, tal tipo de sistema protege o capital intelectual da corporação e a memória organizacional, pois

as decisões não são mais tomadas com base exclusivamente na experiência dos despachantes humanos, mas sim como uma cooperação entre a sugestão oferecida pelo sistema - regido por um processo definido - e a decisão do despachante humano.

Vislumbra-se ainda melhorias para o SAD em questão como:

1. a possibilidade de transformar o Repositório de Comportamentos em uma ferramenta de testes de comportamentos, capaz de emular comportamentos e sugerir para o mantenedor do sistema o melhor comportamento para um cenário específico,
2. tornar a definição do comportamento independente de plataforma utilizando alguma linguagem de especificação de comportamentos,
3. aumentar a inteligência dos agentes viatura, convertendo-os em agentes cognitivos, capazes de deliberar por si próprios e negociar com os agentes de sua sub-região qual a melhor opção para atender às ocorrências que chegam,
4. aumentar a quantidade de agentes do tipo despachante de modo a dar ao modelo um caráter mais distribuído, criando um agente despachante para cada sub-região, minimizando a sobrecarga decorrente da alta quantidade de mensagens trocadas entre os agentes.

O conceito de Repositório de Comportamentos mostrou-se muito interessante do ponto de vista de manutenibilidade de SMA. Uma melhoria que pode ser implementada é estender este conceito para suportar comportamentos definidos utilizando outras técnicas de Inteligência Artificial para prover inteligência aos agentes, tais como Redes Neurais, Árvore de Decisão, Algoritmos Genéticos, Lógica Fuzzy ou Redes Bayesianas.

A arquitetura apresentada pode ser utilizada para auxiliar os despachantes de qualquer companhia elétrica na seleção da viatura mais adequada para atender a uma ocorrência, com o menor tempo de espera por parte do consumidor e menor custo associado a recursos humanos e financeiros. Mas, a solução não é restrita ao domínios de companhias de eletricidade. A abordagem seria útil a qualquer tipo de cenário no qual deseja-se gerenciar uma frota de veículos de modo a posicioná-los em regiões estratégicas em relação a um acumulado de dados históricos e de modo a roteá-los para atender a solicitações em posições geográficas especificadas.

O conceito de Repositório de Comportamentos pode também ser estendido a qualquer aplicação desenvolvida sobre SMA. Qualquer sistema desenvolvido sobre SMA pode ter os comportamentos armazenados num

repositório para facilitar a manutenção e a propagação do comportamento por todos os agentes. Além disso, comportamentos definidos para um tipo de agente específico podem ser reutilizados em outros agentes em aplicações distintas. Exemplo: o comportamento de **cálculo de distância** pode ser utilizado em qualquer contexto no qual tem-se agentes que precisam saber suas distâncias até um determinado ponto. Aplicações para processamento geográfico, otimização de rotas, etc são alguns exemplos citáveis de aplicações que podem ter agentes que compartilham comportamentos.

Apesar de ter sido implementado um protótipo, não foram feitos testes apurados devido à falta de dados disponíveis e acesso aos dados da empresa até a data de escrita deste artigo.

Referências

- [1] Andrea Addis, Giuliano Armano, and Eloisa Vargiu. From a generic multi-agent architecture to multi-agent information retrieval systems. *Sixth International Workshop From Agent Theory to Agent Implementation*, 2008.
- [2] Fabio Luigi Bellifemine, Giovanni Caire, and Dominic Greenwood. *Developing multi-agent systems with JADE*. Wiley, 1th edition, 2007.
- [3] Glauber Boff and Juliano Lopes de Oliveira. Modelagem, implementação e manutenção de regras de negócio em sistemas de informação. *Workshop de Manutenção de Software Moderno - Simpósio Brasileiro de Qualidade de Software*, 2009.
- [4] Gabriela Czibula, Adriana Mihaela Guran, and Grigoreta Sofia Cojocarand Istvan Gergely Czibula. Multiagent decision support systems based on supervised learning. *2008 IEEE International Conference on Automation, Quality and Testing, Robotics - Volume 03*, pages 353–358, 2008.
- [5] Leandro Ramos da Silva and Márcia Ito. Proposta de um sistema multiagentes para acompanhamento de pacientes crônicos. *Anais do XI Congresso Brasileiro de Informática na Saúde*, 2008.
- [6] José Braga de Vasconcelos, Álvaro Rocha, and Chris Kimble. Sistemas de informação de memória organizacional: Uma abordagem ontológica para a definição de competências de grupo. *Actas da 4ª Conferência da Associação Portuguesa de Sistemas de Informação*, 2003.
- [7] Sean Eom and Eyoung Kim. A survey of decision support system applications (1995–2001). *Journal of the Operational Research Society*, 2005.
- [8] Khadidja Harbouche and Mahieddine Djoudi. Agent-based design for e-learning environment. *Journal of Computer Science*, pages 383–389, 2007.
- [9] Telecom Italia. Jade - java agent development framework. Disponível em: <http://jade.tilab.com/>. Acesso em: 24 jan. 2009, 2009.
- [10] T. Nagata and H. Sasaki. A multi-agent approach to power system restoration. *IEEE Transactions on Power Systems*, pages 457–462, 2002.
- [11] Arun Phadke and James Thorp. *Computer Relaying for Power Systems*. Wiley, 2009.
- [12] Christian Rehtanz. *Autonomous Systems and Intelligent Agents in Power System Control and Operation*. Springer-Verlag, 2003.
- [13] Leon M. Tolbert, Hairong Qi, and Fang Z. Peng. Scalable multi-agent system for real-time electric power management. *IEEE Power Engineering Society Summer Meeting*, pages 1676–1679, 2001.
- [14] Tianbiao Yu, Jing Zhou, Feng Xu, Yadong Gong, and Wanshan Wang. Decision support system of product development based on multi-agent. *2009 International Conference on Information Technology and Computer Science*, 2009.