

Método Baseado em Detecção de Mudanças para Determinar Preço de Oferta de Pedidos de Clientes no Ambiente TAC-SCM

Fernando Roberto Pereira, Roni Fabio Banaszewski, Jean Marcelo Simão, Cesar Augusto Tacla

Curso de Pós-Graduação em Engenharia Elétrica e Informática Industrial (CPGEI)
Universidade Tecnológica Federal do Paraná (UTFPR)
Av. Sete de Setembro, 3165
Fone: +55 (41) 33104702
CEP 80230-901 - Curitiba - PR - BRAZIL
{nandoroberto, banaszewski, jeansimao, tacla}@cpgei.ct.utfpr.edu.br

Abstract

This paper presents an adaptation of an approach based on Drift Detection and instance based machine-learning. This adapted approach is applied to agents in stochastic simulation scenarios of the TAC-SCM (Trading Agent Competition - Supply Chain Management) environment in order to determine selling prices of goods. In short, this approach allows detecting frequent market changes and determining a competitive price based on feedback obtained from negotiations with clients. The proposed approach is evaluated by practical experiments. It is also concomitantly presented an approach to minimize the expensive computable cost caused by the instance based machine learning technique used in the experiments.

Resumo

Este artigo apresenta adaptações de uma abordagem fundamentada em técnicas de detecção de variações chamada Drift Detection e de aprendizagem de máquina baseada em instâncias. Esta adaptação é aplicada para realizar venda de produtos em cenários estocásticos providos pelo ambiente de simulação TAC-SCM. Em suma, a abordagem adaptada permite detectar mudanças frequentes de mercado e determinar um preço competitivo em função do feedback obtido nas negociações com os clientes.

Neste âmbito, buscou-se demonstrar por meio de experimentos práticos o comportamento da abordagem proposta. Ademais, é apresentada conjuntamente uma solução para minimizar o custo computacional dispendioso acarretado pela técnica de aprendizagem de máquina baseada em instâncias.

Keywords: Aprendizagem Baseada em Instâncias; *Dynamic Weighted Majority*; Previsão de Preços.

1. INTRODUÇÃO

A Gestão de Cadeias de Suprimentos ou *Supply Chain Management* (SCM) consiste no planejamento e coordenação das atividades de uma organização desde a obtenção de matéria prima até a entrega do produto final. De acordo com [14], o SCM corresponde a um conjunto de abordagens usadas para integrar eficientemente fornecedores, fábricas, depósitos e armazéns, sendo que as mercadorias devem ser distribuídas na quantidade certa, no local certo e na hora certa, minimizando os custos globais do sistema e satisfazendo os requisitos impostos.

A gestão eficiente das cadeias de suprimentos é vital para que organizações se sobressaiam no mercado cada vez mais competitivo [4]. Devido a esta importância, muitas soluções inovadoras são propostas para melhorar o processo de SCM, inclusive adotando métodos oriundos da Inteligência Artificial [15].

Para que estes métodos possam ser estudados sem provocar riscos aos processos das organizações, se fazem necessárias ferramentas que simulem as principais características de um mercado competitivo. Com o objetivo de incentivar estes estudos, as entidades e-Supply Chain Management Lab da Carnegie Mellon University e Swedish Institute of Computer Science (SICS) desenvolveram um ambiente de simulação de cadeia de suprimentos no segmento de computadores [2]. Este ambiente é denominado *Trading Agent Competition - Supply Chain Management (TAC-SCM)*.

O ambiente TAC-SCM permite a competição entre agentes que representam cada qual uma organização. Cada agente deve realizar três atividades principais na interação com o mercado simulado: *i)* a compra de componentes (i.e. placa-mãe, processador, disco rígido e memória) para a produção de computadores; *ii)* a produção de computadores e, *iii)* a venda dos computadores produzidos com o intuito de aumentar o seu lucro. Estas atividades estão vinculadas a certas peculiaridades derivadas do mundo real (e.g. juros bancários, taxas de estocagem de produtos, penalidades por atraso na entrega de produtos), que devem ser contornadas pelos agentes a fim de que se tornem competitivos no mercado.

Dentre as atividades realizadas pelos agentes, o presente artigo aborda com maior ênfase a atividade de venda de computadores. Mesmo que haja influência entre as execuções das três atividades, a atividade de venda é a que mais contribui para aumentar a lucratividade dos agentes na competição, uma vez que busca estabelecer preços de venda competitivos em relação aos preços dos outros agentes do mercado.

No ambiente TAC-SCM, a atividade de venda é realizada em um cenário estocástico. As negociações entre os agentes e os clientes (i.e. consumidores de computadores) ocorrem por meio do protocolo de leilão fechado e de primeiro preço [13].

Neste tipo particular de negociação, os agentes participantes disputam por meio de lances únicos os pedidos de computadores provenientes dos clientes com a condição de não conhecer os lances de seus adversários. O agente que oferecer o lance com o menor preço (menor preço de venda) para um pedido é considerado o vencedor. Conseqüentemente, o agente vencedor deverá vender a quantidade de produtos definido no pedido pelo preço acordado na negociação.

Para determinar o preço de venda dos computadores, é necessário levar em consideração vários fatores, como: a) capacidade de produção

(montagem de computadores); b) condições de mercado; d) disponibilidade de matéria prima (i.e. componentes para montagem dos computadores). Dentre estes fatores existem dois cruciais, condições de mercado e acessibilidade a matéria prima.

As condições de mercado têm relação direta aos preços dos concorrentes, logo apresenta indeterminismo e, por isso, não podem ser fixos. O fator disponibilidade de matéria prima está vinculada às condições de fabricação de cada fornecedor, as quais são determinadas de forma estocástica pelo ambiente TAC-SCM. Ainda, como exemplo de características que são consideradas pelo ambiente TAC-SCM para gerir os fornecedores pode ser citadas: a) capacidade inicial da produção; b) capacidade normal de produção; c) taxa de reputação, dentre outras.

Em situações onde o ambiente é estocástico, os conceitos (no caso deste trabalho, o conceito é *preço de venda competitivo*) mudam continuamente, sendo que tais mudanças podem ocorrer de forma abrupta ou ocorrer gradualmente. Neste contexto, as detecções de variações em aprendizagem dinâmica pertencem a uma classe de problema conhecida como *drift detection* [7], que é contemplada neste artigo.

Neste artigo, apresenta-se a adaptação de uma abordagem proposta em [6] que faz uso de uma técnica de detecção de variações, mas especificamente um algoritmo representante desta classe denominado *Dynamic Weighted Majority (DWM)* em conjunto com uma técnica de aprendizagem de máquina baseada em instâncias conhecida como IB3.

Particularmente, a aprendizagem baseada em instâncias requer um conjunto de características que representam as instâncias de treinamento, as quais devem pertencer a uma determinada classe. Um exemplo de classe é a ‘compra de produtos’.

Em [6], os autores aplicaram o DWM para realizar justamente as atividades de compra de produtos, onde classificam se uma compra é interessante ou não em um cenário com características similares ao apresentado no TAC-SCM. Uma particularidade importante que é encontrada em [6] é que todos os atributos para compor as instâncias de treinamento são conhecidos.

Neste artigo, apresenta-se a adaptação da abordagem proposta em [6] e adequação da mesma por meio de estratégias baseadas em buscas para determinar preços de venda de computadores. Esta adequação é necessária em virtude da falta de um atributo para compor as instâncias, atributo este sendo o preço de venda.

Esta adequação inova e traz como benefício para abordagem proposta em [6] a utilização da mesma em situações onde a priori não se tem todos os atributos para formar a instância de classificação.

Dentre as contribuições já mencionadas, é importante salientar sobre a contribuição na área de negociação, mais especificamente políticas de negociações entre agentes. Onde o método proposto conjuntamente com o algoritmo DWM, buscam fazer descobertas de forma automática e dinâmica das políticas de oferta dos agentes oponentes.

Em suma, esta abordagem habilita o agente a detectar as mudanças frequentes de mercado (i.e. mudança da demanda dos clientes por tipo de computador) e determinar um preço competitivo em função dos lances aceitos/rejeitados obtidos nas negociações com os clientes. Assim, são apresentadas e comparadas por meio de simulações no TAC-SCM duas estratégias de busca, seqüencial e binária, para determinar valores para o atributo preço de venda.

2. FUNDAMENTAÇÃO TEÓRICA

Esta seção possui a seguinte estrutura: a subseção 2.1 aborda a competição *Trading Agent Competition* e o respectivo simulador TAC-SCM, e a subseção 2.2 descreve os conceitos de Inteligência Artificial abordados neste artigo.

2.1. TRADING AGENT COMPETITION

O *Trading Agent Competition* (TAC) é um fórum internacional destinado a incentivar pesquisas em problemas de negociação entre sistemas de agentes, principalmente relacionados a SCM. Este fórum é realizado anualmente e aborda diferentes cenários de cadeias de suprimento. Um deles se refere ao segmento de fabricação e vendas de computadores. Este cenário é simulado em um ambiente denominado TAC-SCM [2].

O ambiente TAC-SCM executa em um servidor e oferece suporte para conexão simultânea de até 6 agentes competidores. Estes devem tomar decisões de forma autônoma, ou seja, sem intervenção humana. Basicamente, uma simulação é iniciada em um instante pré-agendado e é finalizada após o decorrer de 220 dias virtuais, sendo que cada dia virtual tem duração de 15 segundos.

Em termos funcionais, o ambiente TAC-SCM permite a simulação das ações de consumidores de computadores e fornecedores de componentes para a fabricação dos computadores, além de simular serviço bancário, e de produção e armazenagem. Basicamente,

a função dos consumidores é adquirir computadores de algum dos 6 agentes conectados no simulador e a função dos fornecedores é fornecer componentes para estes agentes produzirem os computadores.

Por sua vez, cada agente desempenha funções para gerenciar a cadeia de suprimentos pela qual é responsável. Cada agente realiza atividades de vendas de computadores, compras de componentes, planejamento e coordenação da produção, gerenciamento de estocagem tanto dos computadores prontos quanto dos componentes para a fabricação, e realiza entregas dos computadores para os pedidos feitos pelos consumidores buscando respeitar o prazo estipulado.

Dentre as atividades de um agente, este artigo aborda a atividade de vendas de computadores. Esta atividade é executada diariamente pelo agente após o recebimento de um pacote com vários pedidos de cotações de computadores proveniente dos clientes. Cada pedido, chamado de *Request For Quotation* (RFQ), contém os seguintes atributos: a) Tipo de computador desejado (tipos 1 - 16); b) Quantidade de computadores desejada (1 - 20); c) Data de entrega (3 - 12 dias no futuro); d) Preço reserva (indica o máximo que o cliente deseja pagar pelo produto); e) Penalidade (valor que deve ser pago para cada dia de atraso na entrega. Após o quinto dia de atraso é feito o cancelamento do pedido).

Diariamente os agentes recebem um pacote de RFQs, assim são incumbidos de selecionar um subconjunto deste para efetuar ofertas, devendo definir o preço de venda do produto para cada RFQ. Ao receber os lances, os clientes somente consideram aqueles que satisfaçam os seguintes requisitos: a) o lance deve atender toda a quantidade especificada na RFQ; b) o lance deve conter a mesma data de entrega especificada na RFQ; c) o preço da oferta deve ser inferior ou igual ao preço de reserva especificado na RFQ. As ofertas que não atendem estes requisitos são automaticamente descartadas pelos respectivos clientes.

Após o recebimento das ofertas dos agentes, os clientes determinarão o ganhador de cada RFQ. O critério utilizado para determinar o ganhador é o menor preço vinculado ao lance. Em caso de empate, o ganhador será determinado de forma aleatória de acordo com as melhores ofertas de mesmo valor. No início do dia seguinte, o agente vencedor é notificado com uma ordem de compra para cada RFQ que ganhou.

2.2. TÉCNICAS DE INTELIGÊNCIA ARTIFICIAL

Esta seção é estruturada da seguinte forma: a subseção 2.2.1 descreve a técnica de aprendizagem baseada em instâncias, a subseção 2.2.2 descreve a técnica de detecção de variações e, por fim, a subseção 2.2.3 descreve o algoritmo de detecção de variações DWM.

2.2.1 APRENDIZAGEM BASEADA EM INSTÂNCIAS

Para [12], a técnica de aprendizagem de máquina baseada em instâncias elabora sua hipótese diretamente a partir das próprias instâncias de treinamento.

De forma geral a essência do método baseado em instâncias consiste na representação de instâncias de treinamento como pontos em um espaço n -dimensional. Estes pontos são definidos pelas n características que descrevem as instâncias. Cada característica é representada na forma de um par (*atributo, valor*) e todas as instâncias apresentam as mesmas características.

Será considerado o seguinte exemplo para elucidar como são representadas as instâncias no espaço n -dimensional. Para determinar se um dado dia é apropriado para a prática do tênis, o dia poderia ser representado por duas características as quais remetem a duas dimensões no espaço Euclidiano. As características poderiam ser, temperatura entre os valores 0 até 50°C e tempo, onde tempo seria, dia com sol, chuva, nublado ou com neve, os quais poderiam ser discretizado com os valores 0, 1, 2 e 3 respectivamente. Em função destas características poderiam ser determinadas duas classes, dias apropriados para a prática do tênis ou não. Cada característica representa um eixo no espaço.

O conceito chave do aprendizado baseado em instâncias está na propriedade de qualquer ponto de entrada x específico ter probabilidade de ser semelhante às propriedades de pontos na vizinhança de x [1]. Para identificar os (pontos) vizinhos mais próximos de um ponto de consulta, faz-se uso de uma métrica de distância $D(x_1, x_2)$ baseada nas características dos pontos x_1 e x_2 . Existem diferentes formas de calcular a distância entre duas instâncias, entre estas estão a distância Euclidiana, Mahalanobis e Hamming. Cada uma destas métricas apresenta qualidades particulares que faz com que uma seja mais apropriada do que outra para determinados casos [5].

Entre os algoritmos de aprendizagem baseados em instância, o IBI se apresenta como o algoritmo básico desta classe. Ele é derivado do algoritmo de classificação de padrões Nearest Neighbor [3] e apresenta algumas evoluções sobre este, tais como: a)

normalização do intervalo de valores das características, b) processamento de instâncias de forma incremental e, c) tolerância a valores ausentes de uma característica.

Os algoritmos baseados em instâncias são considerados de simples implementação, uma vez que demandam poucos parâmetros de ajustes. Entretanto, em cenários representados por grandes quantidades de dados, estes algoritmos necessitam fazer uso de mecanismos mais eficientes para encontrar os vizinhos mais próximos de um ponto de consulta x . De fato, o simples cálculo da distância até cada ponto exige um alto tempo de processamento.

2.2.2 DETECÇÃO DE VARIAÇÕES

Em determinadas situações a detecção rápida das variações de conceito (i.e. *drift detection*) é essencial. Esse recurso permite observar as mudanças de contexto que ocorrem no ambiente e em função dessa detecção, selecionar a ação mais pertinente para a situação atual de forma a minimizar o impacto negativo no sistema causado pela variação do ambiente [8].

Segundo [6], a técnica *drift detection* tem três objetivos principais: a) Detectar rapidamente as mudanças de conceitos; b) Distinguir às mudanças de conceitos dos ruídos; c) Reconhecer e lidar com contextos recorrentes.

Alguns sistemas podem captar informações erradas de variação de contexto devido a falhas em algum dispositivo. Estas ocorrências inserem ruído no ambiente, levando a percepções falsas. Estas situações consistem em anomalias que devem ser identificadas, tal identificação deve ser realizada sem afetar os conceitos adquiridos corretamente.

Outro aspecto importante nesta técnica consiste na capacidade de identificar conceitos recorrentes, ou seja, conceitos que se revezam entre estados válidos e inválidos com certo grau de regularidade. Por exemplo, o ciclo das estações do ano (i.e. primavera, verão, outono e inverno) podem ser armazenados e reutilizados em situações futuras.

A técnica *drift detection* pode ser definida como uma modificação no conceito atual. Esta técnica é esclarecida por meio da Figura 1, a qual representa uma variação entre dois conceitos.

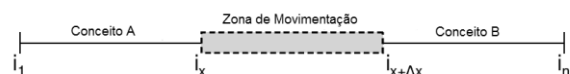


Figura 1: Mudança de conceitos [6].

Na figura 1 os elementos A e B são conceitos, e i_j representa um elemento do conjunto de instâncias. Para exemplificar a variação de conceito, considera-se inicialmente o conceito A como atual e estável, o qual se localiza entre as instâncias i_1 e i_x . Considerando a ocorrência de certo número de instâncias que modificam o estado, representado por Δ_x (i.e. número de instâncias), o conceito B pode assumir o papel de conceito atual. Como observado na figura 1, a mudança do conceito A para o B ocorre entre i_x e $i_{x+\Delta_x}$. Desta forma, quando $\Delta_x = 1$, ocorre uma mudança de conceitos brusca, e em contrapartida, quanto maior for o valor de Δ_x , a mudança de conceitos ocorrerá de forma mais suave.

2.2.3 DYNAMIC WEIGHTED MAJORITY (DWM)

Um dos algoritmos da família *drift detection* é o DWM. Ele consiste em um algoritmo de aprendizagem em tempo real para detectar variações de conceitos. O DWM mantém uma base de aprendizado que utiliza como critério de classificação. Possui um conjunto de entidades denominadas especialistas que tentam através da maioria dos votos expressarem suas opiniões. Estes especialistas são criados e destruídos dinamicamente em resposta às suas mudanças de desempenho. Segundo [10], o DWM é um algoritmo de ponderação que combina a decisões de especialistas, retornando a maioria dos votos sobre uma nova previsão.

É importante observar que o ponto que remete a técnica de *drift detection* consiste no momento de treinamento da base de conhecimento de todos os especialistas, isto reflete nas mudanças de conceitos. A base de conhecimento dos especialistas é formada por exemplos, instâncias que são passadas para o aprendizado dos especialistas.

Basicamente, o algoritmo inicia criando um conjunto de especialistas. Cada especialista pode ter seu próprio método de aprendizado, e possui também um peso de voto. A cada nova instância, o algoritmo a repassa para o conjunto de especialistas e recebe como retorno uma previsão por parte de cada especialista, ou seja, cada especialista retorna a classe que a instância pertence em função do seu conhecimento atual.

O algoritmo prediz baseado na maioria dos votos de previsão dos especialistas. Se um especialista classificar erroneamente uma instância em relação à previsão majoritária, então o algoritmo decrementa seu peso por uma constante de multiplicação β . Se achar necessário, ele também criará um novo especialista com peso 1.

O algoritmo normaliza os especialistas com pesos

de tal ordem que o maior peso será igual a um. Isso impede que qualquer especialista recém adicionado domine a tomada de decisão.

O algoritmo também remove especialistas com pesos inferiores ao limiar definido θ . O parâmetro ρ rege a frequência com que o DWM, irá criar, remover e atualizar (i.e. reduzir e normalizar o peso do voto) dos especialistas. Observa-se que o DWM gera um número variável de especialistas [6].

3. METODOLOGIA

Esta seção apresenta a forma pela qual a técnica de detecção de variações DWM foi adaptada para a atividade de venda de computadores no ambiente TAC-SCM.

De modo geral, o agente se utiliza de um módulo de venda, o qual é responsável por fazer previsões do estado atual do mercado e gerar suas ofertas aos clientes. Este módulo é composto pelo algoritmo DWM, que por sua vez é composto por um conjunto de especialistas, os quais possuem o conhecimento do mercado (i.e. conjunto de instâncias de treinamento).

O agente é composto em sua totalidade por cinco módulos, sendo eles:

- a) *Mediator*: elemento responsável pela integração entre o agente com o servidor TAC-SCM, tem como tarefa distribuir e sincronizar o fluxo de dados de entrada e saída entre o servidor e os módulos do agente;
- b) *Strategy*: este módulo tem como função determinar as metas de compra de componentes, as quais são enviadas para o módulo *Components*; também é de sua responsabilidade estimar o custo de produção de cada tipo de computador o qual é enviado para o módulo *Seller*. Adicionalmente, seleciona as RFQs mais lucrativas formando uma carteira que é enviada para o *Production*. Todas essas tarefas são feitas diariamente;
- c) *Components*: sua tarefa principal é comprar componentes a um baixo custo, respeitando as metas de compra enviadas pelo *Strategy*;
- d) *Production*: é responsável por formar a sub-carteira de RFQs em função da carteira recebida do *Strategy*, a qual deverá conter pedidos de cotação mais lucrativos e que poderão efetivamente ser produzidos em função da capacidade de produção da fábrica e dos componentes em estoque. Também é sua função escalonar a produção dos computadores maximizando a produção e minimizando os ciclos ociosos. A entrega dos produtos também é de sua

responsabilidade;

e) *Seller*: como já mencionado, este módulo é incumbido de determinar os valores de venda dos RFQs, estimando-os em função da situação atual do mercado.

A Figura 2 apresenta a composição interna do módulo *Seller*. Basicamente, este é composto por uma entidade de controle, a qual é responsável por fazer a comunicação com os outros módulos e interagir com a entidade de classificação, que envolve o algoritmo DWM.

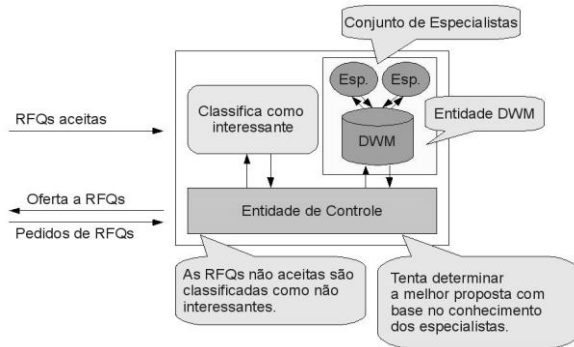


Figura 2: Estrutura interna do módulo de vendas de computadores.

Conforme descrito na seção 2.2.3 o DWM apresenta alguns parâmetros que devem ser definidos a priori. Tais parâmetros devem ser deliberados em função do problema, de forma a maximizar os resultados.

A implementação utilizada foi configurada com os seguintes parâmetros: a) ρ (período) com o valor 20, isto implica que a cada 20 instâncias de treinamentos serão realizadas as seguintes tarefas, reavaliar o peso do voto dos especialistas, eliminação dos especialistas que apresentarem valor inferior ao limiar definido θ e a criação de novos especialistas; b) β (constante de multiplicação) apresenta o valor 0.7, tal valor faz com que o peso do voto do especialista seja decrementado em 30% em função do peso corrente. O peso inicial de cada especialista é igual a 1; c) θ (limiar inferior para remoção) para este parâmetro foi atribuído o valor 0.16, com isto quando o especialista alcançar tal valor será eliminado. Com este respectivo valor para θ , no pior caso o especialista será eliminado na sétima interação (cada interação representa uma atualização do peso do voto do especialista), ou seja, quando ele já obteve mais de 120 instâncias de treinamento.

Desta forma, o processo de execução do algoritmo DWM adaptado para a atividade de vendas de computadores ocorre nas seguintes etapas:

- Treinamento: As ordens de compra ganhas relativas às RFQs do dia anterior são utilizadas para treinar os especialistas, esses especialistas são criados e destruídos dinamicamente conforme seu desempenho. Uma RFQ que deu origem a uma ordem de compra se transforma em uma instância de treinamento interessante, caso contrário, torna-se uma instância não interessante, assim há duas classes. Nessa etapa, todos os dados necessários para formar a instância de treinamento estão disponíveis.
- Classificação: Cada RFQ é classificada em função do conhecimento dos especialistas e cada especialista possui um peso associado. Esse valor corresponde ao voto. Quando o algoritmo DWM solicita a classificação dos especialistas, eles retornam como resposta a classe a qual pertence aquela RFQ. Em função da soma dos pesos dos votos é determinada a classificação. A técnica de aprendizado de máquina utilizada mede a distância entre a instância que desejamos classificar em função da base de conhecimento dos especialistas pela distância Euclidiana;

A figura 3 exemplifica através de um diagrama de sequência as atividades de treinamento e classificação do algoritmo DWM, no decorrer dos dias da simulação.

Com objetivo de representar cada RFQ em função da demanda do mercado, as seguintes características foram utilizadas: a) Preço de reserva; b) Segmento do produto (alto, baixo, médio); c) Quantidade de computadores; d) Prazo de entrega (i.e. data de entrega - dia atual); e) Penalidade; f) Preço. Dentre as características, apenas o preço é desconhecido, os outros dados são extraídos da própria RFQ.

É importante salientar que a característica do preço remete diretamente ao lucro efetivo que o agente terá com a venda dos computadores. Deste modo um valor muito baixo para esta característica pode representar baixa lucratividade ou até mesmo prejuízo, entretanto, um valor muito alto, fará com que o agente dificilmente consiga vender.

Para descobrir valores competitivos de preço de venda, propõe-se o uso de uma estratégia baseada em busca. Esta estratégia baseada em busca é considerada uma heurística (i.e. método sistemático para determinar o valor). A suposição desta incógnita é de

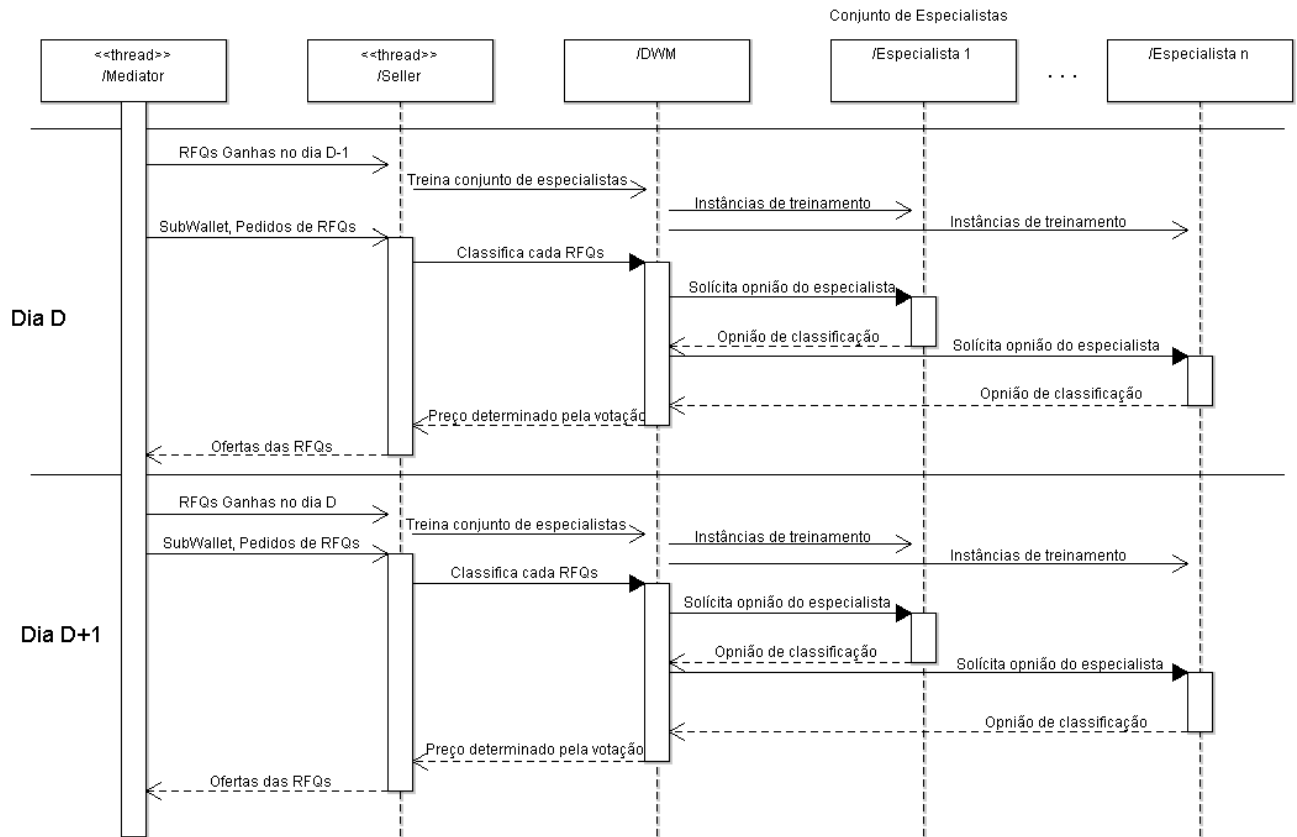


Figura 3: Diagrama de sequência do processo de execução do algoritmo DWM.

extrema importância para o classificador determinar a qual classe determinada RFQ pertence.

Se a suposição do valor para esta característica for desconexa da realidade do mercado atual, a instância dificilmente irá ser classificada como pertencente à classe de instâncias interessantes, ou seja, a suposição feita para tal característica não condiz com leitura do mercado que o conjunto de especialistas possui.

No escopo deste artigo, foram idealizadas duas estratégias, a primeira baseada em busca seqüencial e a segunda é baseada em busca binária. Ambas são descritas em maiores detalhes nas subseções seguintes.

3.1 ESTRATÉGIA DE BUSCA SEQÜENCIAL

Essa abordagem consiste em percorrer linearmente o espaço de busca entre o valor máximo disposto a ser pago pelo cliente, valor este que está contido na RFQ, até o preço de custo do produto. O preço de custo é determinado pela média aritmética do custo de aquisição dos componentes necessários para montar determinado produto dos últimos 10 dias, valor este determinado pelo módulo *Strategy*, conforme descrito na seção 3.

A cada nova RFQ, é solicitado ao algoritmo DWM para prever o preço de venda. O DWM monta a

instância de classificação com os dados vindos da RFQ e no atributo preço coloca o valor do preço de reserva. Em seguida, o algoritmo pede que cada especialista classifique a instância como interessante ou não. Caso a maioria dos votos seja da classe interessante, o preço de venda determinado é considerado válido e será feita uma oferta ao cliente com esse valor. Caso contrário, o preço de venda é decrementado em 1% e será solicitada novamente a classificação. Esse processo ocorrerá até que uma das condições seguintes seja atingida:

- O algoritmo classifica um preço de venda como interessante antes de atingir o preço de custo; i.e. os especialistas chegaram a um consenso e determinaram o preço de venda;
- O preço de venda é decrementado até o preço de custo sem que seja classificado como interessante. Neste caso, a oferta será feita com preço de custo. Aqui existe a suposição que os especialistas não possuem conhecimento suficiente para determinar o preço de venda.

Essa técnica apresenta um custo computacional elevado, pois é necessário um número muito grande de

interações para alcançar seu objetivo. A complexidade dessa técnica no pior caso é da ordem de $O(n)$, onde n é o intervalo do preço de reserva até preço de custo dividido por 1% do preço de reserva. Por exemplo, se o preço de reserva for igual a 2.500 e o preço de custo igual a 2.000, o espaço de busca possui 20 elementos dado por $(2.500-2.000)/(0,01*2.500)$.

A figura 4 ilustra um espaço de duas dimensões que remete a um exemplo de uma instância com duas características, tipo dos computadores e o custo. Cada ponto representa uma instância de treinamento no espaço bidimensional.

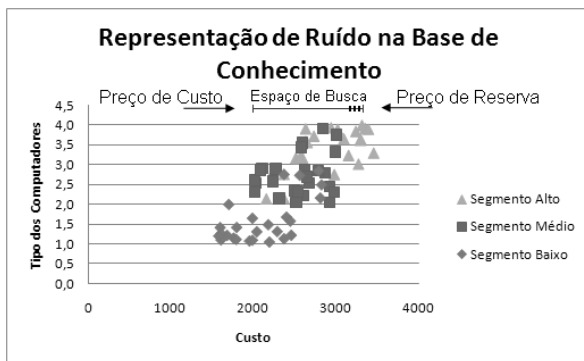


Figura 4: Representação da base de conhecimento de um especialista, representando-se apenas duas características.

Esta estratégia está suscetível a uma grande taxa de erro, interpretações errôneas em função de ruídos, pois ela percorre linearmente o espaço de busca no intuito de classificar uma dada instância como interessante. A figura 4 ilustra a busca linear em função do custo, e também representa regiões do espaço bidimensional que possuem ruído.

Especificamente a figura 4 ilustra três regiões, os segmentos alto, médio e baixo. Cada ponto (instância) é composto por 2 atributos [custo, tipo dos computadores], respectivamente são representados pelos eixos x e y no gráfico. Quanto maior a semelhança nos valores que compõem estes atributos mais próximos às instâncias são representadas no espaço, delimitando regiões semelhantes. Com o passar do tempo e com a mudança do mercado os valores dos atributos tendem a mudar. Estas mudanças fazem com que as instâncias mudem drasticamente sua região no espaço, gerando ruídos.

A geração ruído na base de conhecimento dos especialistas pode ser explicada pelo seguinte exemplo: em um dado momento os valores para os atributos custo e tipo dos computadores do segmento alto ficam na faixa $[(2.500-3.500), (3-4)]$

respectivamente, do segmento médio $[(2.000-3.000), (2-3)]$ e do segmento baixo $[(1.500-2.500), (1-2)]$.

Com a mudança do mercado os valores dos atributos são alterados, onde antes determinada faixa de valores poderia representar o segmento alto com a mudança de mercado representará o segmento médio. Com isso a base de conhecimento dos agentes pode representar instâncias de classes distintas em regiões muito próximas ocasionando os ruídos.

Diante destas observações pode-se deduzir que o simples fato da RFQ ser classificada como interessante uma única vez pela comunidade de especialista, a classificação não abona sua lisura.

3.2 ESTRATÉGIA DE BUSCA BINÁRIA

Mediante os problemas apresentados na estratégia de busca seqüencial buscou-se uma nova abordagem pela pesquisa binária. Nesta busca, são realizadas sucessivas divisões do espaço de busca a procura de uma classificação interessante. Como na primeira técnica proposta, o espaço de busca é dividido entre o preço de reserva e o preço de custo do produto.

O espaço de busca é dividido ao meio ao se iniciar a classificação. O preço de venda neste ponto é passado ao classificador DWM. Se resposta desta classificação pertencer à classe interessante a busca continuará sendo realizada no intervalo superior, caso contrário, será realizada no intervalo inferior. Por exemplo, se o preço de reserva for igual a 2.024 e o preço de custo igual a 1.000, o espaço de busca é igual a 1.024. O espaço de busca é dividido ao meio, resultando no preço de venda 1.512 dado por $(1.000 + 1.024/2)$. O preço de venda é inserido na instância a ser classificada e caso o resultado desta classificação seja interessante, o novo espaço de busca será entre 1.512 e 2.024. Caso contrário, o espaço de busca será de 1000 até 1.512. Esse processo ocorrerá até se atingir o valor máximo de interações definido. Assim para determinar o preço de venda é utilizado uma das duas condições seguintes:

- Em nenhuma das interações a votação dos especialistas obteve a classificação como interessante, e será atribuído à RFQ o preço de custo;
- Se em alguma/algumas das interações a votação dos especialistas resultar interessante, então atribui-se à RFQ o preço de venda obtido na última interação que a classificou como interessante;

A complexidade dessa abordagem é da ordem de $O(\log_2 n)$, onde n é o tamanho do espaço de busca.

Assim se o espaço de busca for igual 1.024, tem-se exatamente 10 interações, ou seja, $\log_2(1.024)$.

Esta estratégia também busca minimizar o problema de ruído na classificação dos pedidos de cotação, tendo como virtude a cobertura de uma maior extensão do espaço de busca, em função dos saltos consecutivos em torno de uma dada região de interesse, assim torna-se possível detectar a qualidade das instâncias de uma dada região ou a detecção de ruídos através desta aproximação suave. A figura 5 semelhante à figura 4 representa as instâncias em um espaço bidimensional, a figura 5 também ilustra como ocorrem às sucessivas divisões do espaço de busca no intuito de uma classificação mais precisa e robusta, afim de filtrar regiões que apresentam ruídos.

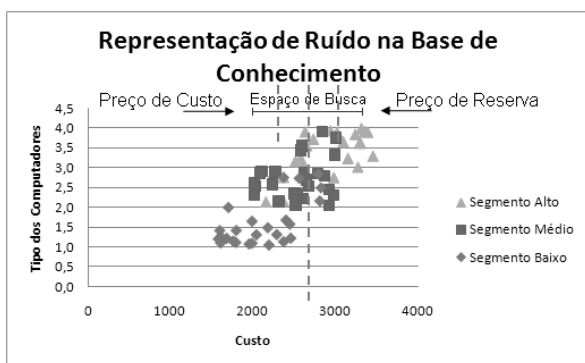


Figura 5: Representação da base de conhecimento de um especialista, representando-se apenas duas características.

4. RESULTADOS

Esta seção apresenta os testes realizados para validar o método proposto, bem como a forma que eles foram realizados e seus respectivos resultados.

Para averiguar se a adaptação do método proposto neste artigo é válida para determinar o preço de venda de pedidos de cotações, foram realizados 10 experimentos práticos. Foi idealizado um cenário com 6 competidores concorrentes vinculados ao simulador TAC-SCM.

Para a realização dos experimentos foram utilizadas 3 classes distintas de competidor-agentes, sendo eles compostos pelos mesmos módulos *Mediator*, *Strategy*, *Components* e *Production*, conforme descrito na seção 3. Eles apenas se diferem pelo módulo de venda, *Seller*.

Foi utilizada esta metodologia no intuito de minimizar a influência de atividades como, compra de

componentes e produção de computadores, em relação à atividade de venda, realizada pelo módulo *Seller*. Que é constituído pelo método proposto.

A primeira classe de agentes utiliza a estratégia de lucro fixo para determinar o preço de venda. Em função do preço de custo é acrescido um valor constante de 10%, assim ele sempre busca uma margem de lucro fixa. A segunda classe de agentes utiliza a estratégia de busca seqüencial, conforme descrita na seção 3.1. Por fim, a terceira classe de agentes emprega a estratégia de busca binária conforme descrita na seção 3.2.

Nos experimentos práticos não foram realizadas comparações entre o método proposto neste artigo, em relação a soluções propostas em outros agentes encontrados na literatura. A justificativa pelo qual não foi realizada esta comparação é explicada pela alta relação entre as atividades que o agente é incumbido a realizar. A comparação do agente como um todo pode comprometer a análise de desempenho da atividade específica de venda de computadores.

A utilização de uma margem fixa de 10% nos agentes da classe de estratégia de lucro fixo tem por intuito produzir baixa lucratividade para todos os agentes. Assim podemos avaliar se os agentes das outras classes conseguem aprender a situação do mercado sem conhecer as ofertas dos concorrentes, apenas com o feedback das próprias RFQs dos pedidos ganhos. Pelo fato dos agentes disporem dos mesmos módulos, eles disputam na maioria das vezes pelas mesmas RFQs, deixando a competição mais acirrada.

Devido ao fato dos agentes com estratégia de lucro fixo possuírem a mesma solução em todos os módulos, a probabilidade deles gerarem valores de oferta iguais para as RFQs é ínfima. Em função dos valores estocásticos gerados pelo simulador na representação dos fornecedores de componentes.

A utilização da técnica do DWM consiste na detecção de variações de demanda do mercado de forma geral, averiguando a oscilação do mesmo. De forma intrínseca o algoritmo consegue detectar as propostas dos oponentes e gerar preços competitivos objetivando maior lucratividade de forma dinâmica.

O gráfico 1 reflete os dados quantitativos dos experimentos realizados. Foram realizadas 10 competições, em cada uma havia 4 agentes da classe de estratégia de lucro fixo, 1 agente da classe estratégia de busca seqüencial. E 1 agente da classe estratégia de busca binária.

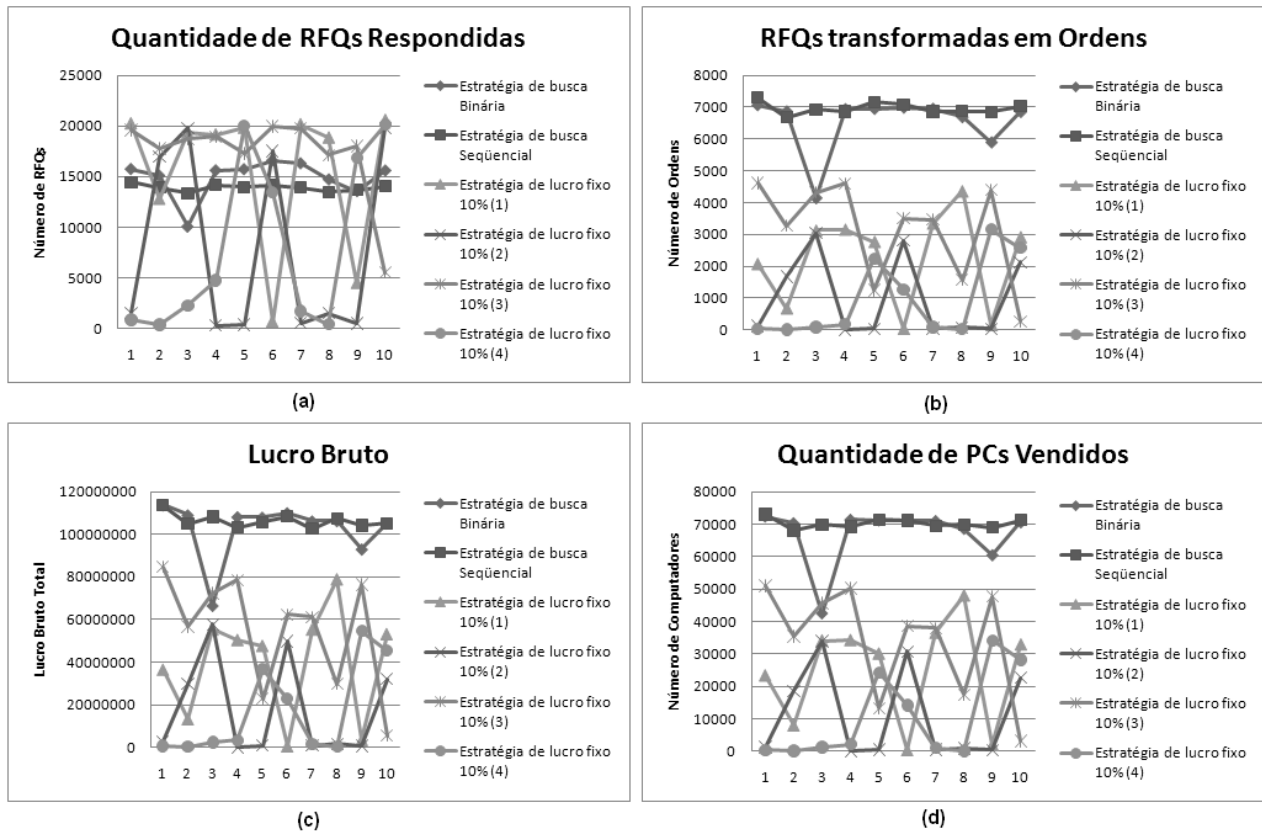


Gráfico 1: Dados quantitativos obtidos nos experimentos. Gráfico 1(a): Representa a quantidade de RFQs respondidas por cada agente. Gráfico 1(b): Representa a quantidade de RFQs que foram transformadas em ordens de compra. Gráfico 1(c): Representa o lucro bruto com a venda dos computadores. Gráfico 1(d): Representa a quantidade de computadores vendidos.

O gráfico 1(a) representa a quantidade de RFQs que cada agente ofertou. Os agentes com estratégia fixa respondem maior número de RFQ, isto se explica pelo fato que os agentes desta classe têm dificuldade em transformar estes pedidos de cotação em ordens de compra. Com isso os outros módulos do agente raciocinam (e.g. verificam a ociosidade da fábrica) e buscam aumentar a sub-carteira para aumentar o número de vendas.

As classes de agentes com estratégia de busca seqüencial e binária apresentam menor quantidade de RFQs, isto é elucidado pelo fato deles conseguirem transformar os RFQs em ordens de compra com maior facilidade. Com isso o módulo *Production* restringe a sub-carteira de RFQs para que não ultrapasse a capacidade máxima de produção.

Os dados apresentados pelo gráfico 1(b) complementam as informações do gráfico 1(a) e reforçam a interpretação obtida pelos dados anteriores. As estratégias de busca seqüencial e binária necessitam de menor número de RFQs para transformar em maior número de ordens. Isto significa que estas duas estratégias conseguem ter melhor desempenho que a estratégia fixa, pois conseguem

vender maior número de computadores.

A interpretação obtida pela análise do gráfico 1(c) mostra a supremacia das duas classes de agentes com estratégia de busca seqüencial e binária, sendo que estes dados revelam o montante total em dinheiro arrecado.

O gráfico 1(d) quantifica o número de computadores vendidos, estes valores remetem diretamente a quantidade de RFQ transformadas em ordens, como descrito na seção 2.1.

5. CONSIDERAÇÕES FINAIS

Em virtude dos dados apurados pela experimentação prática relatados na seção 4, é possível concluir que o método proposto obteve excelentes resultados. Conseguindo detectar as mudanças do mercado, e estabelecendo preço de venda mais competitivo que estratégias mais simplistas como a de lucro fixo.

Os resultados satisfatórios são explicados pelo fato

que abordagem adaptada consegue detectar preço médio de cada segmento do mercado. Deste modo suas ofertas tendem a ser competitivas, assim obtendo grande parcela do mercado. É importante salientar que o método proposto não apenas baixa o preço das ofertas a um valor inferior, ele procura o melhor preço. Isto é percebido pelo gráfico 1(c) onde é possível averiguar a grande diferença de arrecadação de dinheiro no acumulado de cada simulação. Isto demonstra a grande lucratividade da abordagem proposta.

Outro ponto importante que deve ser considerado é a competição direta entre as estratégias de busca seqüencial e binária. Utilizou-se esta abordagem, onde ambas as estratégias competem juntas no intuito de demonstrar que elas alcançam resultados muito semelhantes, como é possível observar no gráfico 1. Isto é explicado pelo fato que ambas utilizam o método proposto. As estratégias apenas se diferem na forma como fazem a suposição dos preços no momento da classificação.

Este artigo não tem por objetivo confrontar às estratégias de busca seqüencial e binária para averiguar qual é a melhor, cada qual apresenta vantagens e desvantagens. O foco deste artigo encontra-se em demonstrar como adaptar a abordagem proposta em [6] em situações onde não se tem todos os valores para os atributos das instâncias.

Outra observação que dever ser feita é sobre a estratégia de busca seqüencial. Esta estratégia se desdobra em relação ao seu concorrente direto, estratégia de busca binária. Esta sutil diferença demonstrada pelo gráfico 1(c), é elucidada pela sua tática agressiva para determinar os preços, ou seja, sempre é considerada a primeira classificação como interessante, conforme descrito na seção 3.1. Assim ela sempre tenta maximizar o seu lucro, buscando o teto máximo do preço em função do conhecimento dos especialistas.

Em contrapartida a estratégia de busca binária é mais comedida, buscando valores mais refinados. Esta característica traz 2 benefícios eminentes, o primeiro remete ao sucessivo refinamento do preço, que em ambientes mais competitivos ou ruidosos esta propriedade é de extrema importância. O segundo benefício é em função da redução do tempo computacional, característica muito importante que deve ser levada em consideração já que o ambiente de simulação TAC-SCM apresenta tempo finito para as realizações das tarefas dos agentes.

Este artigo também contribui e inova no sentido de aumentar a gama de problemas que podem ser

atacados com este método proposto. Assim é demonstrado que não há necessidade de se utilizar somente esta técnica em problemas que dispunham a priori todos os valores para os atributos que compõem às instâncias.

REFERÊNCIAS

- [1] Aha, D.W., Kibler, D., Albert, M.K.. *Instance-Based Learning Algorithms, Machine Learning*. 6(1), Pp. 37-66, 1991.
- [2] Collins, J. Arunachalam, R. Sadeh, N. Eriksson, J. Finne, N. Janson, S. *The Supply Chain Management Game For The 2007 Trading Agent Competition*. Pittsburgh, School Of Computer Science Carnegie Mellon University, 2007.
- [3] Cover, T.M., Hart, P.E.. *Nearest Neighbor Pattern Classification*. Institute Of Electrical And Electronics Engineers Transactions On Information Theory, 13, 21-27, 1967.
- [4] COYLE J.J., BARDI, E.J., LANGLEY, C.J., *Supply Chain Management: A Logistics Perspective*. South-Western College Pub; 8 Edition, 2008.
- [5] Deza, E., Deza, M., *Dictionary Of Distances*. Elsevier, ISBN 0444520872, 2006.
- [6] Enembreck, F. Tacla, C. A. Barthès, J. P. *Learning Negotiation Policies Using Ensemble-Based Drift Detection Techniques*. International Journal On Artificial Intelligence Tools - World Scientific Publishing Company, 2009.
- [7] Gama, J., Medas, P., Castillo, G., Rodrigues, P. *Learning With Drift Detection*. Proc. Of The 17th Brazilian Symposium On Artificial Intelligence – Sbia' 04, LnaI 3171, Ana L.C. Bazzan And Sofiane Labidi (Eds.), Springer, Pp.286-295, Brazil. Isbn 3-540-23237-0, 2004.
- [8] Helmbold, D. P., Long, P. M. *Tracking Drifting Concepts By Minimizing Disagreements*. Machine Learning, 14(1):27-46, 1994.
- [9] Knuth, D. *The Art Of Computer Programming*, Volume 3: Sorting And Searching, Third Edition. Addison-Wesley. ISBN 0-201-89685-0. Section 6.2.1: Searching An Ordered Table, Pp. 409–426, 1997.
- [10] Kolter, J. Z; Maloof, M. A. *Dynamic Weighted Majority: A New Ensemble Method For Tracking Concept Drift*. Ieee Conference On Data Mining,

2003.

- [11] Mitchell, T. *Machine Learning*. New York: Mcgraw Hill, 1997.
- [12] Russell, S. Norvig, P. *Inteligência Artificial*. Elsevier Editoria Ltda, 2004.
- [13] Shoham, Y., Leyton-Brown, K.. *Multiagent Systems: Algorithmic, Game-Theoretic, And Logical Foundations*. Cambridge University Press, 2009.
- [14] Simchi-Levi, D., Kaminsky, P., Simchi-Levi, E. *Designing And Managing The Supply Chain*. Mcgraw-Hill Higher Education, 2000.
- [15] Tranvouez, E. Ferrarini, A. *Multiagent Modelling Of Cooperative Disruption Management In Supply Chains*. IEEE, 2006.