
ALGORITMO DE ROTEAMENTO ADAPTATIVO PARA O BALANCEAMENTO DE CARGA EM REDES DE TELECOMUNICAÇÕES

Aluizio F. R. Araújo*
aluizioa@sel.eesc.sc.usp.br

Karla Vittori*
karlav@sel.eesc.sc.usp.br

*Universidade de São Paulo, Departamento de Engenharia Elétrica, Av. Trabalhador São-carlense, 400, CEP 13566-590, São Carlos - SP, Brasil.

ABSTRACT

This paper presents an adaptive routing algorithm, called Q-Agents, based on the integration of three learning strategies combined with some mechanisms to increase its speed of adaptation. The strategies were: Q-learning, dual reinforcement learning and learning based on ant colony behavior. The proposed algorithm is composed of a set of simple mobile agents that communicate indirectly between themselves and cooperate to find the best paths through the network. The agents select the routes in a distributed way and update information used in this task incrementally. Q-Agents has been applied to a telephone network from *British Telecom* and the mean percentage of lost calls by it was compared with the percentage of two algorithms based on ant colony behavior. The experiments comprised variations of the network traffic patterns, load level and topology and use of noise in information to select the routes. Q-Agents performed better than its competitors, presenting higher capacity of adaptation to the considered situations.

KEYWORDS: Telecommunications networks, routing, Q-learning, ant-based agents.

RESUMO

Este trabalho apresenta um algoritmo de roteamento adaptativo, denominado Agentes-Q, baseado na inte-

Artigo submetido em 15/05/00

1a. Revisão em 14/12/01; 2a. Revisão em 28/05/02

Aceito sob recomendação do Ed. Assoc. Prof. José R. Piqueira

gração de três estratégias de aprendizagem combinadas com alguns mecanismos para aumentar sua velocidade de adaptação. As três estratégias são: aprendizagem-Q, aprendizagem por reforço dual e aprendizagem baseada no comportamento de colônias de formigas. O algoritmo proposto é composto por um conjunto de agentes móveis simples, que se comunicam indiretamente e cooperam na busca dos melhores caminhos na rede. Os agentes selecionam as rotas de forma distribuída e atualizam incrementalmente as informações utilizadas na escolha dos caminhos. Agentes-Q foi aplicado a uma rede telefônica da *British Telecom* e a porcentagem média de chamadas perdidas por ele foi comparada a de dois algoritmos baseados no comportamento de colônias de formigas. Os experimentos envolveram variações nos padrões de tráfego, nível de carga e topologia da rede e a utilização de ruído na informação usada na seleção das rotas. Agentes-Q obteve melhor desempenho que seus competidores, apresentando maior capacidade de adaptação às situações consideradas.

PALAVRAS-CHAVE: Redes de telecomunicações, roteamento, aprendizagem-Q, agentes baseados em formigas.

1 INTRODUÇÃO

As redes de telecomunicações são responsáveis pelo envio de informação entre pontos de origem e destino. As condições de operação da rede, como os padrões de tráfego, nível de carga e topologia, sofrem diversas variações ao longo do tempo, algumas delas não previstas, o

que torna o ambiente dinâmico e complexo. Dentre os diversos elementos que participam do processo de transmissão de informação, destaca-se o sistema de roteamento, que seleciona as rotas a serem percorridas pela informação ao longo da rede, atuando também na sua condução ao destino desejado. O roteamento deve enviar as mensagens considerando objetivos dos usuários e restrições de operação da rede. Assim, este mecanismo desempenha um papel fundamental no controle de operação da rede, afetando diretamente o seu desempenho.

Este trabalho apresenta um algoritmo de roteamento para redes de telecomunicações, denominado Agentes-Q, baseado na integração de três estratégias de aprendizagem aliadas a mecanismos que facilitam a adaptação do roteador às condições da rede. As abordagens utilizadas foram: a aprendizagem-Q (Watkins, 1989; Watkins e Dayan, 1992; Kaelbling et alli., 1996; Sutton e Barto, 1998), aprendizagem por reforço dual (Goetz et alli., 1996) e aprendizagem baseada no comportamento coletivo de formigas (Colorni et alli., 1991; Dorigo, 1992; Dorigo et alli., 1991, 1996, 1999). Deste modo, o algoritmo proposto é composto por um conjunto de agentes móveis simples, que se comunicam indiretamente e cooperam na busca dos melhores caminhos sobre a rede. Os agentes escolhem as rotas considerando o balanço entre os nós que constituem as menores rotas e aqueles que se encontram menos carregados.

Os experimentos realizados envolveram variações dos principais parâmetros de operação da rede: padrão de tráfego, nível de carga e topologia. Além disto, foi considerado ruído nas informações usadas para seleção das rotas. O desempenho dos agentes-Q foi comparado, em termos da porcentagem média de chamadas perdidas, ao de dois algoritmos baseados no comportamento de colônias de formigas, denominados Controle Baseado em Formigas (ABC - *Ant-based Control*), (Schoonderwoerd et alli., 1996, 1997) e Controle Baseado em Formigas Espertas (Bonabeau et alli., 1998), os quais obtiveram desempenho superior a algoritmos utilizados atualmente pela *British Telecom*. Os agentes-Q apresentaram maior capacidade de adaptação às condições consideradas que os dois algoritmos analisados.

As características do algoritmo proposto também foram comparadas às de alguns sistemas dinâmicos e distribuídos utilizados com sucesso pelas seguintes empresas: i) a britânica *British Telecom*; ii) a japonesa NTT *Japan (Nippon Telegraph and Telephone)* e iii) a norte-americana AT&T. Os algoritmos considerados realizam automaticamente o roteamento alternativo das mensagens enviadas sobre a rede quando o caminho direto não se encontra disponível.

Este trabalho é organizado da seguinte forma: na Seção 2, são descritas as características fundamentais do problema de roteamento em redes de telecomunicações, juntamente com os principais algoritmos de roteamento utilizando aprendizado e aqueles alternativos usados em algumas empresas de telefonia. Na Seção 3, o algoritmo proposto é apresentado em detalhes, seguido na Seção 4 pelos experimentos realizados e resultados obtidos. Na Seção 5, uma discussão sobre o trabalho realizado é apresentada e finalmente, na Seção 6, as conclusões obtidas e as perspectivas de trabalhos futuros são mostradas.

2 ROTEAMENTO EM REDES DE TELECOMUNICAÇÕES

O roteamento em redes de telecomunicações é modelado como um grafo não direcionado $G = (R, T)$, no qual cada nó no conjunto R representa um roteador, caracterizado por sua capacidade de processamento e/ou armazenamento de mensagens. Cada arco no conjunto T representa uma linha de transmissão bidirecional que interconecta dois roteadores e se caracteriza por sua capacidade de transmissão de mensagens (Bertsekas e Gallager, 1992).

Os roteadores devem transmitir as mensagens aos destinos desejados maximizando o desempenho da rede e minimizando seus custos de operação (Bellman, 1958). Os algoritmos de roteamento possuem as seguintes funções (Steenstrup, 1995): (1) Reunião de informação sobre o estado da rede e do tráfego e distribuição entre os roteadores da rede; (2) seleção das rotas a serem percorridas pelas mensagens a partir da informação adquirida e (3) condução das mensagens através dos caminhos selecionados. A forma como estas funções são realizadas depende da tecnologia de rede utilizada. Informações sobre as condições da rede e de seu tráfego são armazenadas em tabelas de roteamento, uma para cada nó da rede. Então, quando um dado nó necessita enviar uma mensagem, ele consulta sua tabela para escolher o nó vizinho que constitui a melhor opção de acordo com o estado da rede. O envio das mensagens pode ser realizado de duas formas: através da comutação de circuitos ou da comutação de pacotes. Na primeira estratégia, a rota é selecionada e os recursos da rede para transmissão da mensagem são reservados antes do envio da informação. Tal reserva permanece válida até o final da comunicação. No segundo caso, as mensagens são divididas em unidades menores, denominadas pacotes, e o próximo nó a ser percorrido é selecionado a cada nó visitado, sem reserva prévia de recursos. Assim, cada pacote pertencente a um mesmo par origem-destino pode seguir uma rota diferente na rede.

Os sistemas de roteamento podem ser classificados basicamente como: (1) estáticos ou dinâmicos e (2) centralizados ou distribuídos. Os roteadores estáticos são aqueles em que a seleção das rotas é realizada previamente, baseada na demanda do tráfego e esta escolha permanece inalterada durante a operação da rede. Em alguns algoritmos, rotas alternativas podem ser utilizadas em casos de falhas ou faltas em roteadores ou enlaces. Por outro lado, os roteadores dinâmicos selecionam os caminhos de acordo com as condições atuais da rede. A escolha das rotas pode ser realizada de três formas (Ash, 1997): (1) em intervalos de tempo pré-determinados, caracterizando os algoritmos dependentes do tempo; (2) de acordo com os resultados produzidos pela rota percorrida, algoritmos dependentes do evento e (3) baseada no status atual da rede e do tráfego, algoritmos dependentes do estado. Os algoritmos dependentes do tempo são considerados métodos de roteamento pré-planejado, enquanto os dependentes do evento e do estado são considerados métodos de roteamento dinâmico em tempo real, que constitui a forma mais elaborada de roteamento.

Os algoritmos centralizados são aqueles em que as funções de roteamento são realizadas por um único roteador, que seleciona todas as rotas da rede e as envia aos demais roteadores. Esta estratégia necessita de alta capacidade de processamento da rede, uma vez que considera informação global, o que compromete seu desempenho em caso de falha ou falta no roteador central. No caso dos algoritmos distribuídos, todos os roteadores da rede realizam suas ações de forma independente, compartilhando a informação necessária a seleção das rotas, a qual é usualmente local.

As redes de telecomunicações atuais necessitam de algoritmos que sejam capazes de aprender a lidar com suas condições dinâmicas e estocásticas, agindo de forma autônoma e flexível. Neste sentido, são descritos a seguir alguns algoritmos de roteamento baseados em aprendizagem, alguns baseados em rotas alternativas que têm sido utilizados em empresas de telefonia, como também aqueles que constituem a base do algoritmo proposto e outros que tiveram seu desempenho comparado ao do nosso sistema.

2.1 Algoritmos de Roteamento Dinâmicos em Tempo Real e Distribuídos

A evolução das tecnologias e equipamentos utilizados pelas redes de telecomunicações nas últimas décadas permitiu o desenvolvimento de algoritmos de roteamento dinâmicos e distribuídos, capazes de acompanhar as variações das condições de operação das redes. Neste sen-

tido, foram desenvolvidos diversos algoritmos que modificam continuamente as melhores rotas sobre a rede, de acordo com os efeitos da utilização das mesmas sobre as condições da rede, apresentando a característica de aprendizado.

Os algoritmos considerados constituem os principais sistemas usados em empresas e aqueles baseados na aprendizagem-Q e no comportamento coletivo de formigas, que constituem a base do algoritmo proposto. Eles foram selecionados devido ao seu bom desempenho em relação a outros algoritmos também utilizados em sistemas reais de telecomunicações.

2.1.1 Roteamento Alternativo

Alguns algoritmos de roteamento dinâmico e distribuído consideram dois tipos de caminho sobre a rede: i) um caminho direto, formado por uma linha de transmissão que interliga origem e destino de uma chamada e ii) um caminho alternativo, formado por duas ou mais linhas. Deste modo, os algoritmos desta categoria são utilizados em redes altamente conectadas. Dentre eles, existem os algoritmos dependentes do evento ou do estado. Alguns exemplos da primeira categoria são: o Roteamento Dinâmico Alternativo (DAR - *Dynamic Alternative Routing*), implementado na *British Telecom* em 1996 (Gibbens, 1988) e o Roteamento Dependente do Tempo e do Estado (STR - *State- and Time-Dependent Routing*), utilizado na *NTT Japan* em 1992 (Yamamoto et alli., 1991). Estes algoritmos procuram enviar uma chamada inicialmente pelo caminho direto. Se este caminho não possui nenhum circuito disponível, então o último caminho alternativo utilizado com sucesso, o qual encontra-se armazenado na tabela de roteamento do nó origem, é selecionado. Caso este caminho não possua circuitos disponíveis, a chamada é perdida e um novo caminho alternativo é selecionado aleatoriamente. No algoritmo STR, que constitui uma melhoria do DAR, escolhe-se um conjunto de caminhos alternativos de acordo com a capacidade da rede e nível de tráfego em determinados intervalos de tempo. A cada chamada enviada, seleciona-se dentro deste conjunto a rota que possui um número de circuitos disponíveis, seja na primeira ou segunda linha, menor que um dado limiar, denominado limiar de congestionamento.

Os algoritmos DAR e STR também utilizam reserva de circuitos, que consiste no envio de uma chamada por um caminho alternativo somente quando o número de circuitos disponíveis em cada uma de suas duas linhas é superior a um dado valor. Deste modo, eles realizam o balanceamento de carga sobre a rede, evitando que os caminhos alternativos se tornem congestionados.

Os algoritmos dependentes do evento são eficientes com relação a perda de chamadas, porém não tanto quanto aqueles dependentes do estado (Ash, 1997). Isto ocorre porque os primeiros incorporam somente informação do passado na seleção das rotas, enquanto os últimos consideram também informação atual sobre o estado da rede em sua escolha.

O principal algoritmo de roteamento dependente do estado é o Roteamento de Rede em Tempo Real (RTNR – *Real-time Network Routing*), implementado na AT&T (Ash, 1995, 1997; Ash et alli., 1991; Mitra, 1993). Neste método, os caminhos alternativos a serem usados a cada chamada enviada são os menos carregados. O primeiro caminho a ser selecionado é o direto. Caso ele não possua um circuito disponível, o nó origem solicita ao nó destino uma lista contendo a situação de cada um de seus nós vizinhos. Ao receber esta lista, a origem compara suas informações sobre estes com aquelas enviadas pelo destino e seleciona o caminho com duas linhas que possui o maior número de circuitos disponíveis. Se nenhum caminho alternativo puder ser usado, a chamada é perdida. Este método é utilizado na rede de serviços integrados da AT&T e apresentou melhor desempenho que o algoritmo anteriormente utilizado nesta empresa, o Roteamento Dinâmico Não-Hierárquico (DNHR – *Dynamic Nonhierarchical Routing*), que constitui um sistema de roteamento dinâmico pré-planejado (Ash, 1995, 1997).

2.1.2 Roteamento Baseado na Aprendizagem-Q

A aprendizagem-Q tem sido aplicada com sucesso ao problema de roteamento em redes de telecomunicações, devido a sua capacidade de aprender a lidar com um meio ambiente sem um modelo prévio sobre o seu comportamento e de se adaptar a variações das condições deste meio. Desta forma, obtém-se algoritmos baseados somente em informação local sobre a rede, onde o aprendizado ocorre de forma autônoma através da interação entre um agente e seu meio. O primeiro algoritmo de roteamento baseado na aprendizagem-Q é denominado Roteamento-Q (Littman e Boyan, 1993; Boyan e Littman, 1994) e foi aplicado a uma rede de comutação de pacotes. O Roteamento-Q constitui um sistema distribuído e adaptativo, no qual os valores armazenados na tabela de roteamento dos nós são representados por valores-Q e estão relacionados com o período de tempo estimado para alcançar um determinado destino através de cada um de seus vizinhos. Assim, $Q_i(j, d)$ representa a estimativa do tempo para alcançar o nó destino d a partir do nó i através de seu vizinho j . Quando o agente necessita enviar um pacote a um dado destino d , ele seleciona o nó vizinho j que possui menor valor-Q. Ao

enviar o pacote, o agente recebe a informação sobre o período de tempo estimado pelo vizinho j para alcançar o nó destino d . Esta informação, juntamente com o tempo decorrido na espera do pacote na fila do nó i e na sua transmissão através da conexão (i, j) , é utilizada na atualização do valor-Q referente ao vizinho selecionado:

$$\Delta Q_i(j, d) = \eta[(q_i + s + t) - Q_i(j, d)] \quad (1)$$

na qual:

η = taxa de aprendizagem;

q_i = tempo decorrido na fila do nó i ;

s = tempo decorrido na transmissão pela conexão (i, j) ;

$t = \arg \min_k Q_j(k, d)$ no qual k é vizinho do nó j .

O bom desempenho deste algoritmo estimulou o desenvolvimento de novos sistemas de roteamento baseados na aprendizagem-Q. Dentre eles, destaca-se o Roteamento-Q com Reforço Dual (Kumar e Miikkulainen, 1997), que adiciona a atualização do valor-Q descrita anteriormente à aprendizagem por reforço dual (Goetz et alli., 1996). Este mecanismo permite que dois valores-Q sejam atualizados a cada movimento do agente, um deles na direção de sua trajetória e o outro no sentido inverso a esta. Assim, quando o nó j recebe um pacote de seu vizinho i , cuja origem é o nó o e o destino é o nó d , o agente atualiza, além de $Q_i(j, d)$, o valor-Q referente ao movimento do agente na direção inversa de sua trajetória, baseado no valor-Q armazenado na tabela do nó i em direção a origem o :

$$\Delta Q_j(i, o) = \eta[(q_j + n) - Q_j(i, o)] \quad (2)$$

na qual:

q_j = tempo a ser decorrido na fila do nó j ;

$n = \arg \min_l Q_i(l, o)$, no qual l é vizinho do nó i .

Este algoritmo apresentou maior velocidade de adaptação às condições da rede, obtendo desempenho superior ao Roteamento-Q sob variações do nível de carga da rede. Outra versão deste algoritmo foi posteriormente desenvolvida (Kumar, 1998), a qual utiliza valores de confiança para determinar a taxa de aprendizado dos valores-Q. Sua adaptação foi mais rápida e mais eficiente que a do Roteamento-Q com Reforço Dual sob diversas condições da rede.

2.1.3 Roteamento Baseado no Comportamento Coletivo de Formigas

O comportamento de insetos sociais tem provocado interesse de diversos cientistas (Deneubourg et alli., 1983), devido à sua capacidade de realizar coletivamente tarefas complexas, apesar de constituírem indivíduos simples. Em particular, podemos destacar o processo de busca e captura de alimento realizado por colônias de formigas, através do qual os menores caminhos em direção à fonte de alimento são descobertos. Isto é possível através do comportamento individual aleatório dos indivíduos da colônia e da cooperação entre eles através de um mecanismo de comunicação indireta. Ao se movimentarem sobre um ambiente, as formigas depositam no solo uma substância química denominada feromônio, que atrai novos indivíduos a utilizarem o caminho percorrido. Assim, os menores caminhos apresentarão maior concentração de feromônio, estimulando um número crescente de formigas a percorrê-los. Estas características do comportamento coletivo de formigas têm sido empregadas na resolução de diversos problemas de engenharia (Bevilacqua et alli., 1999), como o transporte de objetos por uma frota de veículos e o problema do caixeiro viajante.

Os principais algoritmos de roteamento baseados no comportamento coletivo de formigas são descritos a seguir:

- Algoritmo ABC: A informação armazenada na tabela de roteamento dos nós representa a probabilidade de um agente selecionar um determinado vizinho em direção ao destino desejado (Schoonderwoerd et alli., 1996, 1997). Assim, os agentes selecionam os nós a serem percorridos de forma probabilística. Em cada nó da rede visitado que se encontra altamente carregado, o agente é atrasado de acordo com a capacidade de processamento ociosa do nó no momento da visita. Este mecanismo visa retardar e diminuir o incremento da probabilidade de escolha de nós altamente carregados ou congestionados. Decorrido o período de atraso do agente, atualiza-se o valor de probabilidade armazenado na tabela de roteamento do nó atual com relação a escolha do último nó visitado em direção a sua origem. A atualização é inversamente proporcional ao período de tempo decorrido desde a partida do agente de sua origem até o instante atual, denominado idade do agente. Este período é composto pela soma do número de nós visitados pelo agente com o atraso sofrido em cada um destes. Assim, o sistema valoriza as menores rotas que se encontram menos carregadas. Um agente originado no nó o e que

se encontra no nó j , vindo de seu nó vizinho i , atualiza o valor de probabilidade $P_j(i, o)$ armazenado em j referente ao movimento no sentido inverso de sua trajetória da seguinte forma:

$$P_j(i, o) = \frac{P_j(i, o) + \Delta P}{1 + \Delta P} \quad (3)$$

na qual:

$P_j(i, o)$ = probabilidade de que uma formiga que possua como destino o nó o seja enviada do nó j ao vizinho i ;

$$\Delta P = \frac{a}{idade} + b; \quad (4)$$

a e b = constantes de proporcionalidade.

As rotas a serem percorridas pelas chamadas são selecionadas de forma determinística, ou seja, o nó vizinho escolhido é aquele que apresenta maior valor de probabilidade em direção ao destino desejado. Este algoritmo obteve melhor desempenho que um algoritmo clássico baseado nos menores caminhos e um algoritmo baseado em agentes móveis utilizado pela empresa *British Telecom* (Appleby e Steward, 1994).

- Algoritmo ABC com agentes espertos: Os indivíduos se movimentam sobre a rede como os agentes ABC. A diferença entre os algoritmos se encontra na forma de atualização da informação armazenada na tabela de roteamento do nó atual (Bonabeau et alli., 1998). Os agentes atualizam os valores de probabilidade com relação à escolha do último nó visitado em direção a todos os nós anteriormente percorridos até o momento, ao invés de somente com relação à origem. Assim, um agente originado no nó o , que se encontra no nó j , vindo de seu vizinho i , atualiza os valores de probabilidade referentes ao movimento em direção aos nós anteriormente visitados:

$$P_j(i, k) = \frac{P_j(i, k) + \Delta P_k}{1 + \Delta P_k} \quad (5)$$

na qual:

$$\Delta P_k = \frac{a}{idade_j - idade_k} + b; \quad (6)$$

k = nós anteriormente visitados (de o a i);

a e b = constantes de proporcionalidade.

A seleção das rotas a serem percorridas pelas chamadas também é determinística, como no algoritmo ABC. Este algoritmo apresentou melhor desempenho que o ABC sob mudanças nos padrões de tráfego da rede.

Verificamos que os sistemas de roteamento baseados somente na aprendizagem-Q ou no comportamento coletivo de formigas se complementam com relação à atualização dos valores armazenados nas tabelas de roteamento. Isto se deve ao fato de que a primeira classe considera nesta tarefa a carga dos nós selecionados em direção ao destino, enquanto a segunda valoriza o comprimento do caminho e o nível de carga dos nós percorridos em direção a origem. Esta última categoria de algoritmos ainda aumenta o volume de informação atualizada sobre os melhores caminhos da rede a cada instante de tempo, ao utilizar um conjunto de agentes móveis que analisam as condições da rede simultaneamente ao envio das chamadas.

3 AGENTES-Q: UM ALGORITMO ADAPTATIVO

O algoritmo de roteamento proposto constitui um sistema distribuído e adaptativo, resultado da combinação das três estratégias de aprendizagem: aprendizagem-Q, aprendizagem por reforço dual e aprendizagem baseada no comportamento coletivo de formigas. Além da integração destas três técnicas, foram adicionados dois mecanismos para aumentar a velocidade de adaptação do algoritmo de roteamento. A escolha da aprendizagem-Q (Watkins, 1989; Kaelbling et al., 1996; Sutton e Barto, 1998) se deve à sua capacidade de aprender a lidar com as condições de um ambiente sem possuir um modelo prévio de seu comportamento, como também de se adaptar incrementalmente às modificações ocorridas no meio. Além disto, a aprendizagem-Q possui sua convergência comprovada (Watkins e Dayan, 1992). A utilização da aprendizagem por reforço dual (Goetz et al., 1996) se encontra vinculada à sua capacidade de atualizar a informação armazenada nas tabelas de roteamento dos nós no sentido direto e inverso ao movimento do agente, tornando a adaptação do sistema mais eficaz em relação à utilização de somente a aprendizagem-Q. A escolha de um algoritmo baseado no comportamento coletivo de formigas (Coloni et al., 1991; Dorigo, 1992; Dorigo et al., 1991, 1996, 1999) está relacionada com a utilização de um conjunto de agentes móveis simples, que realizam suas funções de forma distribuída e concorrente. Os agentes cooperam entre si através de um mecanismo de comunicação indireta, o que permite a troca de informações *online* sobre a qualidade das ações realizadas. A utilização de uma política de decisão probabilística pelos agentes, baseada somente em informação local, possibilita a obtenção de soluções alternativas, que podem ser selecionadas sob variações das condições do ambiente.

As melhores características das três técnicas de apren-

dizagem descritas acima foram reunidas no algoritmo proposto e foram adicionados dois mecanismos extras. O primeiro mecanismo está relacionado com a atualização das tabelas de roteamento e diminui a chance de um nó ser escolhido quando ele se encontra processando grande volume de chamadas. O segundo mecanismo está relacionado com o roteamento das chamadas, no qual a seleção do próximo nó a ser percorrido por uma chamada combina a informação armazenada na tabela de roteamento do nó atual e o nível de carga processado pelos nós candidatos no momento da escolha.

Dentro deste contexto, o algoritmo Agentes-Q possui os seguintes componentes:

- Um conjunto de estados envolvendo todos os roteadores da rede, os quais podem ser visitados pelos agentes em seu movimento em direção ao destino desejado.
- Um conjunto de ações que estabelece as conexões utilizadas pelos agentes ao longo de sua trajetória.
- Um conjunto de agentes que se movimentam entre estados adjacentes do sistema, selecionando suas ações através de uma política de decisão probabilística.
- Dois sinais locais de retorno: um relacionado com o período de tempo dispendido pelo agente para percorrer a rota selecionada até o presente momento e o outro relacionado com o valor da capacidade de processamento ociosa do nó atual.
- Uma função de avaliação dos pares estado-ação, $Q_i(j, d)$, que representa a probabilidade de um agente que se encontra em i escolher o vizinho j para alcançar o destino d . Esta é uma estimativa, que representa o valor esperado dos retornos recebidos ao longo do caminho estabelecido a partir do estado atual do ambiente e da ação selecionada. Os valores-Q são armazenados na tabela de roteamento de cada nó i da rede.
- Uma função de avaliação dos nós vizinhos de i com relação a um determinado destino. Esta função combina os valores-Q dos nós candidatos com sua capacidade de processamento disponível no momento da seleção da rota. Assim, $A_i(j, d)$ representa a avaliação do vizinho j de i em direção ao destino d .

Os valores-Q são armazenados nas tabelas de roteamento dos nós da seguinte forma:

Tabela 1: Tabela de roteamento do nó i .

Nós destinos	Nós vizinhos			
	v_1	v_2	...	v_{m_i}
d_1	$Q_i(v_1, d_1)$	$Q_i(v_2, d_1)$...	$Q_i(v_{m_i}, d_1)$
d_2	$Q_i(v_1, d_2)$	$Q_i(v_2, d_2)$...	$Q_i(v_{m_i}, d_2)$
\vdots	\vdots	\vdots	\vdots	\vdots
d_n	$Q_i(v_1, d_n)$	$Q_i(v_2, d_n)$...	$Q_i(v_{m_i}, d_n)$

na qual:

d_j = nó destino j , $j = 1, 2, \dots, n$; $i \neq j$;

v_k = nó vizinho k , $k = 1, 2, \dots, m_i$;

m_i = número de vizinhos de i ;

n = número de nós da rede;

$Q_i(v_k, d_j)$ = probabilidade estimada pelo nó i com relação a escolha de seu vizinho v_k em direção ao destino d_j .

A seguir, são descritos o roteamento dos agentes e das chamadas realizados pelo algoritmo proposto.

3.1 Roteamento dos Agentes

Os agentes-Q agem de modo similar àqueles utilizados pelo algoritmo ABC com relação à seleção probabilística dos nós da rede a serem visitados e à atualização incremental dos valores armazenados na tabela de roteamento dos nós percorridos. As diferenças entre os algoritmos se encontram na forma de atualização das tabelas e na informação utilizada na seleção das rotas percorridas pelas chamadas.

A cada passo de tempo, um agente é lançado de cada nó da rede em direção a um destino determinado de forma aleatória. Os agentes se movimentam em direção ao destino desejado percorrendo nós vizinhos na rede. A informação armazenada na tabela de roteamento dos nós é representada por valores-Q, na qual $Q_i(j, d)$ é a probabilidade de um agente se movimentar do nó i ao nó j em direção ao destino d . Assim, a seleção do próximo nó a ser visitado por um agente é probabilística. Os agentes-Q ainda possuem uma pequena probabilidade (5%) de selecionar o nó vizinho a ser percorrido seguindo uma distribuição aleatória uniforme. Este mecanismo procura descobrir rotas alternativas, que podem ser utilizadas em casos de falha ou falta em nós e/ou conexões da rede.

Ao se movimentarem ao longo do ambiente, os agentes

armazenam em sua memória os nós origem e destino e o período de tempo decorrido desde o seu lançamento. Este período de tempo é denominado idade do agente e constitui o primeiro sinal de reforço r_j utilizado pelo sistema. Como no algoritmo ABC, a idade de um agente é determinada pela soma do número de nós visitados por ele (m) com o atraso sofrido em cada um dos nós i percorridos.

$$idade = m + \sum_{p=1}^m a_{n(p)} \quad (7)$$

na qual:

m = número de nós visitados;

$n(p)$ = número do nó visitado no passo p do agente;

a_i = atraso no nó i ; $a_i = 80 e^{-0.075 O_i}$ (8)

O_i = capacidade de processamento ociosa do nó i

($0 < O_i < 100$).

O cálculo do atraso é baseado na porcentagem da capacidade máxima de processamento do nó que se encontra disponível no momento da visita do agente (O_i) e realizado através da mesma equação empregada pelo sistema ABC, determinada empiricamente.

O segundo sinal de reforço r_i utilizado pelos agentes em sua trajetória é função da porcentagem da capacidade de processamento máxima do nó atual que está sendo utilizada C_j , denominada carga do nó. Assim, ao visitar um nó, o agente utiliza os dois sinais de reforço obtidos do ambiente ao longo de seu movimento para atualizar dois valores-Q associados à última conexão percorrida. Um destes valores encontra-se armazenado na tabela de roteamento do nó atual e se refere a escolha do nó anteriormente visitado com relação à origem do agente. O outro valor-Q encontra-se armazenado na tabela do último nó visitado e refere-se à escolha do nó em que o agente se encontra com relação ao destino desejado.

Considere um agente originado em o , que utiliza a conexão (i, j) em direção ao destino d . Ao visitar o nó j , o agente recebe os seguintes sinais de reforço da rede:

$$r_j = \beta / \text{idade do agente} \quad (9)$$

$$r_i = \left(\frac{\sigma}{C_j + \delta} \right) \quad (10)$$

nas quais:

r_i = reforço obtido do ambiente referente a ação realizada a partir do nó i ;

C_j = carga do nó j ;

β, σ, δ = constantes de proporcionalidade.

Os parâmetros β e σ são utilizados de forma a manter os reforços r_j e r_i , respectivamente, em níveis baixos, evitando oscilações. Enquanto isto, o parâmetro δ é usado para evitar um erro na operação de divisão usada no cálculo do reforço r_i quando o nível de carga assume valor zero.

A partir destes sinais, o agente atualiza dois valores-Q: (i) o valor $Q_i(j, d)$ correspondente a escolha de j a partir de i para alcançar o destino d e (ii) o valor $Q_j(i, o)$ correspondente a escolha de i a partir de j em direção ao destino o . Como visto anteriormente, cada um destes valores-Q se baseia em um sinal de reforço distinto. As atualizações são realizadas da seguinte forma:

Atualização no sentido direto:

$$Q_i(j, d) = \frac{Q_i(j, d) + \Delta Q_i(j, d)}{\sum_v Q_i(v, d)} \quad (11)$$

Atualização no sentido inverso:

$$Q_j(i, o) = \frac{Q_j(i, o) + \Delta Q_j(i, o)}{\sum_z Q_j(z, o)} \quad (12)$$

nas quais:

$$\Delta Q_i(j, d) = \alpha_i [r_i + \gamma_i \max_z Q_j(z, d) - Q_i(j, d)] \quad (13)$$

$$\Delta Q_j(i, o) = \alpha_j [r_j + \gamma_j \max_v Q_i(v, o) - Q_j(i, o)] \quad (14)$$

γ_i, γ_j = fatores de desconto;

v = nó vizinho de i ;

z = nó vizinho de j .

Os denominadores das Eqs.(11) e (12) constituem termos de normalização dos valores-Q, a fim de que a soma dos valores de todos os vizinhos de um nó com relação a um determinado destino permaneça igual a 1.

Quando o número de chamadas processadas pelo nó j é igual ou superior a 60% de sua capacidade máxima de processamento, a probabilidade $Q_i(j, d)$ é decrementada por ΔQ , ao invés de ser incrementada. Esta estratégia procura penalizar os nós que se encontram processando um grande número de chamadas no momento da visita do agente. Este mecanismo é utilizado em problemas de aprendizagem por reforço para representar uma ação que deve ser evitada e é denominado reforço negativo. Neste caso, deseja-se reduzir a probabilidade de escolha de nós

que estão utilizando grande parte de sua capacidade de processamento, para minimizar o número de chamadas perdidas.

Os valores $Q_j(z, o)$ relativos aos demais nós vizinhos de j com relação ao destino o e $Q_i(v, d)$ referentes aos demais vizinhos de i com relação ao destino d também são atualizados, de forma que a soma dos novos valores seja mantida igual a 1:

Atualização no sentido direto:

$$Q_i(v, d) = \frac{Q_i(v, d)}{\sum_t Q_i(t, d)} \quad (15)$$

Atualização no sentido inverso:

$$Q_j(z, o) = \frac{Q_j(z, o)}{\sum_u Q_j(u, o)} \quad (16)$$

nas quais:

t = nó vizinho de i ;

u = nó vizinho de j .

Esta estratégia de atualização pode ser aplicada somente quando a rede é simétrica com relação aos custos de utilização das conexões, ou seja, o custo para se movimentar do nó i ao nó j é o mesmo para o movimento no sentido inverso. Quando um agente-Q atinge o nó destino, ele é eliminado do sistema.

3.2 Roteamento de Chamadas

As chamadas são introduzidas na rede após um determinado período de tempo, denominado período de inicialização, no qual existem somente agentes na rede. Antes que a chamada seja enviada, toda sua rota é determinada, selecionando-se os nós de forma sequencial e determinística. Os nós vizinhos candidatos a serem percorridos pela chamada são avaliados no momento da seleção, de acordo com seu valor-Q armazenado na tabela de roteamento do nó atual e sua capacidade de processamento ociosa atual. Assim, a avaliação $A_i(j, d)$ do vizinho j do nó i com relação ao destino d é calculada da seguinte forma:

$$A_i(j, d) = (1 - \omega)Q_i(j, d) + \omega O_j \quad (17)$$

na qual:

ω = fator de peso ($0 < \omega < 1$);

$$\omega = \begin{cases} \omega_1 & \text{se } C_j \text{ é crescente} \\ \omega_2 & \text{se } C_j \text{ é decrescente} \end{cases}$$

$$\omega_1 = \frac{1}{1 + \exp[(-k_1 C_j) + k_2]}; \quad (18)$$

$$\omega_2 = \frac{1}{1 + \exp[(-k_3 C_j) + k_4]}; \quad (19)$$

C_j = carga do nó j ;

k_1, k_2, k_3, k_4 = constantes de proporcionalidade;

O_j = capacidade ociosa do nó j .

A partir dos valores de $A_i(j, d)$ calculados para todos os vizinhos de i , escolhe-se o nó x que possui o maior valor de avaliação:

$$x = \arg \max_j A_i(j, d) \quad (20)$$

Inicialmente, $\omega=0$ e o nó vizinho selecionado é aquele que possui maior valor-Q, constituindo assim o menor caminho. À medida que os menores caminhos são continuamente utilizados, eles se tornam propensos ao congestionamento. Então, quando a carga do nó candidato atinge o nível de aproximadamente 93% de sua capacidade máxima de processamento, ω aumenta e a capacidade ociosa do nó passa a exercer maior influência na seleção que seu valor-Q. Desta forma, os nós que estão congestionados ou próximos desta condição não são utilizados por um determinado período de tempo, durante o qual rotas alternativas são percorridas. Após este período, os melhores caminhos se tornam novamente disponíveis para novas chamadas na rede. Assim, quando a carga de um nó decresce e se encontra abaixo de aproximadamente 87% de sua capacidade máxima de processamento, ω decresce e a importância do valor-Q do nó na Eq.(16) passa a pesar mais significativamente que sua capacidade ociosa.

À medida que o caminho é construído, a capacidade ociosa O_i de cada nó i pertencente a rota selecionada é decrementada por um valor fixo. Se, no momento da construção da rota, um dos nós selecionados não possui capacidade de processamento disponível, ou o nó selecionado já foi utilizado na rota atual, a chamada é perdida. Quando a duração de uma chamada chega ao fim, a capacidade de processamento de cada nó pertencente à rota selecionada é incrementada, se tornando novamente disponível para as demais chamadas presentes na rede.

4 DESCRIÇÃO DOS EXPERIMENTOS

Os experimentos realizados compreenderam variações nos principais parâmetros da rede: padrões de tráfego, nível de carga e topologia. Além disto, as rotas percorridas pelas chamadas possuíam uma pequena probabilidade de serem selecionadas aleatoriamente, denominada

ruído. Este teste foi realizado para verificar a robustez do sistema à situação de falha nos roteadores da rede.

Nos testes realizados, o desempenho do algoritmo Agentes-Q foi comparado ao dos algoritmos baseados no comportamento coletivo de formigas ABC e ABC com Formigas Espertas. O algoritmo ABC foi selecionado devido ao seu desempenho superior em relação a um algoritmo clássico de roteamento baseado nos menores caminhos e a um algoritmo baseado em agentes móveis utilizado pela *British Telecom*. O algoritmo ABC com agentes espertos foi escolhido devido ao fato de ter obtido resultados melhores que o algoritmo original ABC. Para ambos os algoritmos, os testes de desempenho anteriormente realizados se restringiram à variação dos padrões de tráfego da rede. Este artigo relata também testes para variações nos demais parâmetros da rede.

4.1 Características da Rede Utilizada

O modelo de rede de comutação de circuitos utilizado é aquele da rede de telefonia SDH da *British Telecom* (Figura 1), o qual foi empregado pelos algoritmos ABC e ABC com agentes espertos. As características da rede foram mantidas idênticas àquelas utilizadas anteriormente.

Assim, o modelo da rede se constitui em um grafo de 30 nós, no qual cada nó i possui as seguintes características:

- Número de identificação pertencente ao conjunto de números inteiros $[0, 29]$.
- Capacidade máxima de processamento de chamadas, C_i , igual a 40 chamadas.
- Capacidade ociosa de processamento de chamadas, O_i , $0 < O_i < 100$, que corresponde a porcentagem da capacidade máxima de processamento de chamadas que se encontra disponível.
- Conjunto de vizinhos denominado N_i , formado pelos nós adjacentes a i .
- Probabilidade de constituir um nó terminal de uma chamada, seja este origem ou destino. Estas probabilidades de chamadas são determinadas através de uma distribuição aleatória em $[0.01, 0.07]$, sendo normalizadas após sua geração. O conjunto de probabilidades de todos os nós da rede constituírem pontos terminais representa assim, os padrões de tráfego aplicados a rede.
- Tabela de roteamento, contendo número de linhas correspondente aos possíveis destinos na rede, no

caso 29, e com número de colunas determinado pelo número de nós vizinhos do nó i (N_i).

Cada laço que conecta dois nós quaisquer da rede possui capacidade de processamento de chamadas ilimitada. Deste modo, a capacidade de processamento dos nós constitui o único fator limitante à operação da rede.

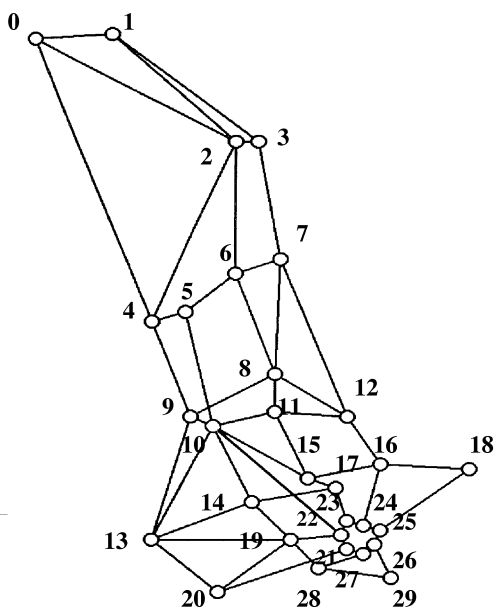


Figura 1: Rede de Telecomunicações Britânica.

Os valores dos parâmetros utilizados na atualização dos valores-Q, α_i , σ , δ , γ_i , α_j , β e γ_j e no roteamento de chamadas, k_1, k_2, k_3 e k_4 , foram determinados empiricamente. Durante os primeiros 500 passos de tempo, período em que não há chamadas telefônicas na rede, mas somente a presença de agentes, foram utilizados: $\alpha_i = 0.95$, $\sigma = 0.0$, $\gamma_i = 1.0$, $\alpha_j = 0.95$, $\beta = 6.0$ e $\gamma_j = 1.0$. Após 500 passos de tempo, quando então as chamadas são introduzidas na rede, os parâmetros do algoritmo assumem os seguintes valores: $\alpha_i = 0.08$, $\sigma = 0.05$, $\delta = 0.1$, $\gamma_i = 1.0$, $\alpha_j = 0.2$, $\beta = 8.0$ e $\gamma_j = 1.0$. Inicialmente, quando os agentes não possuem conhecimento sobre o meio ambiente, as correções dos valores-Q devem ser maiores, a fim de que os menores caminhos sejam rapidamente obtidos. A partir do momento em que há tráfego de chamadas na rede, as modificações destes valores devem ser menores, de forma que o sistema se adapte de forma mais suave às mudanças nas características da rede. Os parâmetros utilizados no roteamento de chamadas foram: $k_1 = 28.0$, $k_2 = 26.0$, $k_3 = 33.5$ e $k_4 = 30.0$. Os parâmetros utilizados pelos algoritmos ABC e ABC com agentes espertos na atualização dos valores de probabilidade foram: $a = 0.08$ e $b = 0.005$.

4.2 Detalhes da Simulação

Cada passo de tempo da simulação realizada envolve as seguintes etapas:

- Lançamento de um agente a partir de cada nó da rede, com destino determinado através de uma distribuição aleatória.
- Processamento de todas as formigas presentes na rede. Esta etapa compreende a escolha probabilística de um nó vizinho a ser visitado pelo agente e a transferência deste para o nó selecionado. Note que o agente pode sofrer atraso neste nó, de acordo com o nível de gerenciamento do mesmo.
- Geração de uma chamada em média por passo de tempo na rede, de acordo com uma distribuição de Poisson. Os nós de origem e destino são definidos através da distribuição aleatória de chamadas. A duração da chamada segue uma distribuição exponencial, com média de 170 passos de tempo.
- Processamento de todas as chamadas presentes na rede. Esta etapa consiste na seleção determinística dos nós a serem percorridos pela chamada, na redução de sua capacidade de processamento ociosa em 2.5% e no envio da chamada.

Os experimentos envolveram 10 conjuntos de probabilidades de chamadas, que representam os padrões de tráfego da rede. Cada conjunto de probabilidades foi utilizado para gerar a demanda de chamadas ao longo de 15.500 passos de tempo. Nos primeiros 500 passos de tempo das simulações, havia somente agentes sobre a rede, como foi realizado pelos algoritmos ABC e ABC com agentes espertos. Após este período, cada uma das seqüências de chamadas foi dividida em dois blocos, A e B, correspondentes a um período de adaptação e um período de teste, cada um com duração de 7.500 passos de tempo.

Os valores-Q armazenados nas tabelas de roteamento foram inicializados seguindo uma distribuição uniforme, ou seja, todos os vizinhos de um determinado nó possuíam a mesma probabilidade de serem escolhidos inicialmente.

4.3 Testes Realizados

Os experimentos realizados envolveram situações enfrentadas pelas redes de telecomunicações atuais. Assim, foram realizadas variações permanentes e temporárias dos principais parâmetros de operação da rede, que compreendem os padrões de tráfego, nível de carga e topologia

da mesma. Até o presente momento, os algoritmos de roteamento apresentados na literatura foram testados sob variações em apenas uma ou duas destas características, normalmente os padrões de tráfego ou o nível de carga da rede. Além disso, foi introduzido ruído nos valores-Q armazenados na tabela de roteamento dos nós.

São descritos 8 testes realizados com os três algoritmos implementados: ABC, ABC com agentes espertos e Agentes-Q. Em todos os experimentos, as variações nas condições da rede e do tráfego foram realizadas no período de teste. Em cada um dos testes, somente um dos parâmetros considerados foi modificado, a fim de verificar o efeito de cada um deles sobre o desempenho dos algoritmos. Os dois primeiros testes são iguais aos realizados por Schoonderwoerd et alii. (1996, 1997) e compreenderam a variação dos padrões de tráfego da rede, representados pela distribuição de probabilidades de chamadas. Os demais testes compreenderam a alteração permanente da carga da rede para os níveis médio e alto, a desconexão permanente ou temporária de um nó da rede e a adição de ruído no roteamento de chamadas durante todo o período de teste. Neste último caso, as rotas seguidas por todas as chamadas da rede possuíam uma pequena probabilidade de serem selecionadas aleatoriamente.

Os experimentos realizados são definidos da seguinte forma:

1. As chamadas geradas nos períodos de adaptação e teste seguiram o mesmo conjunto de probabilidades. Assim, no período de adaptação as chamadas foram geradas de acordo com o subconjunto de probabilidades 1A e no período de teste de acordo com o subconjunto 1B. Isto se repete até a utilização dos subconjuntos 10A e 10B.
2. As chamadas geradas nos períodos de adaptação e teste não pertenceram ao mesmo conjunto de probabilidades. Deste modo, utilizou-se no período de adaptação o subconjunto 1A de probabilidades de chamadas, enquanto no período de teste foi utilizado o subconjunto 2B. Esta estratégia foi seguida até a utilização dos subconjuntos 10A e 1B.
3. A carga da rede foi alterada para o nível médio (geração de 1,5 chamada em média a cada passo de tempo) durante todo o período de teste.
4. A carga da rede foi alterada para o nível alto (geração de 2 chamadas em média a cada passo de tempo) durante todo o período de teste.
5. O nó 10 foi desconectado da rede durante todo o período de teste.

6. O nó 10 foi temporariamente desconectado da rede durante o período de teste (do passo de tempo 8.000 ao 9.000) e posteriormente restabelecido.
7. Um fator de ruído igual a 5% foi utilizado no roteamento de chamadas durante o período de teste.
8. Um fator de ruído igual a 10% foi utilizado no roteamento de chamadas durante o período de teste.

4.4 Resultados Obtidos

O desempenho dos algoritmos foi monitorado durante os períodos de adaptação e teste, através da porcentagem média de chamadas perdidas obtidas a cada 500 passos de tempo, sobre os 10 conjuntos de probabilidades de chamadas gerados. Os resultados obtidos foram divididos em quatro grupos, de acordo com as variações dos parâmetros considerados. Em cada grupo, são apresentadas inicialmente tabelas contendo os valores médios percentuais de chamadas perdidas pelos algoritmos ao longo dos 15.000 passos de tempo e seus respectivos desvios-padrão. Os valores médios percentuais obtidos foram submetidos a um teste estatístico, denominado teste de hipóteses, para verificar se houve diferença significativa entre os resultados apresentados pelos três algoritmos implementados (Montgomery, 1997). A distribuição estatística utilizada foi a distribuição *t-Student*, com um nível de significância de 0.05. Posteriormente, são apresentados os gráficos com as médias de chamadas perdidas em cada teste a cada 500 passos de tempo.

4.4.1 Variação dos Padrões de Tráfego da Rede

De acordo com os testes estatísticos realizados, houve diferença significativa entre os valores médios percentuais de chamadas perdidas produzidos pelos três algoritmos nos experimentos envolvendo a variação dos padrões de tráfego da rede.

Tabela 2: Resultados obtidos sob probabilidades de chamadas inalteradas nos períodos de adaptação e teste.

Algoritmos	Média de Chamadas Perdidas		Desvio-Padrão	
	Adapt.	Teste	Adapt.	Teste
Agentes ABC	5,42%	4,59%	0,66%	0,19%
Agentes ABC espertos	4,48%	3,97%	0,52%	0,13%
Agentes-Q	1,65%	1,74%	0,26%	0,03%

Nos dois experimentos, os agentes-Q obtiveram as menores perdas, sendo seguidos pelos agentes ABC espertos

Tabela 3: Resultados obtidos sob probabilidades de chamadas alteradas nos períodos de adaptação e teste.

Algoritmos	Média de Chamadas Perdidas		Desvio-Padrão	
	Adapt.	Teste	Adapt.	Teste
Agentes ABC	5,21%	5,05%	0,58%	0,04%
Agentes ABC espertos	4,40%	4,37%	0,43%	0,05%
Agentes-Q	1,55%	1,78%	0,35%	0,03%

e estes, pelos agentes ABC. Sob probabilidades de chamadas pertencentes ao mesmo conjunto nos períodos de adaptação e teste (Tabela 2), os agentes-Q obtiveram valores de desvio-padrão menores que os produzidos pelos agentes ABC espertos em ambos os períodos, os quais foram menores que os produzidos pelos agentes ABC. A variação das probabilidades de chamadas utilizadas nos períodos de adaptação e teste (Tabela 3) provocou um acréscimo nos valores médios de chamadas perdidas pelos três algoritmos no período de teste, de forma menos acentuada no caso dos agentes-Q. Isto demonstra maior capacidade de adaptação destes agentes à mudança dos padrões de tráfego da rede considerada. Na fase de teste, os valores de desvio-padrão apresentados pelos algoritmos ABC e ABC com agentes espertos foram menores que os obtidos no primeiro experimento (Tabela 2), enquanto os dos agentes-Q foram iguais aos do teste anterior, evidenciando a capacidade dos agentes de aprender a lidar com a nova situação do ambiente.

Sob probabilidades de chamadas inalteradas nos períodos de adaptação e teste (Figura 2), os agentes ABC e ABC espertos diminuíram as perdas obtidas no período de teste, devido a sua capacidade de aprendizado incremental.

Os agentes-Q mantiveram as perdas quase inalteradas em ambos os períodos de adaptação e teste, adaptando-se de forma mais eficaz às condições da rede em relação aos seus competidores. A alteração das probabilidades de chamadas utilizadas nos períodos de adaptação e teste (Figura 3) produziu inicialmente um acréscimo nas perdas dos algoritmos ABC e ABC com agentes espertos, para em seguida estabilizarem as perdas em um determinado nível, não apresentando o decréscimo obtido no primeiro experimento. Os agentes-Q aumentaram suas perdas de modo inferior àquelas dos demais algoritmos e posteriormente mantiveram os valores médios de chamadas perdidas em um determinado patamar. Deste modo, os agentes-Q apresentaram maior velocidade e qualidade de adaptação à variação dos padrões de tráfego da rede considerada em relação aos seus

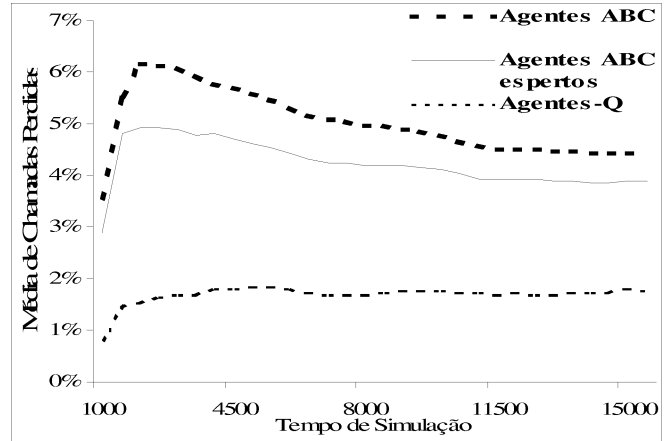


Figura 2: Desempenho dos três algoritmos a cada 500 passos de tempo sob probabilidades de chamadas inalteradas.

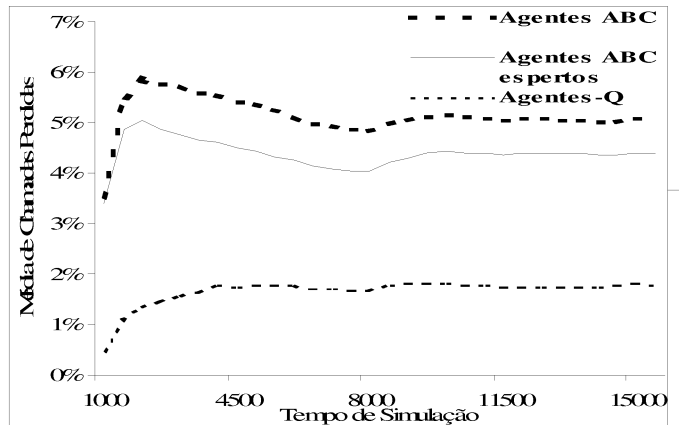


Figura 3: Desempenho dos três algoritmos a cada 500 passos de tempo sob probabilidades de chamadas alteradas.

competidores.

4.4.2 Variações do Nível de Carga da Rede

As variações do nível de carga da rede produziram acréscimos nas perdas dos algoritmos superiores à mudança dos padrões de tráfego. O aumento nos valores médios de chamadas perdidas pelos três algoritmos foi proporcional ao nível de carga considerado. Assim, as perdas produzidas pelos agentes sob o nível alto de carga durante o período de teste (Tabela 5) foram maiores que as obtidas sob o nível médio (Tabela 4). Os agentes-Q produziram os menores valores médios percentuais de chamadas perdidas nos dois experimentos envolvendo a variação da carga, sendo seguidos pelos agentes ABC espertos e estes, pelos agentes ABC.

Tabela 4: Resultados obtidos sob alteração permanente da carga da rede para o nível médio durante todo o período de teste.

Algoritmos	Média de Chamadas Perdidas		Desvio-Padrão	
	Adapt.	Teste	Adapt.	Teste
Agentes ABC	5,47%	7,67%	0,49%	0,47%
Agentes ABC espertos	4,30%	7,18%	0,42%	0,53%
Agentes-Q	1,62%	5,14%	0,38%	0,59%

Tabela 5: Resultados obtidos sob alteração da carga da rede para o nível alto durante todo o período de teste.

Algoritmos	Média de Chamadas Perdidas		Desvio-Padrão	
	Adapt.	Teste	Adapt.	Teste
Agentes ABC	5,28%	10,67%	0,79%	1,29%
Agentes ABC espertos	4,40%	10,28%	0,55%	1,27%
Agentes-Q	1,56%	8,88%	0,77%	1,47%

Os valores de desvio-padrão obtidos pelos algoritmos também sofreram acréscimos, proporcionais ao nível considerado, sendo mais acentuado no caso dos agentes-Q. Este comportamento evidencia que os agentes-Q apresentam maior sensibilidade a variação do nível de carga da rede em relação aos seus competidores.

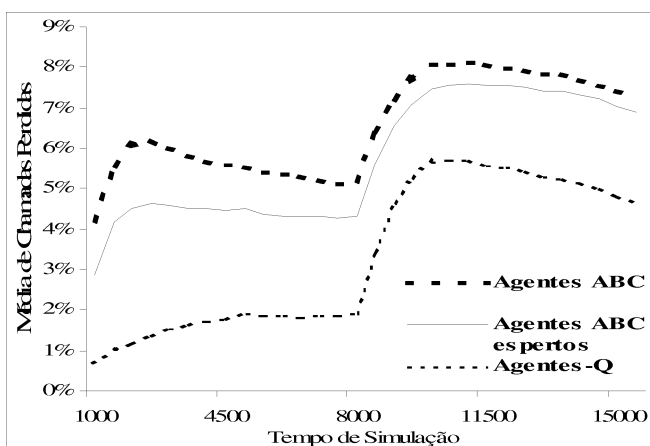


Figura 4: Desempenho dos três algoritmos a cada 500 passos de tempo sob o nível médio de carga durante o período de teste.

As perdas produzidas pelos três algoritmos sob a mudança permanente do nível de carga da rede para o nível

médio (Figura 4) e para o nível alto (Figura 5) sofreram um grande acréscimo inicialmente, de acordo com o nível considerado, sendo mais acentuado em relação às perdas obtidas no período de adaptação no caso dos agentes-Q. Após um determinado período de tempo, todos os agentes aprenderam e apresentaram decréscimo nas perdas, novamente de forma mais acentuada no caso dos agentes-Q. Deste modo, estes agentes se adaptaram melhor e de forma mais rápida que os demais às variações do nível de carga da rede consideradas.

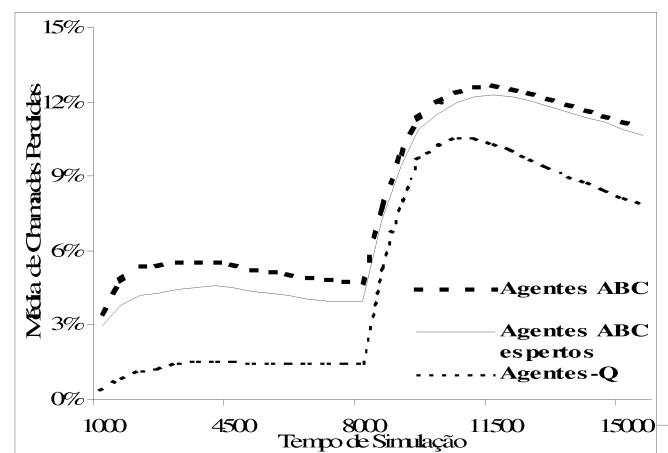


Figura 5: Desempenho dos três algoritmos a cada 500 passos de tempo sob o nível alto de carga durante o período de teste.

Contudo, o aumento de carga também deteriorou o desempenho dos agentes-Q mais rapidamente em relação ao de seus competidores.

4.4.3 Variações da Topologia da Rede

As perdas obtidas pelos algoritmos sob mudanças na topologia da rede foram significativamente diferentes. Os agentes ABC produziram as maiores perdas, sendo seguidos pelos agentes ABC espertos e estes, pelos agentes-Q.

Tabela 6: Resultados obtidos sob desconexão permanente do nó 10 durante todo o período de teste.

Algoritmos	Média de Chamadas Perdidas		Desvio-Padrão	
	Adapt.	Teste	Adapt.	Teste
Agentes ABC	5,17%	7,82%	0,58%	1,62%
Agentes ABC espertos	3,58%	6,92%	0,27%	1,67%
Agentes-Q	1,00%	5,07%	0,26%	1,99%

Tabela 7: Resultados obtidos sob desconexão temporária do nó 10 durante o período de teste (do passo de tempo 8.000 ao 9.000).

Algoritmos	Média de Chamadas Perdidas		Desvio-Padrão	
	Adapt.	Teste	Adapt.	Teste
Agentes ABC	4,75%	6,14%	0,41%	0,51%
Agentes ABC espertos	4,10%	5,22%	0,35%	0,33%
Agentes-Q	1,33%	2,63%	0,25%	0,24%

A desconexão permanente do nó 10 (Tabela 6) causou acréscimos nas perdas e nos valores de desvio obtidos pelos três algoritmos. Os agentes-Q apresentaram os maiores desvios, evidenciando maior sensibilidade a mudança deste parâmetro da rede em relação aos demais. A retirada temporária do nó 10 (Tabela 7) causou menor aumento nas perdas e nos desvios obtidos pelos agentes em relação ao teste anterior. Neste caso, os desvios obtidos pelos agentes-Q são menores que os dos demais. Isto comprova que os algoritmos aprenderam a lidar corretamente com a condição da rede, em especial os agentes-Q.

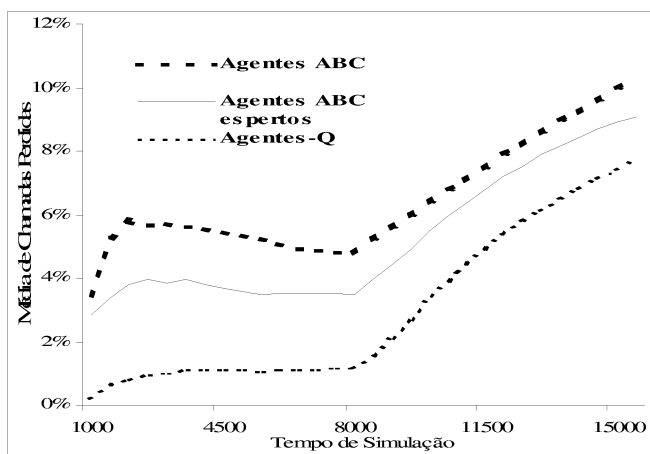


Figura 6: Desempenho dos três algoritmos a cada 500 passos de tempo sob a desconexão do nó 10 durante o período de teste.

As mudanças na topologia da rede provocaram aumentos significativos nas perdas produzidas pelos três algoritmos, porém menores que aqueles obtidos sob alteração permanente do nível de carga da mesma. Assim, a desconexão do nó 10 (Figura 6) durante todo o período de teste provocou acréscimos nos valores médios de chamadas perdidas por todos os agentes. A localização do nó 10 e o grande número de vizinhos que ele possui sugerem que a perda de chamadas sob sua des-

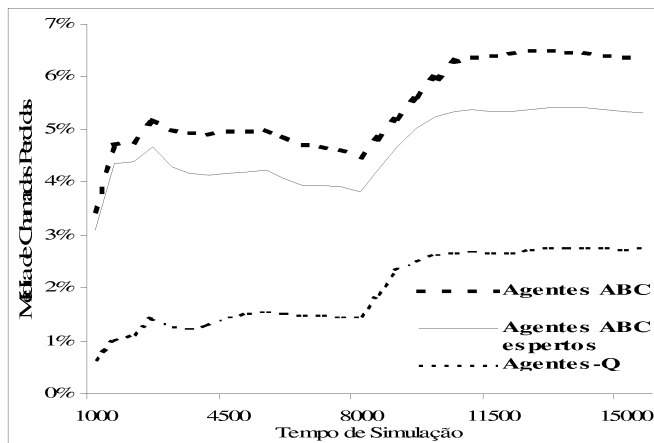


Figura 7: Desempenho dos três algoritmos a cada 500 passos de tempo sob desconexão temporária do nó 10 no período de teste, do passo 8.000 ao 9.000.

conexão permanente seria naturalmente elevada e não apresentaria tendência de decréscimo. Quando este nó foi restabelecido (Figura 7), todos os algoritmos se comportaram de modo similar, adaptando-se à nova situação e mantendo os valores de chamadas perdidas em níveis praticamente constantes. Os agentes-Q obtiveram, mais uma vez, as menores perdas em ambos os experimentos envolvendo mudanças na topologia da rede, adaptando-se de forma mais eficiente à nova condição do ambiente que seus competidores.

4.4.4 Utilização de Ruído no Roteamento de Chamadas

De acordo com os testes estatísticos realizados, os resultados produzidos pelos três algoritmos sob a utilização permanente de um ruído no roteamento de chamadas foram significativamente diferentes. A utilização de um ruído igual a 5% durante o período de teste (Tabela 8) provocou aumentos significativos nas perdas obtidas pelos três algoritmos. Os agentes ABC novamente obtiveram o maior valor médio de chamadas perdidas, sendo seguidos pelos agentes ABC espertos e estes, pelos agentes-Q.

Os desvios produzidos pelos algoritmos sofreram acréscimos nos casos dos agentes ABC espertos e agentes-Q em relação ao primeiro experimento realizado (Tabela 2), em que não havia presença de ruído no roteamento de chamadas. Os agentes ABC, por sua vez, obtiveram desvio-padrão menor que o apresentado no primeiro teste. Isto demonstra maior influência deste fator no desempenho dos algoritmos ABC com agentes espertos e Agentes-Q em relação ao algoritmo ABC. A utilização

Tabela 8: Resultados obtidos sob utilização de um ruído de 5% no roteamento de chamadas durante todo o período de teste.

Algoritmos	Média de Chamadas Perdidas		Desvio-Padrão	
	Adapt.	Teste	Adapt.	Teste
Agentes ABC	5,29%	5,36%	0,60%	0,16%
Agentes ABC espertos	4,40%	4,93%	0,43%	0,20%
Agentes-Q	1,65%	2,74%	0,27%	0,31%

Tabela 9: Resultados obtidos sob utilização de um ruído de 10% no roteamento de chamadas durante todo o período de teste.

Algoritmos	Média de Chamadas Perdidas		Desvio-Padrão	
	Adapt.	Teste	Adapt.	Teste
Agentes ABC	5,29%	6,17%	0,56%	0,52%
Agentes ABC espertos	4,48%	5,84%	0,40%	0,57%
Agentes-Q	1,56%	3,63%	0,36%	0,66%

de um ruído igual a 10% no período de teste (Tabela 9) causou maiores acréscimos nas perdas produzidas pelos três algoritmos que sob 5%. Os agentes-Q obtiveram as menores perdas, sendo seguidos pelos agentes ABC espertos e ABC. Os valores de desvio-padrão também foram maiores que os produzidos no experimento anterior.

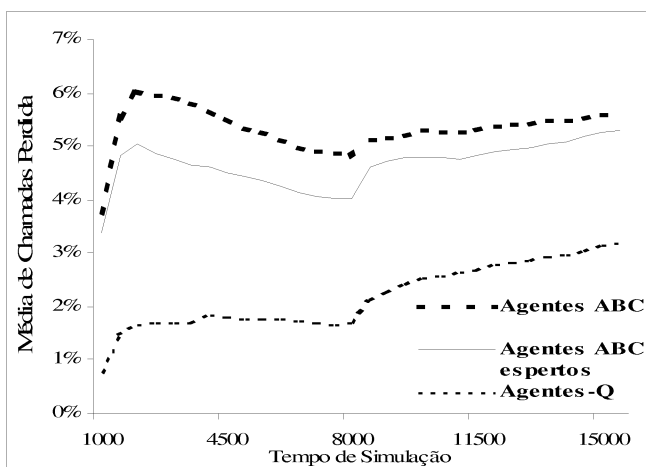


Figura 8: Desempenho dos três algoritmos a cada 500 passos de tempo sob um ruído de 5% no roteamento de chamadas durante o período de teste.

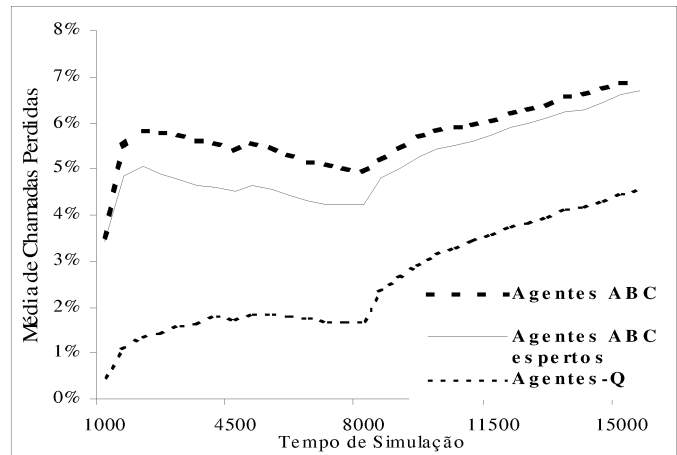


Figura 9: Desempenho dos três algoritmos a cada 500 passos de tempo sob um ruído de 10% no roteamento de chamadas durante o período de teste.

Mais uma vez, os agentes-Q apresentaram o maior desvio, seguidos pelos agentes ABC espertos e estes, pelos agentes ABC, demonstrando que o ruído causa maiores efeitos negativos no desempenho do algoritmo proposto que no dos demais.

A utilização de um ruído igual a 5% no período de teste (Figura 8) provocou acréscimos nas perdas produzidas pelos três algoritmos analisados, sendo mais acentuado no caso dos agentes-Q. Este comportamento era esperado, uma vez que o ruído é utilizado nas rotas percorridas por todas as chamadas da rede a cada passo de tempo, afetando diretamente o desempenho do sistema. Sob a utilização de um ruído de 10% durante o período de teste (Figura 9), os agentes se comportaram como no teste anterior, porém os acréscimos nos valores médios de chamadas perdidas em relação ao período de adaptação foram maiores que sob um ruído de 5%, principalmente no caso dos agentes-Q. É importante mencionar que as perdas produzidas pelos três algoritmos foram ascendentes para os dois valores de ruído utilizados. Isto evidencia a grande influência do ruído no desempenho dos algoritmos, em especial dos agentes-Q.

5 DISCUSSÃO

O roteamento dinâmico trouxe vários benefícios para as concessionárias de comunicação, como a melhoria do desempenho e da qualidade dos serviços, bem como a redução dos custos de operação. Para os consumidores, as vantagens incluem a disponibilidade de novos serviços, com aumento na qualidade e confiabilidade dos mesmos, a um custo reduzido. Os diversos algoritmos dinâmi-

cos desenvolvidos atualizam as tabelas de roteamento de acordo com um dos três parâmetros: tempo, evento ou estado da rede. A primeira categoria pertence ao roteamento pré-planejado e as demais, ao roteamento em tempo real.

Os algoritmos dependentes do estado são os que produzem as menores perdas sob diversas condições da rede (Ash, 1997). Dentre eles, destaca-se o algoritmo RTNR, utilizado com sucesso na AT&T. Este sistema é utilizado em redes altamente ou totalmente conectadas (Ash, 1991), o que também ocorre com o algoritmo dependente do evento DAR, de forma que as chamadas são enviadas por caminhos com no máximo duas linhas de transmissão. Como a rede utilizada neste trabalho possui um baixo grau de conectividade, não foi possível utilizá-la para comparar o desempenho do algoritmo proposto com aquele dos métodos DAR e RTNR sob as diversas situações consideradas. Além disto, estes algoritmos consideram no roteamento as características das linhas de transmissão e na rede utilizada neste trabalho, as informações obtidas na época de sua realização se referem as condições dos nós.

O algoritmo proposto constitui um sistema em tempo real e poderia ser classificado como dependente do evento e do estado. Isto se deve ao fato de que a seleção dos caminhos a serem percorridos pelas chamadas se baseia em dois fatores: os valores armazenados nas tabelas de roteamento e a carga atual dos nós. O primeiro fator representa a qualidade dos caminhos percorridos anteriormente pelos agentes, onde cada movimento realizado constitui um evento e o segundo aspecto considerado representa o estado imediato da rede. Neste caso, a rota selecionada constitui a menor e menos carregada num dado momento e não apenas um caminho satisfatório, obtido através da simples consulta a tabela de roteamento, como ocorre nos demais algoritmos considerados. Deste modo, os agentes-Q são capazes de selecionar rotas melhores e mais rapidamente que os demais sob diferentes situações.

O algoritmo proposto também foi aplicado a um modelo da rede de telefonia da AT&T (Vittori e Araújo, 2001), que possui número de nós superior e grau de conectividade inferior ao da rede usada neste trabalho. Seu desempenho se manteve melhor que o dos algoritmos ABC e ABC com agentes espertos sob os mesmos testes aqui apresentados, demonstrando sua capacidade de adaptação a diferentes topologias de redes. A única mudança realizada no algoritmo para esta aplicação consistiu no ajuste dos parâmetros utilizados na atualização dos valores-Q. Verificamos que os dois modelos utilizados consistem em redes com baixo grau de conectivi-

dade, caminhos de diversos comprimentos e onde a distribuição de linhas de transmissão é não uniforme.

Os algoritmos implementados neste trabalho possuem um tempo de processamento praticamente igual. Todos eles realizaram cada simulação com dez blocos de chamadas em cinco minutos em uma estação SunW, Ultra-1.

A grande dificuldade encontrada por estes algoritmos de roteamento em uma implementação real consiste na adaptação rápida e eficaz a mudanças nas condições da rede, principalmente quando os aspectos modificados retornam a sua situação habitual. Assim, este constitui um dos principais aspectos a serem considerados em nossos trabalhos futuros.

6 CONCLUSÕES

Neste trabalho foi apresentado o algoritmo de roteamento para redes de telecomunicações Agentes-Q, que constitui um sistema distribuído e adaptativo. O algoritmo proposto se baseia em propriedades de três estratégias de aprendizagem: aprendizagem-Q, aprendizagem por reforço dual e aprendizagem baseada no comportamento coletivo de formigas. Além da integração destas três estratégias, foram adicionados alguns mecanismos aos agentes-Q, a fim de obter um algoritmo eficaz diante das condições dinâmicas e estocásticas das redes de telecomunicações. Os experimentos realizados envolveram variações permanentes e temporárias dos principais parâmetros de operação da rede: padrões de tráfego, nível de carga e topologia. Além disto, foi inserido ruído nas informações armazenadas na tabela de roteamento dos nós, caracterizando uma falha nos roteadores. O desempenho do algoritmo proposto, com relação à porcentagem média de chamadas perdidas, foi comparado ao de dois algoritmos baseados no comportamento coletivo de formigas e desenvolvidos recentemente, os quais obtiveram melhor desempenho com relação a sistemas estáticos e dinâmicos. Os agentes-Q obtiveram perdas de chamadas menores que seus competidores em todas as situações consideradas, apresentando maior velocidade e qualidade de adaptação às diversas variações da rede. Entretanto, o algoritmo proposto demonstrou ser mais sensível que os demais a variações no nível de carga da rede e a presença de ruído na seleção das rotas a serem seguidas pelas chamadas.

Deste modo, uma importante questão de pesquisa será investigar novos mecanismos de otimização que devem ser anexados aos agentes-Q, de forma a melhorar sua capacidade de resposta às características dinâmicas da rede e aumentar a flexibilidade e robustez do sistema às

variações das condições da rede. O comportamento dos agentes-Q demonstrou que o compromisso entre a exploração de uma solução promissora e a exploração de soluções alternativas, que constitui uma questão essencial dos algoritmos de aprendizagem por reforço, deve ser aprimorada. Mesmo assim, o algoritmo proposto demonstrou ser robusto e adaptativo o suficiente às situações consideradas, atingindo o balanço adequado entre dinamismo e estabilidade.

Uma outra questão interessante a ser considerada é a comparação do desempenho do algoritmo proposto com o daqueles empregados nas concessionárias de telecomunicações, levando-se em conta outras características da rede, como a capacidade limitada de transmissão das linhas e tráfego heterogêneo entre os diversos componentes da rede.

REFERÊNCIAS

- Appleby, S. e Steward, S. (1994). Mobile software agents for control in telecommunications networks. *BT Technology Journal*, Vol. 12, nº 2.
- Ash, G. R. (1995). Dynamic network evolution, with examples from AT&T's evolving dynamic network. *IEEE Communications Magazine*, Vol. 33, nº 7, pp. 26-39.
- Ash, G. R. (1997). *Dynamic routing in telecommunications networks*. McGraw-Hill, New York.
- Ash, G. R., Chen, J. S., Frey, A. E. e Huang, B. D. (1991). Real-time Network Routing in a Dynamic Class-of-Service Network. *Proc. of the 13th International Teletraffic Congress*, pp. 187-194.
- Bellman, R. (1958). On a routing problem. *Quarterly of Applied Mathematics*, Vol. 16, nº 1, pp. 87-90.
- Bertsekas, D. e Gallager, R. (1992). *Data networks*. Prentice-Hall.
- Bevilacqua, R., Castaño, M. C., Monteiro, L. H. A e Piqueira, J. R. C. (1999). Estratégias adaptativas em sociedades de formigas. *Controle & Automação*, Vol. 10, nº 3, pp. 167-175.
- Bonabeau, E., Henaux, F., Guérin, S., Snyers, D., Kunts, P. e Théraulaz, G. (1998). Routing in telecommunications networks with "smart" ant-like agents telecommunication applications. *Lectures Notes in AI*, Vol. 1437. Springer Verlag.
- Boyan, J. A. e Littman, M. L. (1994). Packet routing in dynamically changing networks: a reinforcement learning approach. *Advance in Neural Information Processing Systems* Vol. 6, pp. 671-678. Morgan Kaufmann, San Mateo, CA.
- Coloni, A, Dorigo, M. e Maniezzo, V. (1991). Distributed optimization by ant colonies. *Proc. of the European Conference on Artificial Life (ECAL 91)*, pp. 134-142, Elsevier.
- Deneuborg, J. L. Pasteels, J. M. e Verhaeghe, J. C. (1983). Probabilistic behaviour in ants: a strategy of errors?. *Journal of Theoretical Biology*, Vol. 105, pp. 256-271.
- Dorigo, M. (1992). *Optimization, learning and natural algorithms (in Italian)*. PhD Dissertation, Dipartimento di Elettronica e Informazione, Politecnico di Milano, Italy.
- Dorigo, M., Maniezzo, V. e Coloni, A. (1991) Positive feedback as a search strategy. Technical Report 91-106, Dipartimento di Elettronica, Politecnico di Milano, Italy.
- Dorigo, M., Maniezzo, V. e Coloni, A (1996). The ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man and Cybernetics B*, Vol. 26, pp. 1-13.
- Dorigo, M., Di Caro, G. e Gambardella, L. M (1999). Ant algorithms for discrete optimization. *Artificial Life*, Vol. 5, nº 2, pp. 137-172.
- Gibbens, R. J. (1988). *Dynamic routing in circuit-switched networks: the Dynamic Alternative Routing strategy*. Ph.D. Dissertation, University of Cambridge.
- Goetz, P., Kumar, S. e Miikkulainen, R. (1996). On-line adaptation of signal predistorter through dual reinforcement learning. *Proc. Machine Learning: Proceedings of the 13th Annual Conference*.
- Kaelbling, L. P., Littman, M. L. e Moore, A W. (1996). Reinforcement learning: A survey. *Journal of AI Research*, Vol. 4, pp. 237-285.
- Kumar, S. (1998). *Confidence based dual reinforcement Q-routing: an on-line adaptive network routing algorithm*. Master's thesis, Department of Computation Science, University of Texas.
- Kumar, S. e Miikkulainen, R. (1997). Dual reinforcement Q-routing: an on-line adaptive routing algorithm. *Proc. of Artificial Neural Networks in Engineering*.

- Littman, M. e Boyan, J. A. (1993). A distributed reinforcement learning scheme for network routing. *Proc. of the First International Workshop on Applications of Neural Networks to Telecommunications*, pp. 45-51. Hillsdale, Erlbaum, New Jersey.
- Mitra, D., Gibbens, R. J. e Huang, B. D. (1993). State-dependent routing on symmetric loss networks with trunk reservations, I. *IEEE Transactions on Communications*, Vol. 41.
- Montgomery, D. C. (1997). *Design and analysis of experiments*. John Wiley, New York.
- Schoonderwoerd, R., Holland, O., Bruten, J. e Rothkrantz, L. (1996). Ant-based load balancing in telecommunications networks. *Adaptive Behavior*, Vol. 5, n° 2, pp. 169-207.
- Schoonderwoerd, R., Holland, O. e Bruten, J. (1997). Ant-like agents for load balancing in telecommunications networks. *Proc. of the First International Conference on Autonomous Agents*, pp. 209-216. ACM Press.
- Steenstrup, M. E. (1995). *Routing in telecommunications networks*. Prentice-Hall.
- Sutton, R. S. e Barto, A. G. (1998). *Reinforcement learning: An introduction..* MIT Press, Cambridge, MA.
- Vittori, K. e Araújo, A. F. R. (2001). Agent-oriented routing in telecommunications networks. *IEICE Transactions on Communications*, Vol. E84-B, n°11, pp. 3006-3013.
- Watkins, C. J. C. H. (1989). *Models of delayed reinforcement learning*. PhD Dissertation, Psychology Department, Cambridge University.
- Watkins, C. J. C. H. e Dayan, P. (1992). Q-learning. *Machine Learning*, Vol. 8, n° 3, pp. 279-292.
- Yamamoto, H., Mase, K. Inoue, A. e Suyama, M. (1991). A state- and time-dependent dynamic routing scheme for telephone networks. *Proc. of the 13th International Teletraffic Congress*, Copenhagen, Denmark.