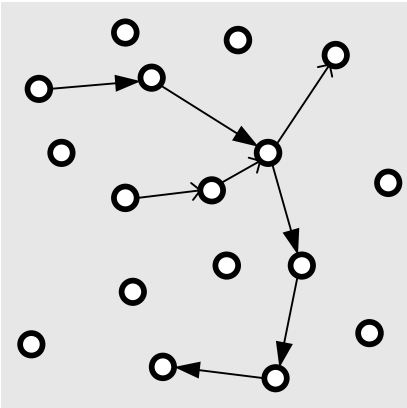
Universidade de Aveiro
2006

Departamento de Electrónica, Telecomunicações e Informática

Vítor Manuel Jesus Silva

Contributos para a Qualidade de Serviço em Redes Móveis Ad Hoc




Universidade de Aveiro
2006
Departamento de Electrónica, Telecomunicações e Informática

Vítor Manuel Jesus Silva

Contributos para a Qualidade de Serviço em Redes Móveis Ad Hoc

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Electrónica e Telecomunicações, realizada sob a orientação científica do Professor Doutor Rui Luís Andrade Aguiar, Professor Auxiliar do Departamento de Engenharia Electrónica e Telecomunicações da Universidade de Aveiro

o júri

presidente

Prof. Dr. José Luís Oliveira

professor associado do Departamento de Electrónica e Telecomunicações da Universidade de Aveiro

Prof. Dr. Edmundo Monteiro

professor associado da Faculdade de Ciências e Tecnologia da Universidade de Coimbra

Prof. Dr. Rui Luís Andrade Aguiar

professor auxiliar do Departamento de Electrónica e Telecomunicações da Universidade de Aveiro

resumo

Esta dissertação discute os problemas associados às garantias de resposta de uma rede móvel ad hoc aos seus utilizadores. A percepção generalizada é de que o problema é complexo, devido a factores como as tecnologias sem fios e a mobilidade. Após analisar em pormenor as diversas componentes das redes móveis ad hoc, conclui-se que o problema é transversal a todas as camadas de rede. A dissertação explora técnicas como a estimação da capacidade livre por nó, a criação de canais de comunicação implícitos e a emulação de uma rede com fios sobre uma rede móvel ad hoc. A partir daí, torna-se possível desenvolver um modelo geral de Qualidade-de-Serviço para redes ad hoc sem comprometer o conceito intuitivo de Qualidade de Serviço. Construído o modelo propôs-se uma implementação como prova do conceito. Essa implementação foi simulada tendo-se obtidos resultados promissores e indicando que o modelo não só é adequado a redes móveis ad hoc isoladas como também é útil para a interligação destas a redes com fios.

abstract

This work discusses the problem of providing guarantees to users of a mobile ad hoc network. It is usually acknowledged that this problem is complex due to, amongst others, the underlying wireless technology and mobility. After investigating the relevant aspects, it is stated that the problem is transversal to all network layers. This works exploits techniques such as per-node free capacity estimation, implicit channel creation and emulation of wired networks on top of mobile ad hoc networks. From there, it was possible to develop a general enough QoS model without compromising the classical concept of QoS. After developing the model, an implementation was designed as a proof-of-concept. This implementation was simulated and the results obtained are quite promising giving further indication that the model is not only suitable to stand-alone mobile ad hoc networks but also for interoperation with wired networks.

Índice de conteúdos.

1 Introdução....13
1.1. Contexto....13
1.2. Estrutura, Objectivos e Metodologias....14
1.3. Principais contributos desta dissertação....17
1.4. Agradecimentos....17
2 Redes Móveis Ad hoc....20
2.1. Conceito e oportunidade....20
Conceito....20
Questões de utilização da rede....22
Cenários de utilizador final....23
2.2. Propriedades....24
Propriedades básicas das redes ad hoc....24
Tipologia das redes ad-hoc....26
2.3. tecnologias L1 e L2 de suporte....26
2.3.1. Tecnologias possíveis....26
2.3.2. HiPeRLAN/2....28
2.3.3. IEEE802.11....29
Introdução....29
Funcionamento geral....31
Problema do nó exposto/nó escondido....34
Suporte a QdS....35
Capacidade....37
Interacções com camadas superiores....41
2.4. encaminhamento em redes ad hoc....43
2.4.1. protocolos de encaminhamento....43
Introdução....43
Propriedades de um protocolo de encaminhamento....44
2.4.2. Protocolos por-pedido....47
Conceito....47
AODV....48
DSR....51
2.4.3. Outros protocolos....53
Conceitos....53
OLSR....53
2.4.4. Análise comparativa....55
Protocolos reactivos e proactivos....55
comparação breve dos protocolos analisados....57
comparação entre AODV e DSR....59
2.4.5. Técnicas genéricas de melhoria do encaminhamento....64
Introdução....64
Optimização do DSR....65
A melhor rota....66
Multi-rota....69
3 Qualidade-de-Serviço....74
3.1. Colocação do problema....74
3.1.1. Introdução....74
3.1.2. Principais modelos implantados....80

IntServ / RSVP. 80
DiffServ. 84
3.1.3. Discussão integrada. 88
Problemas conhecidos nas redes sem fios. 88
Aplicabilidade a redes ad hoc. 90
3.2. QdS em redes ad hoc. 92
3.2.1. Introdução. 92
3.2.2. Modelos de QdS para redes ad hoc. 93
FQMM. 93
2LQoS. 96
DLite. 98
SWAN. 98
INSIGNIA. 103
3.2.3. encaminhamento de QdS. 107
AODV com encaminhamento de QdS. 108
QOLSR. 108
CEDAR. 109
Sondagem baseada em bilhetes. 111
AQOR 112
3.2.4. Questões de colaboração. 113
4 AADQ. 116
4.1. Crítica ao estado-da-arte. 116
Modelos. 116
QdS em redes ad hoc. 117
Validade de encaminhamento de QdS e sinalização específica de QdS. 119
Requisitos de um modelo de QdS. 121
Planeamento. 122
4.2. AADQ. 123
Motivação. 123
Modelo simples para a capacidade de redes ad hoc num cenário 'hotspot'. 124
Componentes do AADQ. 132
Facilidade de concretização. 139
4.3. Prova do conceito: TSQ. 140
4.3.1. TSQ 140
Motivação. 140
Funcionamento. 141
Planeamento prévio da rede. 142
Gestão de tráfego na origem. 142
Análise dos parâmetros principais. 145
4.3.2. dTSQ 146
Total distribuição do TSQ. 146
4.3.3. Simulações. 148
Objectivos. 148
Parametrização das simulações. 149
Análise dos resultados 150
5 Conclusões. 154
5.1. Conclusões principais. 154
5.2. Direcções Futuras. 155
Referências. 159

Índice de figuras.

Fig. 1. Estrutura da dissertação. 15
Fig. 2. A rede ad hoc como rede de acesso. 22
Fig. 3. Trama IEEE802.11. 31
Fig. 4. Acesso ao meio no IEEE802.11 (modo DCF). 31
Fig. 5. Diagrama temporal do modo PCF.. 32
Fig. 6. Problema do nó exposto/nó escondido 35
Fig. 7. Diagrama temporal mostrando a detecção virtual de portadora. 35
Fig. 8. Débito-entre-extremos médio numa rede ad hoc. 38
Fig. 9. Capacidade por nó numa rede ad hoc (simulação). 40
Fig. 10. Débito-entre-extremos para uma sessão TCP entre os nós (1) e (5) 42
Fig. 11. Genealogia de alguns protocolos de encaminhamento. 56
Fig. 12. Peso do tráfego de controlo do OLSR em função da densidade da rede s. 58
Fig. 13. Desempenho comparado para várias cargas do AODV e DSR . 63
Fig. 14. Distribuição da diferença de tamanho das rotas para a rota mais curta. 68
Fig. 15. Distribuição do tamanho das rotas em relação à rota mais curta. 68
Fig. 16. Comparação entre SP e MP usando derivados do DSR 72
Fig. 17. DiffServ vs IntServ. 78
Fig. 18. Arquitectura IntServ. 80
Fig. 19. Modelo de referência para um nó que implemente IntServ 81
Fig. 20. Rede DiffServ. 85
Fig. 21. Encaminhador de fronteira no DiffServ. 86
Fig. 22. Rede heterogénea. 90
Fig. 23. Modelo de referência de nó no FQMM 94
Fig. 24. Papeis dos nós no FQMM 95
Fig. 25. Simulação de FQMM 96
Fig. 26. 2LQoS 97
Fig. 27. Modelo de nó do SWAN 99
Fig. 28. Comportamento genérico de uma rede de pacotes utilizado pelo SWAN. 100
Fig. 29. Resultados do SWAN 102
Fig. 30. Modelo de referência de nó INSIGNIA 103
Fig. 31. Campo INSIGNIA no pacote IP 104
Fig. 32. Desempenho do INSIGNIA 105
Fig. 33. Efeito de suavização do INSIGNIA sobre o TCP 106
Fig. 35. 4 saltos de uma rede ad hoc. 124
Fig. 36. modelo de tráfego para um cálculo simples da capacidade de uma rede ad hoc (modelo 'hotspot') 126
Fig. 37. Taxa de perdas com carga crescente num cenário de 'hotspot'. 128
Fig. 38. Taxa de perdas com carga crescente num cenário de 'hotspot' – área 750m X 750m. 129
Fig. 39. Taxa de perdas com carga crescente num cenário de 'hotspot' – área 2000m X 2000m. 129
Fig. 40. Distribuição do atraso para 20 nós e área de 750m X 750m num cenário de 'hotspot'. 131
Fig. 41. Distribuição dos atrasos para 20 nós e duas áreas diferentes. 132
Fig. 42. Arquitectura de rede do AADQ. 134
Fig. 43. Arquitectura da entidade SE. 134
Fig. 44. Perspectiva do AADQ segundo camadas de rede. 139
Fig. 45. Algoritmo de escalonamento. 143
Fig. 46. TSQ: tratamento do tráfego. 143
Fig. 47. Regulação do TSQ sobre fluxos TCP – 2 saltos, sessão única. 145
Fig. 48. Regulação do TSQ de fluxos TCP – 2 sessões com um salto em comum. 146
Fig. 49. Resultados do TSQ. 151

Índice de tabelas.

tab. 1. Teatros de operação atómicos de redes ad hoc considerados neste trabalho. 27
tab. 2. Tecnologias de nível de ligação sem fios mais relevantes actualmente. 28
tab. 3. Família IEEE802.11 (2005)....30
tab. 4. Alguns parâmetros do IEEE802.11....33
tab. 5. Parâmetros de simulação de vários trabalhos onde se comparam protocolos de encaminhamento. 60
tab. 6. Exemplos de canais AADQ. 133
tab. 7. Canais definidos para o TSQ. 141
tab. 8. Definição dos canais usados na simulação. 148
tab. 9. Parâmetros da simulação. 150
tab. 10. Resultados do TSQ – sessões FTP de fundo nas simulações. 152

Tabela de Acrónimos.

AAAC – Authentication, Authorization, Accounting, Charging
AAAL5 – ATM Adaptation Layer 5
ACK – ACKnowledge
ADM – Add and Drop Multiplexer
AF – Assured Forwarding
AIFS – Arbitrary Inter-Frame Spacing
AIMD – Additive Increase Multiplicative Decrease
AP – Access Point
ARP – Address Resolution Protocol
ATM – Asynchronous Transfer Mode
BA – Behaviour Aggregate
BB – Bandwidth-Broker
BE – Best Effort
BP – Backoff Period
CBR – Constant Bit rate
CC – Central Controler
CDMA – Code Division Multiple Access
CL – Controlled Load
CM – Centralized Mode
CSMA – Collision Sense Médium Access
CSMA/CA – CSMA with Collision Avoidance
CTS – Clear-To-Send
CW – Contention Window
DAD – Duplicate Address Resolution
DCF – Distributed Coordination Function
DIFS – Distributed Inter-Frame Spacing
DM – Distributed Mode
DS – DiffServ Domain
ECN – Esplicit Congestion Notification
EDCF – Extended DCF
EF – Expedited Forwarding
EFM – Ethernet in the First Mile
EIFS – Extended Inter-Frame Spacing
ER – Edge Route
FCS – Frame Check Sequence
GFP – General Framing Procedure
GPS – Generalizad Processor Sharing
GS – Guaranteed Service
GSM – Global Services for Mobile
HCF – Hybrid Coordination Function
HiPeRLAN – High Performance Radio LAN
IP – Internet Protocol
IPv6 – IP versão 6
IR – Interior Router
ISP – Internet Service Provider

L1 – layer 1
L2 – layer 2
LAN – Local Area Network
MAC – Medium Access Control
MP – MultiPath
MT – Mobile Terminal
NAV – Network Allocation Vector
PAN – Personal Area Network
PCF – Point Coordination Function
PHB – Per-Hop Behaviour
PIFS – Point Inter-Frame Spacing
PPP – Point-to-Point Protocol
QdS – Qualidade-de-Serviço
QoS – Quality-of-Service
RERR – Route ERRor
RREP – Route REPly
RREQ – Route REQuest
RT – Reat-Time
RTP – Real-Time Protocol
RTS – Request-To-Send
S-D – par origem-destino
SDH – Synchronous Digital Hierarchy
SIFS – Small Inter-Frame Spacing
SLA – Service Level Agreement
SLS – Service Level Specification
SP – SinglePath
STA – STAtion
TC – Traffic Class
TCA – Traffic Conditioning Agreement
TCP – Transmission Control Protocol
TDD – Time Domain Division
TDMA – Time Division Multiplexing
TTL – Time-To-Live
TXOP – Transmission Opportunity
UDP – User Datagram Protocol
UMTS – Universal Mobile Telecommunications Service
VLAN – Virtual LAN
VoIP – Voice over IP
WAN – Wide Area Network
WCDMA – wireless CDMA
WFQ – Wighted Fair Queueing
xDSL – variante de Digital Subscriber Line
xPON – variante de Passive Optical Network

1 Introdução.

1.1. Contexto

Esta dissertação tem como ambiente de investigação as redes ad hoc. Uma rede móvel ad hoc é uma colecção de computadores móveis que têm a capacidade de, autonomamente, criar uma rede de comunicações sem assistência de quaisquer entidades centrais.

Sendo os nós móveis, usam tecnologias sem fios. Datam da década de 1970 os primeiros desenvolvimentos de redes de pacotes sobre rádio, as *Packet-Radio Networks*, no âmbito de projectos da ARPA[1]. Curiosamente, estas redes rádio foram parte do desenvolvimento da Internet tal como a conhecemos agora. A primeira demonstração do TCP foi a 27 de Agosto de 1976 em S. Francisco [101]. Um autocarro do SRI[2], em andamento, transmitiu um longo relatório por rádio para uma estação base no edifício do SRI que, por sua vez, usava a infrastrutura da ARPANET (o embrião da Internet) para fazer chegar os pacotes de dados a Boston. O TCP surgia assim como o protocolo que permitia a uma aplicação ignorar a heterogeneidade das redes e o autocarro com o transmissor rádio iniciava uma nova era nas comunicações.

As organizações militares imediatamente viram as potencialidades destas tecnologias e na década de 80 investiram no desenvolvimento de tecnologias que permitissem comunicações rádio sem recursos a estações base excessivamente vulneráveis a ataques. Conjuntamente com desenvolvimentos em teoria do sinal, nomeadamente técnicas de espalhamento de espectro muito resistentes a ataques electrónicos, e a possibilidade técnica de produzir computadores sucessivamente mais pequenos e potentes, aplicações mais sofisticadas começaram a ter oportunidade de desenvolvimento.

[1] Advanced Research Projects Agency, agora DARPA (Defense Advanced Research Projects Agency)
[2] Stanford Research Institute (http://www.sri.com)

A massificação destas tecnologias começa a dar-se na década de 90. Talvez o melhor exemplo deste fenómeno seja a proliferação do computador portátil e do PDA com interface Wi-Fi. A partir deste ponto, a investigação em redes ad hoc ganha um novo ânimo ao ser orientada também para aplicações generalistas.

Com este novo ânimo, as redes ad hoc enfrentam igualmente novos problemas. Esta dissertação aborda o problema de conseguir fornecer algum tipo de garantias a utilizadores da rede. Dado que as redes ad hoc, devido à mobilidade e às tecnologias de transmissão de sinal, são extremamente frágeis e imprevisíveis, o problema de conseguir fornecer garantias da resposta rede pode tornar-se extremamente complexo.

Especificamente, esta dissertação propõe um novo modelo de QdS para redes ad hoc baseado em pressupostos específicos. Entre eles, a noção de que uma rede ad hoc pode ser criteriosamente planeada, o que aparenta ser um contra-senso no contexto de redes ad-hoc. Nesta dissertação são derivadas algumas regras de planeamento. Entre elas, uma expressão que permite obter uma estimativa da capacidade livre da rede de forma a poder-se reservar alguma da capacidade para utilizadores que necessitem de serviços mais exigentes. O modelo foi depois simulado no ns-2 [98].

1.2. Estrutura, Objectivos e Metodologias

Na Fig. 1 está representada a estrutura desta dissertação. No capítulo 2, as redes ad hoc são apresentadas da perspectiva de identificar fraquezas e virtudes. O segundo objectivo é encontrar cenários realistas que possam simplificar o problema geral de Qualidade-de-Serviço (QdS). A identificação de que redes ad hoc podem ser administradas é um passo importante: poder haver uma entidade centralizadora e supervisionadora é importante por permitir concentrar nela parte da complexidade de gestão.

As tecnologias L2 relevantes são analisadas, ainda dentro deste capítulo. Destas, o IEEE802.11 é detalhado. Esta tecnologia é usada na maioria das redes ad hoc e, devido a isso, é directamente responsável por parte da ineficiência que se atribui às redes ad hoc. É necessário isolar as razões pelas quais isso acontece. Em particular, o problema da

capacidade destas redes é atacado com um modelo simples mas que produz resultados úteis. Os problemas específicos que se propagam para as camadas acima (e.g., difusão e TCP) são depois analisados para posteriormente encontrar soluções. A ênfase é sempre sobre o que impede a obtenção de garantias da rede e que tipo de mecanismos se podem projectar para as minimizar.

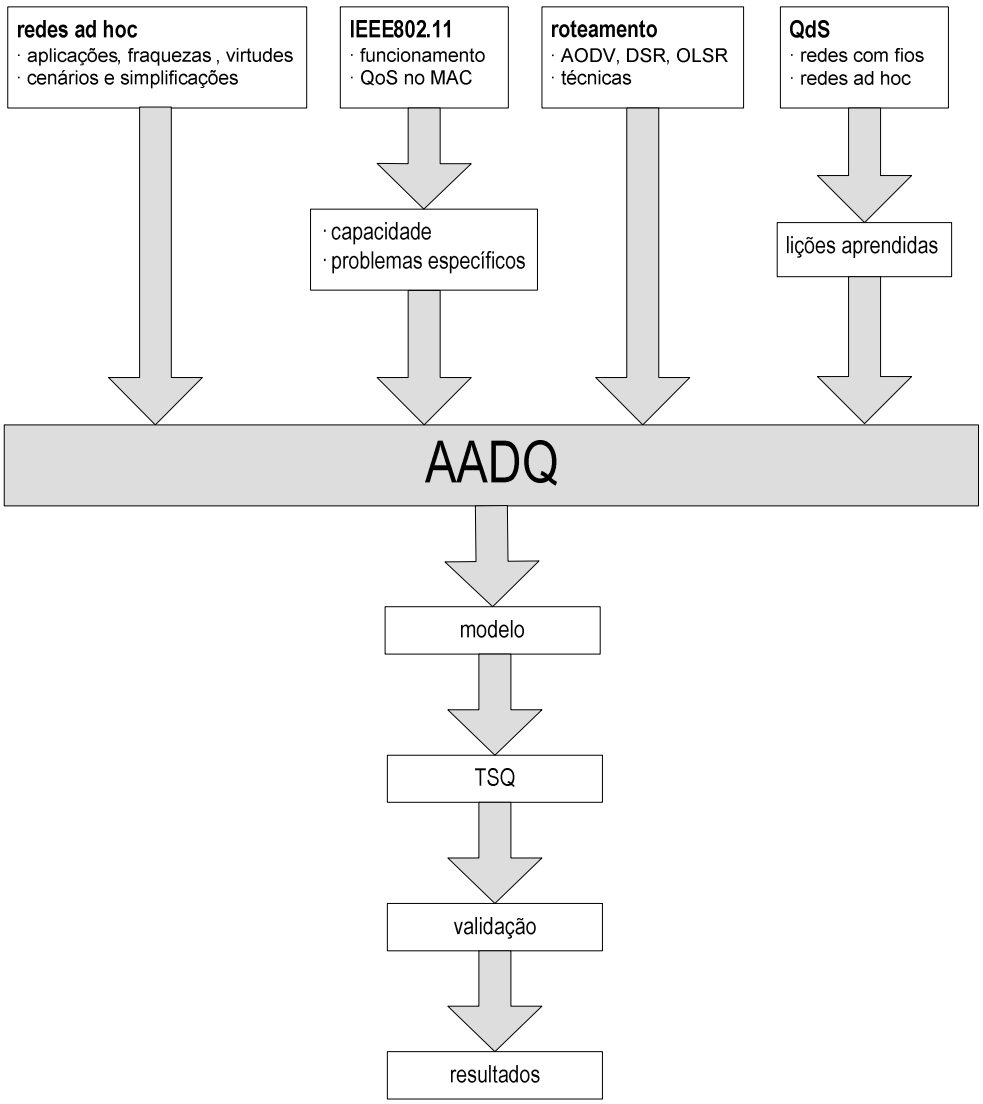


Fig. 1. Estrutura da dissertação.

De seguida, entra-se na camada de "rede" tal como é encarada no eterno modelo ISO/OSI. As especificidades das rede ad hoc ganham aqui visibilidade. A conectividade L3 ("routing") é detalhada exemplificando-a com os protocolos que se consideram mais

relevantes: DSR, AODV e OLSR. Os problemas do MAC analisados nas secções anteriores são agora omnipresentes e, mais grave, subtis. Além disso, os protocolos introduzem tanto complexidade como ineficiências, sendo necessário compreende-las. O capítulo inicia ainda o estudo de capacidade em redes ad hoc.

A estratégia para o capítulo é a seguinte:

- apresentação *objectiva* do assunto evitando entrar em discussões de eficiência e aplicabilidade. É sempre de um ponto de vista de estado-da-arte sem comentar quaisquer decisões de projecto.
- *discussão*, frequentemente integrando todo o capítulo.

No capítulo 3 o estado-da-arte em Qualidade-de-Serviço (QdS) é apresentado. O objectivo é, por um lado, aprender com o exemplo das redes com fios e, por outro, compreender onde falham as várias propostas até ao momento para QdS em redes ad hoc. Com efeito, à excepção do SWAN que demonstra alguma eficiência mas é mais uma optimização da camada de transporte do que modelo de QdS, todas as propostas ou são eminentemente teóricas, sem ser possível uma aplicação prática, ou falham quando se lhes pede garantias efectivas.

No capítulo 4, integralmente escrito de raiz, o modelo proposto é apresentado, o "Ad hoc Administered with DiffServ-like QoS" (AADQ). O "Time-Slotted QoS", a sua concretização e prova-de-conceito demonstra possuir todas as características exigidas aos mecanismos de QdS, sendo que a mais importante, a eficiência do estabelecimento de canais numa rede de comutação de pacotes, é manifestamente conseguida. Finalmente, os resultados obtidos por simulação são apresentados.

O capítulo 5 conclui a dissertação apresentando também direcções futuras.

1.3. Principais contributos desta dissertação

Durante este trabalho de mestrado, um conjunto de conceitos ou técnicas novos foram desenvolvidos. Apresenta-se de seguida os principais contributos, por ordem de aparecimento nesta dissertação:

- estudo da capacidade de uma rede ad hoc num cenário de 'hotspot'
- desenvolvimento da noção de que uma rede ad hoc pode ser planeada e regras básicas de planeamento
- técnicas de optimização do protocolo de encaminhamento DSR
- um modelo de QdS para redes ad hoc geral (o AADQ), sem restrições *a priori* e capaz de interoperação com modelos de QdS para redes fixas
- uma implementação flexível do modelo (TSQ/dTSQ) que demonstra versatilidade
- um algoritmo de escalonamento capaz de controlar, com parâmetros simples e numa rede ad hoc, o débito de fontes de tráfego sobre protocolos de transporte como o TCP; consegue também linearizar o débito do TCP, mesmo para longas distâncias.

1.4. Agradecimentos

Quero agradecer primeiro aos meus colegas da Prisma e Siemens: à Prisma o apoio quando pedi para usufruir de um dia por semana para trabalhar neste mestrado; na Siemens, onde actualmente colaboro, agradeço por mo terem cedido.

Em particular, devo agradecimentos ao Engº Carlo Marques e ao Engº Fernando Correia, meus superiores directos no grupo de Hardware da Siemens. Aos meus colegas, devo especial agradecimento ao José Augusto Neves, ao Nelson Marques e ao Luís João Carvalho que, por alturas de um importante projecto, compreenderam que o mestrado se tinha tornado numa prioridade absoluta (a escrita da dissertação).

Ao João Pedro Marques, "flatmate" e também colega da Siemens, agradeço as ajudas pontuais que me foi prestando, particularmente em problemas relacionados com Linux, onde o seu à-vontade e a sua experiência me pouparam muito tempo.

À Fátima Gonçalves e ao Filipe Cunha, meus colegas de mestrado, por termos criado o grupo das pessoas-que-não-têm-vida-pessoal-porque-ou-se-está-a-trabalhar-no-mestrado-ou-se-está-com-sentimentos-de-culpa-por-não-estar.

Ao Prof. Dr. Rui Aguiar, por me ter dado oportunidade e confiança para desenvolver as minhas ideias, especialmente no momento em que propus um trabalho diferente de mestrado.

Finalmente, porque o menor e contornado esconde o maior e contornando, dedico este trabalho às três mulheres da minha vida: mãe, irmã e Natacha.

Vítor Jesus Silva

Agosto 2006

2 Redes Móveis Ad hoc.

Neste capítulo faz-se um resumo selectivo do estado-da-arte. Começa-se por elaborar sobre o conceito de rede ad hoc e as suas presumíveis aplicações no futuro. De seguida, comenta-se algumas opções existentes para a implementação das camadas L1 e L2, discutindo com pormenor o IEEE802.11. É levantada uma discussão comparada dos protocolos de encaminhamento mais relevantes. Finalmente, abordam-se tópicos sobre encaminhamento como multi-rota.

2.1. Conceito e oportunidade

Conceito.

O conceito de rede ad hoc surge naturalmente quando existe um conjunto de entidades com as seguintes características:

(i) entidades móveis, podendo usar o espaço livremente,

(ii) capazes de comunicar entre si, em difusão ou direccionado[1]

(iii) a tecnologia de meio físico não impõe nenhuma limitação quanto à organização das entidades comunicantes no espaço

(iv) cada entidade tem um alcance de transmissão e recepção limitado

Uma das características essenciais e diferenciadoras das redes ad hoc é o ponto (iii). As redes 'tradicionais', em virtude de serem suportadas por cablagem ('ethernet', redes ópticas, etc.) têm sempre uma organização imposta pela topologia física do 'site' de implementação. Mesmo muitas tecnologias sem fios (como o WCDMA) requerem essa organização.

[1] Não são usados os termos 'broadcast' e 'unicast' porque são termos mais frequentemente usados no contexto de encaminhamento na camada de rede. A ênfase aqui é na transmissão de sinal.

Formalmente, uma rede móvel ad hoc é uma colecção de computadores móveis que têm a capacidade de, autonomamente, criar uma rede de comunicações sem assistência de quaisquer entidades centrais. Portanto, a nenhum utilizador pode ser atribuído um papel permanentemente distinto dos restantes.

Classificar uma rede ad hoc em função das já existentes não é simples de um ponto de vista funcional, na medida em que há sempre tendência para se enquadrar cada conceito de acordo com a tecnologia física que usa – SDH é uma rede óptica, o GSM é uma rede celular, a 'ethernet' é ponto-a-ponto, etc.. Talvez a melhor forma de catalogar cada tecnologia no domínio das redes é aceitar que a tecnologia serve uma qualquer necessidade[1] e ordenar a classificação em função do cenário sugerido por essa necessidade.

Assim, o cenário-problema que uma rede ad hoc resolve é de um grupo de dispositivos com um alcance curto de comunicação que, por via de saltos múltiplos[2] ("multi-hop"), geram conectividade dentro da área de interesse. Se a área de interesse for a Internet global, pode imaginar-se uma rede ad hoc como rede de acesso à WAN.

O objectivo final ainda é, como o é para qualquer rede de comunicação, fazer chegar, de forma controlada, dados de um extremo ao outro, tipicamente em longas distâncias (definidas à custa da ordem de grandeza do alcance da tecnologia L1).

Um potencial cenário de utilização está ilustrado na Fig. 2. Na criação do cenário, foram usadas várias tecnologias para melhor evidenciar o papel de uma rede ad hoc como tecnologia que *não pode ser substituída por nenhuma outra existente*. Imediatamente se compreende o papel da rede ad hoc: sem ela, como garantir conectividade dos utilizadores no primeiro troço, sem recorrer a redes com gestão centralizada?

Convém referir um tipo especial de redes ad hoc, fora do âmbito desta dissertação, definida no UMTS. Esta arquitectura prevê a utilização das estações móveis em modo multi-salto no sentido de aumentar a capacidade da rede e cobertura. É bem conhecido que o UMTS, e genericamente qualquer sistema celular que se baseie no CDMA, tem

[1] ou a iminente sensação dela, como tantas vezes acontece no negócio das redes de comunicação
[2] Nesta dissertação é usada frequentemente a designação "multi-salto" como sinónimo de "saltos múltiplos".

cobertura e capacidade interdependentes e são função da carga momentânea. O modo ad hoc pode aumentar a capacidade, num dado instante, da célula CDMA. Essa aplicação não será objecto desta dissertação.

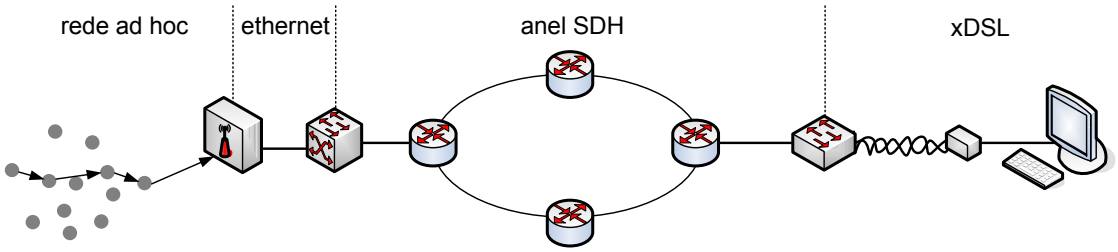


Fig. 2. *A rede ad hoc como rede de acesso.*

Faz-se um último comentário sobre a natureza das redes ad hoc. Identifica-se muitas vezes uma rede ad hoc quando há operação distribuída. Contudo, pode não ser assim, i.e., pode haver entidades com papéis diferentes numa rede. Há sempre um elevado grau de distribuição numa rede ad hoc, evidentemente, mas por si, a distribuição não é um requisito necessário. Ao longo da dissertação, este tópico será recorrentemente discutido.

Um ponto relativamente assente é que uma rede ad hoc não deve ser encarada como uma rede de trânsito, i.e., uma rede que transporta tráfego que nem é originado nem é terminado dentro da rede [1]. Embora tecnicamente seja possível, no estado-da-arte actual constituiria um cenário que potenciaria as ineficiências das redes ad-hoc sem explorar devidamente as suas vantagens.

Questões de utilização da rede.

É frequente centrar as redes ad hoc no seu carácter de espontaneidade (*ad-hoc* significa isso mesmo)[1]. Mas esta visão pode comprometer uma visão mais global da natureza e utilidade das redes ad-hoc. A facilidade ou não de utilização da rede não define por si a rede ad hoc. Se são, ou serão, redes "espontâneas", isso dever-se-á a um esforço adicional na concepção da rede e à custa da experiência entretanto adquirida no campo das redes de comunicação,

[1] Os anglo-saxónicos diriam *on-the-fly*.

nomeadamente mecanismos de autoconfiguração. O ideal será, em termos simplistas, abrir o 'laptop', o 'software' do computador tratar da burocracia da rede (registo, autenticação, aquisição de endereço, descoberta de serviços, encaminhamento, etc.) e estar imediatamente "ligado" ao computador remoto. Mas, mais uma vez, a espontaneidade da rede está no esforço de coordenação de todos os nós participantes no sentido de criar conectividade geograficamente alargada.

Já a colaboração dos nós em prole da boa utilização da rede é discutível. Um forte argumento contra a implantação das redes ad hoc consiste em não ter garantias de que um nó irá aplicar os seus recursos (energia, alcance, processamento, etc.) no auxílio dos outros quando não tirar proveito imediato. Em questões ligadas a Qualidade de Serviço (QdS), o problema ainda se agrava mais, quando é exigido um controlo apertado dos recursos disponíveis próprios e de terceiros.

Pelo menos inicialmente, a solução passa por admitir boa-fé e colaboração ilimitada dos intervenientes. Qualquer rede é extremamente vulnerável a interferências externas e ataques do tipo Negação-de-Serviço (Denial-of-Service, DoS) e, de forma geral, as redes actuais são extremamente frágeis. Se funcionam bem é porque geralmente há boa utilização delas [2]. Deve, contudo, ser referido que o problema da cooperação tem um lugar próprio de investigação no domínio das redes ad hoc. Este tópico será abordado adiante.

Cenários de utilizador final.

São comummente referidas as seguintes aplicações para redes ad hoc:

(i) grupo de utilizadores genéricos: utilizadores móveis, num local pequeno, que partilham dados entre si ou usam a rede para acesso à Internet.

(ii) aplicações profissionais: militares, agentes de segurança, etc., ligados entre si em zonas sem infrastruturas.

(iii) veículos em estradas. Foi recentemente criado um grupo de trabalho para estudar formas de aplicar as redes ad hoc a veículos: iniciativa *car-to-car* [3].

(iv) redes de sensores. Espera-se que a tecnologia futura permita desenvolver sensores de tamanho e custo muito reduzidos de forma a poder automatizar-se muitas actividades. Por exemplo, numa área florestal podem ser colocados estrategicamente muitos sensores de fogo e incluir, para a área toda, apenas um ponto de acesso à rede que capta os dados para uma central. Sensores colocados longe desse ponto de acesso podem usar os restantes sensores para ter conectividade.

(v) alargamento do alcance. Uma rede ad hoc pode aumentar o alcance efectivo de uma dada tecnologia sem fios usando a camada de rede.

2.2. Propriedades

Propriedades básicas das redes ad hoc.

Dados os cenários de utilizador final indicados na secção anterior e as características que definem uma rede ad hoc, algumas propriedades *a priori* podem já ser inferidas. A enumeração destas propriedades será útil ao longo desta dissertação.

(i) nós com recursos que podem ser muito limitados. Sendo o ambiente móvel, os nós terão de ser alimentados com baterias e fisicamente terão de ser pequenos para permitir portabilidade. Isto pode constituir um severo obstáculo à liberdade de concepção dos algoritmos de funcionamento da rede. Como exemplo, a somar às dificuldades intrínsecas de uma rede de comunicações, o algoritmo de encaminhamento pode ter de ser sensível à energia instantânea do equipamento ("power-aware") para ser eficiente de um ponto de vista energético.

(ii) cada nó tem o papel de encaminhador e origem/terminador de tráfego. Para piorar, exige-se que os nós tenham a função adicional de encaminhadores, o que acrescenta significativa complexidade ao 'software' sem benefício directo para o utilizador. Por outro lado, sendo cada nó um encaminhador, desde logo há problemas de segurança como

mascarar a origem ("spoofing")[1] e intersecção de tráfego alheio ("snooping")[2].

(iii) pode existir várias rotas para o mesmo destino. Pode ser considerado um benefício e um malefício. É, contudo, uma característica particular das redes ad hoc, por questões de largura de banda. Multi-caminho nas redes com fios é quase sempre entendido no contexto de tolerância a falhas ou balanceamento de carga e não como forma de obter maior largura de banda. Este ponto será discutido mais tarde.

(iv) desconhecimento da rede. Pode ser um dos problemas mais difíceis de resolver e raramente é mencionado. Um nó que se associe a uma rede ad hoc não tem qualquer conhecimento da rede e não pode presumir qualquer garantia dela. Alguns aspectos determinantes podem ser os seguintes:

- dimensão da rede
- tipo de nós que a compõe: recursos e personalidade (cooperantes ou não)
- serviços (no sentido de rede, tal como ligação à Internet)
- podendo não haver administração, a qualidade da informação que circula pode não ser acreditada – particularmente importante em algoritmos de encaminhamento

Para além disso, o problema da instabilidade da rede é determinante. Havendo entradas/saídas e quebra/formação de rotas a uma taxa arbitrária (do ponto de vista do nó), a rede pode, num curtíssimo período de tempo, alterar as suas características drasticamente (tal como a capacidade). Este aspecto será discutido mais tarde, principalmente na comparação de algoritmos de encaminhamento proactivos e reactivos.

(v) redes des-hierarquizadas. A possibilidade de não existir conceitos ligados a LANs tais como agregação de endereços de rede, domínios de

[1] *spoofing*: genericamente, mascarar a origem, manipulando directamente os pacotes enviados ou reencaminhados alterando-lhes o endereço de origem.
[2] *snooping* (ou eavesdropping): interceptar tráfego que não lhe é dirigido, bloqueando-o ou não, com a intenção de o analisar.

> "colisões", partição em grupos, etc., cria um ambiente extremamente anárquico. Se, no caso de uma LAN, é necessária uma porta de saída ("gateway"), na mesma sub-rede, numa rede ad hoc cada nó pode ser ele próprio uma sub-rede. Continua, no entanto, a haver sub-redes e portas de saída para efeitos de um ARP[1] sem fios: um pacote IP que não seja destino a um nó da mesma sub-rede, é encaminhado para o nó que serve de porta de saída.

Tipologia das redes ad-hoc.

Para lançar as bases do trabalho, admitiram-se 3 teatros de operação atómicos e disjuntos (nenhum substitui na parte ou no todo outro) de redes ad hoc. Esta estrutura é derivada das propriedades acima apontadas. A classificação permite apontar vantagens e desvantagens de cada cenário, no esforço de conter as desvantagens ao particionar os obstáculos – tab. 1.

Como será argumentado ao longo desta dissertação, o teatro em que a rede actua é decisivo para ajustar as suas características – o papel do planeamento pode ser decisivo. Principalmente do ponto de vista de QdS, todos os dados obtidos a priori sobre a rede e os seus objectivos são de considerar para se conseguir reduzir a complexidade do problema. De notar ainda que pode haver redes mistas e o propósito de traçar estas tipologias rígidas serve fundamentalmente para clarificação de opções atómicas.

2.3. tecnologias L1 e L2 de suporte

2.3.1. Tecnologias possíveis.

Uma rede ad hoc só faz sentido com tecnologias sem fios. Várias tecnologias actualmente cumprem esse fim embora cada uma delas tenha, tipicamente, um fim específico. As

[1] *Address Resolution Protocol*, mecanismo de tradução de endereços de rede em endereços de ligação e v.v.

principais tecnologias existentes são o IEEE802.15 ('bluetooth'), o IEEE802.11 (Wi-Fi), o IEEE 802.16/.20 (Wi-Max) e o HiPeRLAN.

designação	exemplo	descrição	características
não-administradas	- espaço aberto - dispositivos muito diferentes (desde PDAs a computadores portáteis) - ponto de acesso comum - rede ad hoc para conectividade de Internet.	São redes que não estão sob alçada de nenhuma organização pelo que o grau de anarquia é elevado. Este cenário corresponde, essencialmente, a um conjunto de nós que entram no alcance uns dos outros e procuraram conectividade formando uma rede ad hoc.	(-) desconhecimento total das características da rede (dispositivos, conectividade com Internet, serviços de suporte a QdS, etc.) (-) pouca ou nenhuma coordenação e organização entre/de os nós (-) cada nó compete pelo máximo de recursos da rede
administradas	um "hotspot" privado	Havendo uma entidade superior, podem impor-se regras e negociar-se a utilização de recursos. Pode haver classes de utilizadores e um dado comportamento pode ser exigido. P.ex., a entidade administrativa pode determinar que não mais de N nós se podem associar à rede, dada a sua capacidade, previamente planeada e definida.	(+) regida pela figura do ISP, há coordenação administrativa e gestão dos recursos por utilizador (-) o utilizador pode continuar sem saber que tipo de rede tem disponível e quais os seus recursos (-) os recursos da rede disponíveis estão ainda muito dependentes do comportamento de cada nó
planeada	soldados	Têm a vantagem de ser uma rede planeada ao pormenor, pelo que cada nó conhece as características da restante rede, os seus limites, cooperando para que a rede não ultrapasse o eventual ponto óptimo de funcionamento	(+) principais características da rede sobre controlo (-) pouca flexibilidade, optimizada para um fim muito particular

tab. 1. *Teatros de operação atómicos de redes ad hoc considerados neste trabalho.*

A tab. 2 compara sucintamente estas tecnologias segundo a sua aplicabilidade em redes ad hoc. Será dado ênfase nesta dissertação ao IEEE802.11 por ser a camada física mais usada e por influenciar decisivamente o projecto da rede.

tecnologia	aplicações	comentários
802.15 -'bluetooth' -UWB	PAN < 10 m < 1Mbps	O principal propósito é eliminar fios em equipamentos de interacção humana (rato, teclado, telemóveis, câmaras, etc.). Para além disso, um dos principais requisitos é permitir baixo custo e pequenas dimensões. O seu carácter genérico, contudo, permite imaginar aplicações para redes ad hoc e até se refere que as redes ad hoc começaram por ser visualizadas com o 'bluetooth' – as "piconets" e as "scatternets". Vários projectos existem para implementar redes ad hocs baseadas em 'bluetooth', e.g. o projecto BEDD [4]. De um ponto de vista de aumento do alcance de um dispositivo 'bluetooth', as redes ad hoc podem ser uma solução possível.
802.16/20 -WiMax	rede de acesso sem fios	O WiMax enquadra-se nas redes de acesso e cabe no grupo de tecnologias como xDSL, cabo, xPON, EFM, etc.
IEEE802.11 -WiFi	LAN sem fios	Torna-se cada vez mais a norma de facto para as redes ad hoc e, genericamente, para redes locais sem fios.
HiPeRLAN (ETSI)	LAN sem fios	Muito semelhante ao IEEE802.11, distinguindo-se por ter um MAC diferente (baseado em TDMA). Seja por razões técnicas ou por questões de mercado, é uma norma sem grande adesão.

tab. 2. *Tecnologias de nível de ligação sem fios mais relevantes actualmente.*

2.3.2. HiPeRLAN/2

O HiPeRLAN (High Performance Radio LAN) é de origem europeia tendo sido normalizado pelo ETSI. Muito embora a camada física seja igual à do IEEE802.11, de origem americana, o MAC é bastante diferente.

A técnica de acesso múltiplo escolhida foi o TDMA/TDD. As ranhuras-temporais e os canais lógicos daí derivados tornam fácil implementar QdS mas exigem um ponto de coordenação e a figura do access point (AP). O HiPeRLAN tem dois modos de funcionamento. O modo centralizado (CM), em tudo análogo ao PCF do IEEE802.11 (a discutir à frente), implica que todo o tráfego passe pelo AP, incluindo a alocação de ranhuras-temporais. O modo directo (DM) prevê que dois nós possam comunicar

directamente, sendo que a função do AP é desempenhada por um nó escolhido arbitrariamente (o CC, controlador central), definindo-se agregados centrados no CC.

Nas comparações com o IEEE802.11, o HiPeRLAN mostra ter maior débito [5] [6], muito por força da existência de um AP dedicado à coordenação e por impor um tamanho fixo de trama. O tamanho fixo da trama está na razão da camada de convergência que o ETSI propôs para servir de adaptação aos protocolos acima, muito ao estilo do ATM. Do ponto de vista da robustez física, o HiPeRLAN tenta controlar as variações das características do canal com algoritmos de controlo de potência e de selecção dinâmica das portadoras.

As implementações de redes ad hoc baseadas em HiPeRLAN implicam sempre manipular a L1 e/ou a L2, construindo-se "rotas" baseadas em canais físicos ou lógicos. É mais correctamente definido como um esquema de encaminhamento. Cada CC tem de fazer o interface com o agregado vizinho à custa da multiplexagem na frequência ou no tempo. Uma solução multi-agregado possível [7] é usar agregados com portadoras diferentes o que resolve desde logo problemas de sincronismo. Em [8], usando um AP, são usadas duas bandas, 5 GHz e 60 GHz, e na segunda banda certos canais lógicos são reservados temporariamente pelo AP para criar a "rota".

A aceitação no mercado é muito fraca – basta fazer uma pesquisa por equipamentos que suportem HiPeRLAN. O mercado está claramente a adoptar o IEEE802.11.

2.3.3. IEEE802.11

Introdução.

Como dito, esta tecnologia está a tornar-se a norma 'de facto' para redes locais sem fios. As variantes com mais aceitação no mercado no mercado, .11b e .11g, parecem girar mais em termos da largura de banda permitida e menos em enrobustecer a tecnologia. Se facilmente se comunica a 54Mbps de taxa de linha, outros aspectos podem ser melhorados que tornem, de forma global, esta taxa de linha mais efectiva do que realmente é.

Consiste em várias adendas, tal como descrito na tab. 3. As relevantes para esta dissertação são o próprio IEEE802.11 e o recente IEEE802.11e. De resto, a interoperação com a família IEEE802 (onde se incluem as famílias 'ethernet') proporciona um conjunto notável de funcionalidades já devidamente normalizadas e divulgadas. Por exemplo, o 802.1 (a definição de ponte – "bridge"), desde logo proporciona interoperabilidade com LANs e o IEEE802.11e usa o 802.1D (classes de tráfego). Talvez devido a este contexto e à (relativa) simplicidade típica das normas IEEE, o IEEE802.11 tornou-se um caso de sucesso, em detrimento, p.ex., do HiPeRLAN. Por outro lado, o esforço de interoperabilidade entre fabricantes, com a aliança Wi-Fi, anulou o principal obstáculo à implantação do "wireless" no mercado da electrónica de consumo.

802.11	norma original (PHY+MAC)	**802.11j**	normalização para o Japão
802.11a	54 Mbps na banda de 5 GHz	**802.11k**	gestão de recursos de rádio
802.11b	11 e 5.5 Mbps na banda de 2.4 GHz	**802.11m**	actualização da família IEEE802.11
802.11c	operação em ponte ("bridge")	**802.11n**	novos débitos (até aos 100 Mbps reais)
802.11d	bandas rádio utilizadas	**802.11p**	ambientes de mobilidade veicular
802.11e	suporte a QdS (discutido abaixo)	**802.11r**	suporte a transições entre APs
802.11F	interligação de APs (IAPP)	**802.11s**	"ESS mesh networking"
802.11g	54 Mbps (OFDM) em 2.4 GHz	**802.11T**	métricas e métodos de teste (WPP)
802.11h	normalização para a Europa	**802.11u**	interoperação com outros tipos de redes
802.11i	segurança e encriptação	**802.11v**	OAM

tab. 3. *Família IEEE802.11 (2005)*

Para além disso, o IEEE802.11 é a tecnologia adoptada na esmagadora maioria das redes ad hoc por duas razões essencialmente. Primeiro, devido à simplicidade do MAC (não há qualquer tipo de planeamento prévio de recursos, basta assumir que os nós comunicam pelo mesmo canal); em segundo, devido à percepção generalizada de que será o IEEE802.11 que irá estabelecer-se como tecnologia sem fios *de facto* para redes ad hoc. A excepção que se perspectiva será eventualmente o 'bluetooth' (IEEE802.15) mas em aplicações diferentes.

O algoritmo de acesso ao meio (CSMA/CA) baseia-se em ouvir-antes-de-falar ("listen-before-talk") o que só por si permite uma distribuição eficaz do acesso ao meio. Aliás, a utilização de um AP actualmente, numa aplicação simples de LAN sem fios, só se justifica para simplificar o acesso à rede local, aumentando a cobertura L1 (orçamento da ligação – "link budget"). De um ponto de vista de QdS, o IEEE802.11 tem várias lacunas e a capacidade de resposta do IEEE parece ter falhado nalgum ponto a meio da conceito – o IEEE802.11e, que deveria fornecer mecanismos de prioritização na partilha do meio, tem resultados modestos, quando comparados com outros mecanismos [99].

Sendo de utilização genérica e operando na banda ISM (com extensões a 5 GHz e outras adaptadas ao local de uso – Europa, Japão e Américas), o IEEE802.11 prevê utilização em qualquer ambiente. A tecnologia é suficientemente robusta em relação a interferências principalmente devido ao CDMA do PHY. O próprio CSMA/CA fornece também alguma robustez extra já que só há início de transmissão quando o canal está limpo (RTS/CTS).

Funcionamento geral.

A Fig. 3 representa uma trama de dados e a Fig. 4 um diagrama temporal genérico do acesso ao meio do 802.11[1].

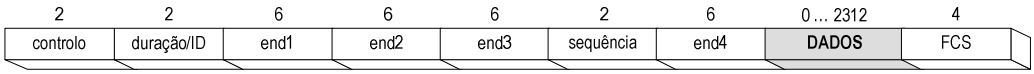


Fig. 3. *Trama IEEE802.11.*

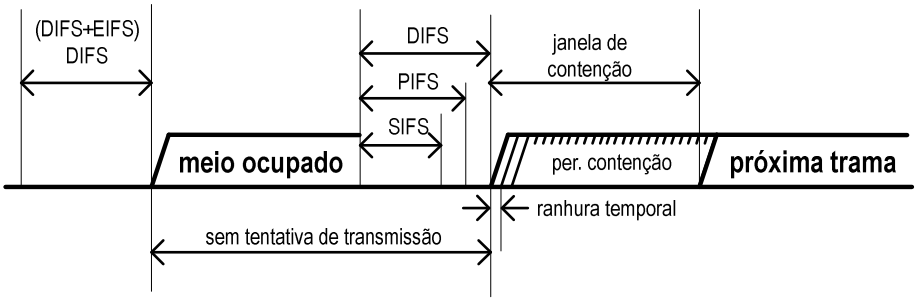


Fig. 4. *Acesso ao meio no IEEE802.11 (modo DCF).*

[1] Esta secção é baseada no capítulo 9 ("MAC sublayer functional description") do IEEE802.11 (versão de 1999). Figuras, tabelas e tabelas associadas foram de lá retiradas, excepto menção em contrário. A fig. 4 foi ligeiramente alterada para incluir o EIFS.

O IEEE802.11 define dois modos de operação: DCF ("distributed coordination function") e PCF ("point coordination function"), embora o DCF tenha maior importância. Em primeiro lugar, o PCF é uma extensão do DCF. Em segundo lugar, a norma obriga que o equipamento que siga a norma IEEE802.11 implemente DCF, deixando o PCF como opção.

O PCF é o modo usado em conjugação com um ponto de acesso (AP ou PC[1]) e adiciona ao DCF mecanismos de interrogação ("polling") para efeitos de atribuição de tempo de antena às estações móveis (STA) – Fig. 5. Obviamente, um AP tem de implementar DCF+PCF enquanto uma STA só necessita de implementar DCF[2]. O mecanismo de interrogação usado privilegia cenários que necessitam de maior determinismo ao eliminar a contenção de acesso: o AP vai interrogando cada STA e alocando ranhuras-temporais para que cada uma possa transmitir sem recorrer a períodos de contenção. Ao ser atribuído direito de transmissão a uma dada estação, o modo DCF é usado de forma normal entre a STA e o PC que actua como intermediário e memória de armazenamento ("buffer") de tramas entre a STA origem e a STA destino. Para manter o sincronismo entre as estações existe uma espécie de super-trama que o AP transmite periodicamente, onde efectua a interrogação às STAs com uma trama de publicitação ("beacon frame").

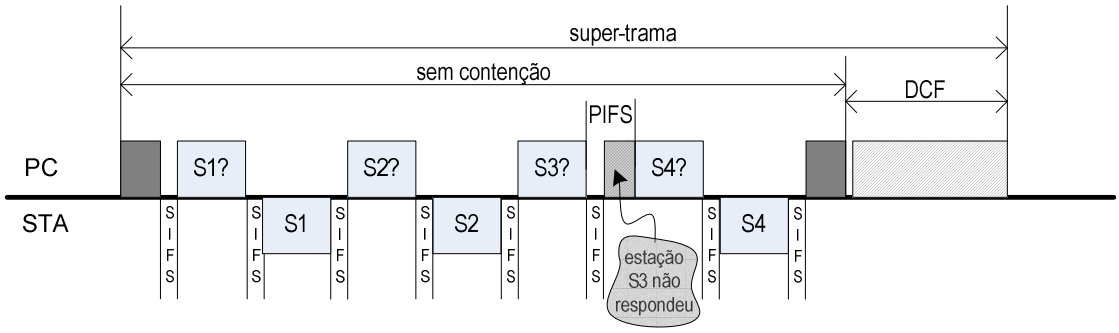


Fig. 5. *Diagrama temporal do modo PCF. O Ponto de Acesso transmite um sinal piloto de sincronização, vai interrogando cada estação e enviando as tramas armazenadas para cada estação seguindo-se um período em que a estação interrogada tem oportunidade para transmitir pacotes. No fim do período sem contenção, surge um período em que as estações entram em modo DCF.*

[1] AP: Access Point; PC: Point Coordinator
[2] Daí até a designação de "CF-pollable" às STA que implementam PCF (CF de "contention free").

Os parâmetros fundamentais do IEEE802.11 relacionam-se com a temporização do algoritmo de acesso ao meio e são essencialmente definidos pelas características do PHY. Os mais importantes encontram-se na tab. 4 (cf. Fig. 4).

parâmetro	nome	descrição
IFS (DIFS, PIFS, SIFS, EIFS)	espaçamento inter-trama ("inter-frame spacing")	Intervalos de tempo entre a estação determinar que o meio está livre e o início da transmissão do primeiro pacote ou fragmento na fila de espera. Definidos no texto.
BP	período de contenção ("backoff period")	Ao detectar que o meio está livre, cada STA espera obrigatoriamente um IFS (adequado ao contexto da próxima trama a transmitir). O período de contenção é o número de ranhuras-temporais (1 r.t. = 9 us no IEEE802.11a) que decorrem entre o instante após o IFS e o instante em que inicia a transmissão. É gerado pelo produto de um número aleatório (distribuição uniforme), definido no instante da primeira transmissão após a última transmissão bem sucedida. Após uma transmissão com colisão ou sem sucesso (meio ocupado), este número é recalculado.
CW	janela de contenção ("contention window")	Largura em ranhuras-temporais do BP. Este número vai crescendo com 2n com n o número de transmissões sem sucesso até atingir um máximo fixo. É parâmetro de entrada para BP.
NAV	vector de alocação do meio ("network allocation vector")	Discutido à frente (problema do nó exposto/nó escondido)

tab. 4. *Alguns parâmetros do IEEE802.11.*

O IFS usado depende do estado actual e da versão do IEEE802.11. A um IFS pequeno corresponde a uma prioridade mais alta da estação (contenção menor, menor tempo de espera antes de transmitir, maior probabilidade de transmitir antes de outra estação tentar).

DIFS é usado em transmissões iniciais de tramas de dados. Se a transmissão for no modo 'fragment-burst', é usado o SIFS entre cada fragmento do mesmo pacote. SIFS é usado em acções que necessitem de rápida resposta (`RTS/CTS/ACK`) e resposta a interrogação ("polling") pelo PC já que no modo PCF o meio é gerido pelo PC, não havendo necessidade de mecanismos de contenção elaborados. PIFS é o equivalente ao DIFS mas no modo PCF. EIFS é usado quando, a meio de uma transmissão, há detecção de que a trama não foi correctamente recebida pelo destino (a própria STA detecta que o FCS da trama transmitida não corresponde ao FCS lido do ar), havendo um período de espera

para garantir que a estação destino detectou o erro. A um EIFS segue-se o algoritmo de acesso normal (livre→DIFS→Backoff Period).

Os tempos entre tramas, portanto, ordenam-se por SIFS < PIFS < DIFS < EIFS. Aliás, a manipulação em tempo real do IFS é um dos mecanismos possíveis para suportar prioridades relativas. É esta a base de operação do IEEE802.11e na tentativa de suportar aplicações que necessitem de QdS.

Problema do nó exposto/nó escondido.

O 802.3 define um comprimento do cabo máximo para controlar a difusão das tramas e, em especial, a detecção das colisões. Sendo o IEEE802.11 sem fios, com transmissão em difusão[1], e havendo sempre duas zonas de interferência (recepção e detecção), foi necessário criar mecanismos de confirmação ("acknowledgement") para que a estação transmissora determine fiavelmente se a transmissão teve sucesso. Na Fig. 6 está representado o problema. As duas situações (numa há impossibilidade de comunicação e noutra há sub-eficiência) estão previstas e resolvidas através do MAC usando uma transacção pedido-de-transmissão ("request-to-send" – `RTS`) → autorização-de-transmissão ("clear-to-send" – `CTS`) → confirmação-de-recepção ("acknowledge" – `ACK`). Esta é a razão pela qual a determinação de meio implica duas condições satisfeitas:

- o PHY detecta meio livre *e*
- contador NAV nulo (Fig. 7). O NAV ("Network Allocation Vector") é um contador que tenta prever quando irá terminar a comunicação entre os nós após a transacção `RTS/CTS`. O campo 'Duration/ID' (Fig. 3) do pacote `RTS` indica a estimativa do transmissor do tempo que vai ocupar o meio. A estação receptora transmite um pacote CTS que inclui a duração anunciada pela estação transmissora, para informação de eventuais nós que estejam no alcance do receptor *mas não* no alcance do transmissor. Ou seja, qualquer nó (incluindo os escondidos) espera pelo menos este tempo, ainda que não detecte o meio ocupado.

[1] tirando aplicações específicas, a generalidade das aplicações usa antenas omnidireccionais

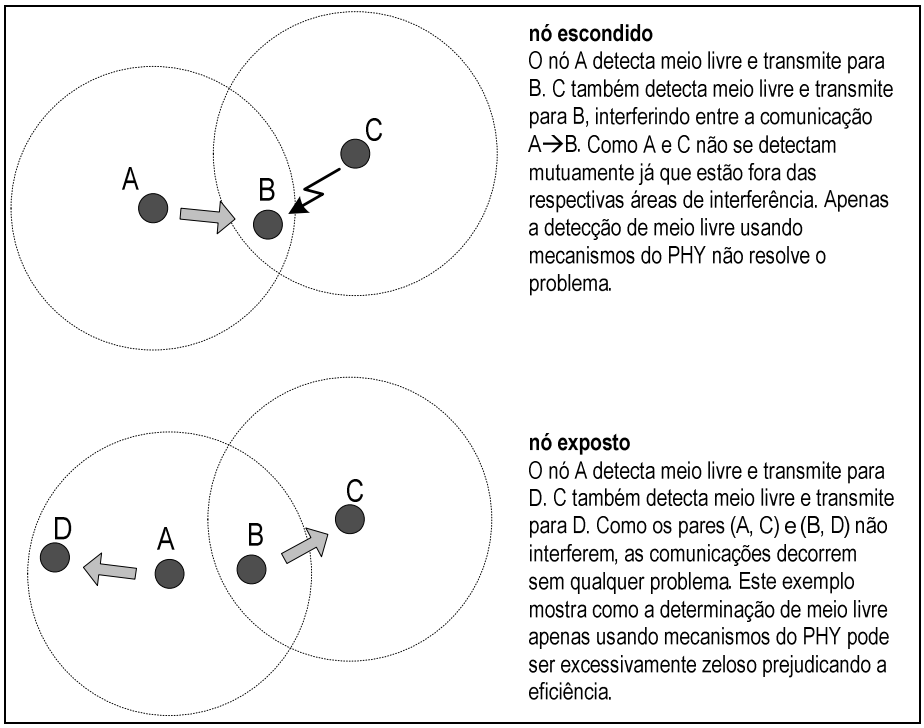


Fig. 6. *Problema do nó exposto/ nó escondido*

Na literatura encontra-se por vezes a designação deste método de determinação de meio livre como Detecção Virtual de Portadora ("Virtual Carrier Sense").

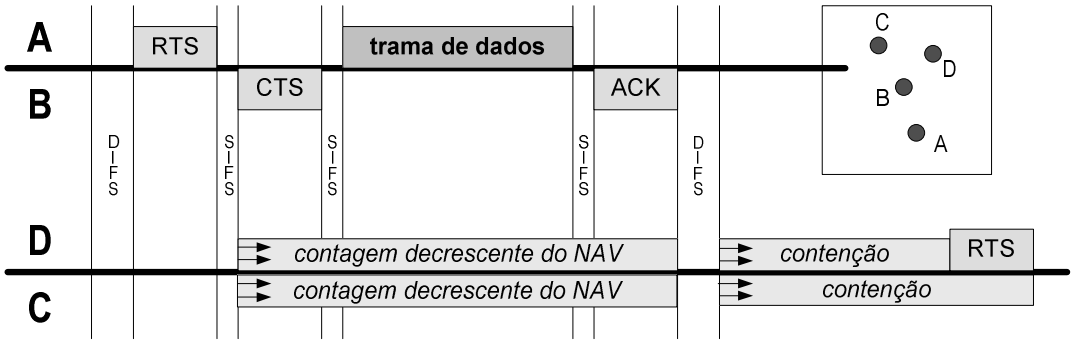


Fig. 7. *Diagrama temporal mostrando a detecção virtual de portadora. Neste exemplo, todos os nós tentam transmitir uma trama para B, notando que eles interferem entre si directa ou indirectamente. B ouve todos, C e D ouvem B mas não A e C ouve D.*

Suporte a QdS.

O IEEE802.11 original pode ser considerado, do ponto de vista do suporte a QdS, como orientado à estação e não orientado ao tráfego: não há nenhum mecanismo que permita

tratamento diferenciado de pacotes provenientes da mesma origem; quando muito, cada estação receptora só pode diferenciar pacotes baseando-se na estação de origem.

Concretamente, em modo DCF, não há qualquer forma de estabelecer prioridades entre tramas porque cada estação apenas tem uma única fila de espera à saída. No modo PCF, tecnicamente seria possível usar o PC em conjugação com um algoritmo mais sofisticado de controlo de acesso do que o 'round-robin' (como o WFQ[1]), e suportar diferenciação de tráfego. Contudo, seria sempre um esquema de diferenciação de estações e nunca de diferenciação de tráfego.

O IEEE802.11e tenta fornecer mecanismos mais elaborados para a diferenciação de tráfego definindo um novo MAC (EDCF, retrocompatível com o DCF), um modo especial HCF ("hybrid coordination function") e marcando as tramas com a indicação do tipo de tráfego.

No EDCF, essencialmente, a prioridade de determinada trama é o resultado de duas técnicas. Primeiro, criaram-se tipos de tráfego (vindos do IEEE 802.1D) que podem ser usados para criar diferentes filas em cada STA. As classes de tráfego são variações do 'best-effort' (BE), tempo-real (RT) e sondagem ("probing"). Como qualquer classificação de tráfego, é livre de ser usado da forma que a implementação entender embora se sugiram aplicações específicas de vídeo e áudio. Em segundo, criaram-se mecanismos de prioritização de acesso ao meio.

Assim, criaram-se 3 bits de filas de espera distintas e mecanismos de prioritização entre elas. A prioritização (4 níveis) é feito à custa do período de contenção ("backoff-period" e factor de persistência) e da definição da AIFS (Arbitrary IFS). O período de contenção e o AIFS são função da fila TC ('Traffic Classification'), diferenciando desta forma o tráfego. Um escalonador à saída das filas determina qual deve ser servida em cada instante – formalmente, atribui a oportunidade de transmissão, TXOP, à fila com mais prioridade. O AIFS consiste na flexibilização do DIFS, permitindo tempos maiores para que uma estação com tráfego prioritário tenha acesso ao meio antes de outra com tráfego menos

[1] WFQ: Weighted Fair Queueing; é um algoritmo de escalonamento que se aproxima bastante do teórico e desejável GPS (Generalized Processor Sharing)

prioritário. Na prática, a solução tira prioridade às estações com tráfego não prioritário, impondo-lhes um tempo entre tramas superior ao do IEEE802.11 original – para tráfego prioritário AIFS=DIFS, aumentando o tempo de IFS consoante o pacote tenha menos prioridade.

O HCF explora o determinismo do mecanismo de interrogação do PCF e a alocação controlada do meio via PC. É um misto de DCF e PCF mas com manipulação do IFS dando ao HC (controlador híbrido – "hybrid controller") prioridade absoluta sobre o meio havendo, estatisticamente, garantias superiores ao EDCF. As etiquetas de tipo de tráfego (AC – Categoria de Acesso – "Access category") são de 4 bits, metade para o EDCF e metade para o HCF, permitindo outras 8 filas para o modo HCF.

De um ponto de vista de redes ad hoc, o HCF não é um modo muito útil na medida em que é pensado para a utilização em LANs. Já há vários estudos sobre a aplicação de VoIP sobre o IEEE802.11 com pontos de acesso e torna-se claro que o problema não é propriamente a largura de banda disponível – 11 Mbps do IEEE802.11b dariam para cerca de 1000 "chamadas" quando na realidade não se consegue mais de cerca de 20 "chamadas" com um atraso abaixo dos 250 ms e perda de pacotes aceitável para o codec utilizado [9]. Espera-se que o HCF tenha melhor desempenho especialmente se conjugado com um protocolo de sinalização, um cenário previsto durante o desenvolvimento do modo de operação.

Capacidade.

A capacidade de uma rede baseada no IEEE802.11 é difícil de calcular a priori por depender muito do contexto de utilização. No modo PCF, num cenário de LAN, é mais simples e determinístico, mas em modo DCF e numa aplicação multi-salto, o cenário pode ser o factor mais importante. Pode contudo afirmar-se que o limite, geralmente, não está na largura de banda disponível mas nas restrições que o acesso ao meio impõe.

Pode por isso dizer-se que o parâmetro essencial de desempenho do IEEE802.11 é o atraso: controla directamente o débito-entre-extremos do TCP (janela que o TCP mantém em tempo-real) e a perda de pacotes do UDP (tamanho das filas de cada nó), para referir os protocolos de transporte mais utilizados. O atraso, por sua vez, a partir de uma certa

utilização do meio, quase não depende da taxa de linha, mas do tempo de acesso ao meio. Notar que, a 11 Mbps, uma trama de 1 kB demora 74 µs a ser transmitida. O valor global para o atraso, entre *apenas dois nós* (incluindo STA-PC, no modo PCF), de cerca de 2 ms é o melhor que se consegue[1].

O tempo de acesso ao meio está intimamente ligado à competição dos nós pela oportunidade de transmissão e, daí, à densidade espacial dos nós e ao perfil de tráfego que cada nó gera[2]. Assim, tem mais impacto o número de pacotes que se tenta transmitir (ou seja, a taxa de tentativas de acesso ao meio) do que a quantidade de dados que se transmite. A Fig. 8 ilustra esse efeito: para uma redução do tamanho do pacote de 1500/64=23, a largura de banda apenas caiu um factor de aproximadamente 4. Este raciocínio, juntamente com as condições em que ele se verifica (a Fig. 8 *apenas* pretende ilustrar o efeito porque o factor de 4 referido está abaixo do real para certos cenários), será discutido recorrentemente ao longo desta dissertação.

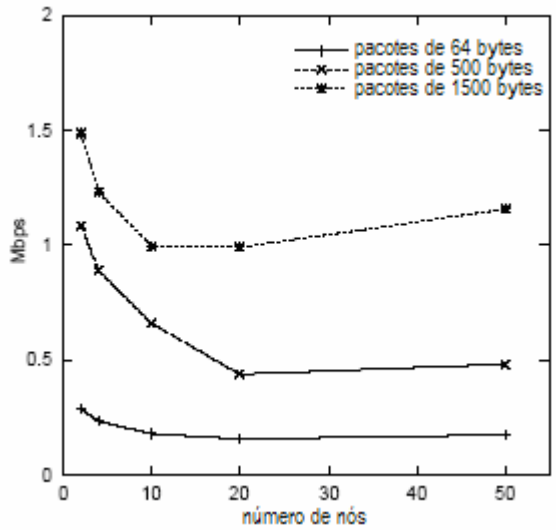


Fig. 8. *Débito-entre-extremos médio numa rede ad hoc. Todos os nós estão dentro do alcance de todos. As três curvas correspondem a pacotes de tamanho fixo mas diferente. [10]*

De um ponto de vista de redes ad hoc, o débito-entre-extremos efectivo ainda é mais baixo na medida em que cada nó encaminha tráfego não próprio, portanto, cada fluxo

[1] Ao longo da dissertação aparecerão numerosos exemplos deste tipo de valores.
[2] Falta referir o efeito de interferências de sinais rádio externos. Sendo um parâmetro associado ao PHY e de acesso interdito ao projectista (não se pode controlar o ambiente de operação), optou-se por não o incluir.

contribui com $O(1/m)$, para uma rota de m saltos. Com mais rigor [10] [11], e ignorando por ora aspectos de mobilidade, podemos supor que a capacidade inicial de cada nó é $O(n)$, para n nós. Se cada fluxo usar rotas de m saltos, o débito-entre-extremos é reduzido em $O(n/m)$. Relembrando que uma rede ad hoc faz sentido enquanto forma de extender geograficamente a conectividade, é de admitir um círculo de diâmetro $\sqrt{n}$ para o teatro de operação (nós distribuídos uniformemente pelo espaço e espaçamentos maximizados): $O(n/\sqrt{n})$. Ou seja, cada nó participante na rede tem um débito-entre-extremos disponível que segue $O(1/\sqrt{n})$. Finalmente, é necessário notar que cada nó pode transmitir em paralelo, desde que não no mesmo domínio de colisões, o que aumenta a capacidade. Contudo, tem de dividir essa capacidade com os seus vizinhos directos e os vizinhos da zona de detecção. Como, estatisticamente, nesta simplificação, a densidade da rede é fixa (não se prevêem cenários de grandes e rápidas entradas e saídas de nós ou frequentes particionamentos e religações da rede), este factor adicional de capacidade é fixo, não dependendo de n. Portanto, o comportamento da capacidade livre por nó, em função do número de nós é, grosseiramente,

$$\Theta = O(1/\sqrt{n}) \qquad \textbf{(eq. 1)}$$

A Fig. 9 mostra que este resultado é bastante válido. Não se incluem as características da simulação porque só se pretende ilustrar a forma da curva. A mobilidade, salvo aspectos muitos específicos de utilização (como à frente se analisa), baixa ainda mais a capacidade por, pelo menos, dois efeitos: (i) custo ("overhead") de encaminhamento; (ii) latência de descoberta de rotas.

O facto de a capacidade *disponível* por nó diminuir com o tamanho da rede é efectivamente preocupante e faz questionar o próprio conceito: uma rede ad hoc pequena (<5 nós) pode ter pouca utilidade e uma rede adhoc muito grande (>1000 nós) não é eficiente e pode mesmo ser considerado que a própria rede consome todo o tráfego que nela é gerado. Parece existir, portanto, uma zona de interesse das redes ad hoc muito bem definida. Esta zona de validade será discutida à frente.

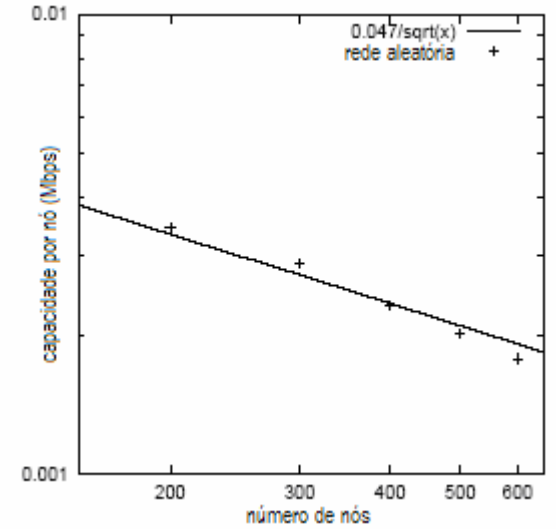


Fig. 9. *Capacidade por nó numa rede ad hoc (simulação) [10].*

Se por um lado a (eq. 1) pressupõe condições pessimistas (as rotas atravessam a rede toda, o tráfego tem um perfil indefinido e todos os nós geram tráfego indiscriminadamente tentando capturar o máximo de recursos), por outro lado, como qualquer modelo simples, implica uma utilização da rede que pode não corresponder à realidade. É neste ponto que os cenários apresentados nas secções anteriores ganham extrema utilidade. Por exemplo, se a rede ad hoc for utilizada como rede de acesso, em que se espera que todos os nós usem um ponto fixo de acesso à WAN (ponto de encaixe – "attachment"), este modelo está mais perto da realidade. Se se admitir que muito tráfego é local e entre nós próximos, o modelo é exageradamente pessimista.

Por um lado, é necessário definir grau de mobilidade de uma rede ad hoc. A mobilidade dos nós é melhor definida em relação à duração dos fluxos de tráfego entre destino e origem. Por essa razão, se a mesma rota, fornecida pelo encaminhamento, se mantiver válida por um tempo T e os fluxos durarem $\Delta t << T$, a rede ad hoc pode considerar-se estática. Se $\Delta t >> T$ a rede é extremamente dinâmica e a (eq. 1) precisa de o levar em conta.

Aliás, [12] argumenta que a capacidade de uma rede ad hoc aumenta até quase ao limite do permitido pela L2 bastando para tal uma espécie de delegação do encaminhamento na L2: a probabilidade de, num dado instante futuro, o par destino-origem estar a 1 salto de distância aumenta com a mobilidade dos nós (depende, naturalmente, do modelo de mobilidade dos nós) e é questão de distribuir os pacotes por vários nós da rede e de cada nó ter memória local muito grande para poder guardar os pacotes até serem entregues, aproximando-se do conceito de redes tolerantes ao atraso ("Delay Tolerant Networks"). De um ponto de vista de capacidade, é um esquema muito válido, mas do ponto de vista da generalidade da abordagem, são necessárias condições muito particulares para que o modelo seja útil (muitos nós e grande densidade, memória suficiente em cada nó, aplicações com atrasos muito permissivos, etc.). Pode, todavia, considerar-se que mobilidade também ajuda a entregar pacotes. Designar-se-á este mecanismo como *entrega de pacotes via mobilidade*, no contexto desta dissertação.

Por outro lado, a inteligência distribuída na rede, na figura dos protocolos das camadas superiores, incluindo o encaminhamento, pode explorar a real capacidade da rede em função da utilização que se quer dela. *É exactamente neste ponto que este trabalho de mestrado começa.* Em última análise, o objectivo deste trabalho é contribuir para que a rede seja optimizada ao nível da capacidade, notando que optimizar pode não ser maximizar unicamente a capacidade.

Interacções com camadas superiores.

Entendem-se por "camadas superiores" o protocolo IP e os protocolos de transporte (TCP, UDP, RTP, etc.). Vários estudos [14] [15] [16] mostram que a maior parte das aplicações comuns funcionam sub-optimamente no meio IEEE802.11 multi-salto, em particular as que usam o TCP como protocolo de transporte. Quase todos os problemas mostram que ou o MAC IEEE802.11 causa problemas não previstos aquando do projecto do TCP ou o próprio TCP está demasiadamente optimizado para redes com fios. A Fig. 10 mostra uma sessão de tráfego TCP simulada (nós espaçados de 200m para alcances IEEE802.11 de 250m e amostragem de débito-entre-extremos instantâneo para intervalos de 1s) para 120s [16].

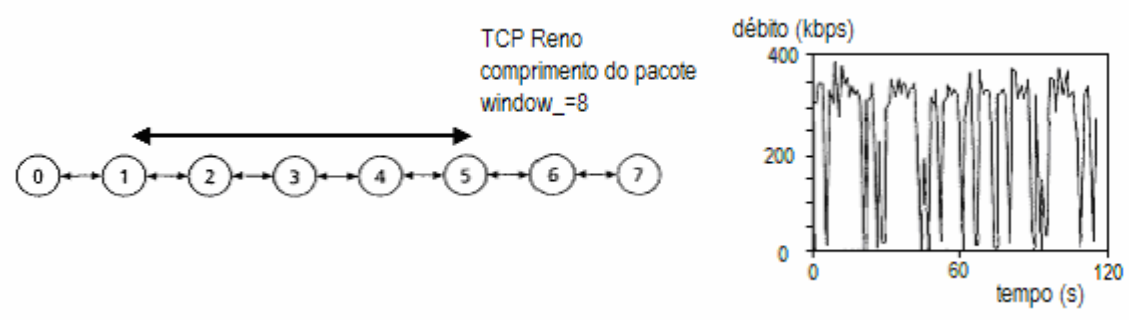


Fig. 10. *Débito-entre-extremos para uma sessão TCP entre os nós (1) e (5). [16]*

Não há razão nenhuma aparente, do ponto de vista da camada de transporte, para os mínimos do gráfico – havendo mesmo 5 instantes em que o débito-entre-extremos é nulo. O problema está no mecanismo que determina que há quebra de ligação, ao nível do MAC. Ao contrário da 'ethernet' cablada, não é simples determinar se um vizinho anterior ainda está dentro do alcance. O IEEE802.11 define falha quando à 7ª transmissão de `RTS` não há o devido `CTS` do nó vizinho. Neste caso, onde foi usado o DSR como protocolo de encaminhamento, houve frequentes quebras de ligação que geraram falhas de rotas tendo sido iniciadas novas descobertas. Por cada quebra de ligação, o TCP inicia o procedimento de início-lento ("slow-start").

O problema aqui é o mesmo que foi referido anteriormente: não é exactamente a falta de largura de banda que determina o desempenho sub-óptimo do IEEE802.11 mas a taxa de pacotes (ou seja, a taxa de acesso ao meio) tentada. A prova disso é que a afinação da janela do TCP é fundamental para minimizar este problema até ao limite de uma janela de 1 pacote onde o TCP funciona muito bem [16] à custa de, na prática, anular uma parte significativa da inteligência do TCP levando a baixos débitos. Uma solução possível para este caso consistiria em dotar o IEEE802.11 de mecanismos mais sóbrios de protecção de quebras de ligação, tais como os existentes no MACAW [15]. Este MAC implementa diferenças ao nível da transacção `RTS-CTS-ACK`, no tratamento dado ao backoff e nas filas básicas existentes em cada nó (uma por vizinho, p.ex.). Outras técnicas, como a proposta em [17], que introduz variância do atraso ("jitter") propositado nas sessões TCP, atenuam os mínimos da Fig. 10 e tornam o débito-entre-extremos mais suave ao longo do tempo.

Um problema adicional reconhecido ao IEEE802.11 é o de potenciar a injustiça ("unfairness") do TCP [16]. Há dois aspectos neste problema. Em primeiro lugar, o efeito de captura do TCP: uma sessão pode ocupar todo o canal. Em segundo lugar, nós directamente ligados com uma sessão TCP estabelecida entre eles inibem outras sessões multi-salto, incluindo as originadas nos próprios nós. Aliás, este efeito de uma sessão dominar o canal extremo-a-extremo já foi observado até em WANs, em que há muitas tecnologias diferentes pelo meio.

Para redes ad hoc, o melhor parâmetro de regulação do TCP poderá não ser o RTT mas o número de pacotes ao longo da rota. Seria interessante adaptar ligeiramente o TCP para considerar não o RTT mas o número de pacotes na fila do nó mais carregado ao longo da rota (as modificações não seriam muitas e profundas). De forma geral, existem várias tentativas para adaptar o TCP ao meio ad hoc (e.g., [19]) mas que, por saírem fora do âmbito desta dissertação, não serão discutidas.

2.4. encaminhamento em redes ad hoc

2.4.1. protocolos de encaminhamento

Introdução.

Havendo ligação L2, é necessário criar uma rota L3 entre a origem e o destino. Define-se *salto* como um vizinho directo de um nó, do ponto de vista da camada de rede. Uma *ligação* é o canal que liga o nó a um vizinho directo, do ponto de vista da camada de dados. Neste contexto é o mesmo mas outros haverá em que isso não acontece: um vizinho para a camada de rede pode estar a muitas "ligações" de distância, bastando para tal que a camada de dados emule a ligação extremo-a-extremo.

Existem muitos protocolos de encaminhamento, alguns para cenários muito específicos (e.g., redes de sensores), outros usando técnicas emprestadas de outros domínios (e.g., localização por GPS), outros para redes muito grandes, outros para redes muito pequenas,

etc. Virtualmente há um protocolo de encaminhamento para cada caso. Uma lista encontra-se em [20]. Neste trabalho analisa-se brevemente os seguintes protocolos:

- AODV, muito popular;
- DSR, igualmente popular;
- OLSR, como exemplo de um protocolo proactivo;

Propriedades de um protocolo de encaminhamento.

Um protocolo de encaminhamento tem de possuir à partida certos requisitos básicos [1]:

- *auto-iniciado* ("self-starting"). O protocolo de encaminhamento tem de ser autónomo e iniciar descobertas de rotas logo que for necessário.
- *capacidade de manutenção*. Quando uma rota deixar de ser válida (uma ligação partida, p.ex.), a tabela de encaminhamento tem de ser actualizada num tempo adequado.
- *livre de ciclos*. Tem de conseguir que as rotas não tenham ciclos ou, se tiverem, forma de minimizar as suas consequências (como o TTL).
- *esforço pedido à rede ("overhead")*. O tráfego das operações de encaminhamento deve ser em quantidade desprezável quando comparado com o tráfego útil. *Notar que o tráfego útil deve ser dimensionado no ponto da rede perto da capacidade máxima.* Assim consegue-se um parâmetro que só depende da rede em si, e não da utilização que se der dela. Este é talvez um ponto onde alguns trabalhos falham ao quererem comparar esforços absolutos entre protocolos de encaminhamento, ponderando o peso do controlo em cada pacote ("overhead") em função do tráfego na rede, esquecendo que esse peso é quase independente do tráfego na rede.
- *convergência rápida*. Especialmente em redes muito dinâmicas, este parâmetro é fundamental e intervem directamente, p.ex., na taxa de perda de pacotes. Uma medida possível para este parâmetro liga-se ao tamanho das filas de cada nó (e ao tráfego gerado). Enquanto não há rota, as filas de saída dos nós com pacotes por enviar vão enchendo. Em termos globais, o número

médio de pacotes no sistema pode estar limitado pela velocidade de convergência do protocolo de encaminhamento.

No âmbito específico das redes ad hoc, um protocolo de encaminhamento tem, ou pode ter, requisitos extra:

- *simples* (quando traduzido em "software" do nó: pequeno e rápido). Notar que, em redes com fios, os encaminhadores são tipicamente equipamentos dedicados e não de utilizador. Contudo, numa rede ad hoc, cada nó é também um encaminhador. Se se considerar que um nó pode ser um mero telemóvel, este requisito pode ser quase proibitivo.
- *escalável*. Como indicado na secção "capacidade", uma rede ad hoc tem um limite intrínseco de funcionamento e a respectiva métrica pode ser o tamanho das rotas em saltos. Contudo, na faixa de operação esperada, o protocolo de encaminhamento deve ser tão eficiente para redes pequenas (e.g. 1 salto), como para redes grandes (e.g. 5 saltos). Para redes muito grandes (>10 saltos?), o protocolo de encaminhamento deve ser adequado a esse facto.
- *execução distribuída*. Pela natureza distribuída das redes ad hoc, não pode haver nenhum nó com papel excessivamente centralizador.

Para além destes requisitos, são também de interesse os seguintes:

- *selecção da rota na origem*. As redes ad hoc podem ter uma vantagem em relação às redes de tecnologia com fios que consiste em haver múltiplas rotas entre o mesmo par S-D (origem-destino). Por isso, para beneficiar directa e adaptativamente desta vantagem, a origem deve poder seleccionar o caminho de (no limite) cada pacote, sem envolver a rede demasiadamente. Esta técnica, encaminhamento-na-origem, é implementada facilmente se o protocolo de rede permitir a indicação dos saltos sucessivos. O IPv6, p.ex., permite-o de raiz mas, obviamente, não tem mecanismos de, por si, descobrir rotas.

- *suportar encaminhamento de QdS*. Encaminhamento de QdS é escolher rotas, entre as várias possíveis, que consigam transportar um fluxo com determinados requisitos (largura-de-banda, atraso, etc.). Discutido adiante.
- *capacidade de multi-rota*. No seguimento do raciocínio anterior, multi-rota ("multipath") é uma técnica que deve ser aproveitada quando se justificar. A técnica é discutida mais adiante.
- *suporte a difusão de grupo* ("multicast"). O protocolo deve suportar ou permitir difusão de grupo. A difusão de grupo é, seguramente, uma área extremamente promissora quer em termos do que permite ao nível da actividade humana (conteúdos de grupo) quer ao nível técnico (eficiência na entrega de pacotes para muitos destinatários em simultâneo).
- *medição remota*. O protocolo de encaminhamento deve permitir medição remota e/ou mecanismos de interrogação para as propriedades de determinada rota. Sondagem ("probing") é um exemplo. Contudo, não deve ser uma função do encaminhamento implementar estas técnicas – apenas permitir.
- *reagir a quebras de ligações de forma localizada* [21]. Ou seja, deve ser possível apenas envolver os nós mais próximos para solucionar, do ponto de vista de encaminhamento, uma quebra de ligação e não envolver a rede toda ou toda a rota S-D. Este requisito pode conflituar claramente com o primeiro ao alterar uma rota sem informar a origem que pode pretender que o seu tráfego tenha uma rota precisa.
- *aceitar sugestões de rotas*. Um protocolo deve poder aceitar rotas indicadas por terceiros. P.ex., um ISP pode indicar os saltos da rota a um certo nó e a rede respeitar esta decisão, contornando o algoritmo de encaminhamento.

Finalmente, como nota à parte, é de referir que o tráfego de controlo associado aos algoritmos de encaminhamento corre, tipicamente, sobre UDP pelo que a perda de pacotes sem qualquer notificação é uma realidade que se deve tomar em conta.

2.4.2. Protocolos por-pedido.

Conceito.

Os protocolos tradicionais das redes com fios baseiam-se no conhecimento mais ou menos detalhado da topologia da rede e no custo que cada porto de encaminhamento (neste sentido, ligação) apresenta. Enquanto que o custo é um conceito que pode ser aplicado às redes ad hoc, o conhecimento da topologia é muito difícil de obter. E quanto mais dinâmica for a rede, mais complicado se torna. Neste sentido, uma nova abordagem às tarefas de obtenção de rotas extremo-a-extremo foi proposta. A essência desta nova abordagem é a seguinte: cada nó não tem necessidade absoluta de conhecimento das rotas para todos os nós na rede mas apenas daqueles com os quais mantém sessões de tráfego. Adicionalmente, e como corolário, cada nó pode apenas manter para uma determinada rota o vizinho para onde encaminhar o pacote, não conhecendo, portanto, a rota toda.

Tipicamente, é usado inundação ("flooding") para fazer chegar um pacote pré-formatado de pedido de rota (geralmente designado por `RREQ`) a todos os nós. O nó destino, ao reconhecer-se no pedido, responde com um pacote pré-formatado (geralmente designado por `RREP`, "route reply"). Para tornar os protocolos mais robustos ou eficientes e para diminuir o esforço causado pela descoberta de rotas, várias técnicas foram desenvolvidas. Por exemplo,

- ouvir pacotes de terceiros para aprender rotas
- responder a pedidos de `RREQ` mesmo sem ser o nó destino, bastando que conheça uma rota adequada
- transmissão pela rota inversa (a rota utilizada mas com a ordem dos saltos invertida) da indicação de ligação quebrada nalgum ponto da rota, ao nó origem (`RERR`)
- salvamento ("salvaging") de um pacote por parte de um salto que detectou o próximo salto quebrado ao usar uma rota alternativa que aprendeu por si
- descoberta de vizinhos ao nível da camada de rede
- etc.

Uma possível forma de discutir protocolos de encaminhamento em redes ad hoc consiste em separar o mecanismo central de obtenção e manutenção de rotas dos mecanismos de compensação das ineficiências do mecanismo central (optimização e/ou correcção). Genericamente, seguir-se-á esta abordagem. A compensação de ineficiências é consequência de problemas tais como:

- controlo da inundação para obtenção de rotas, um problema que pode ser complexo [22] [23];
- rotas cíclicas;
- propagação de ligações quebradas (mobilidade e/ou desactivação de nós) e eliminação de rotas inválidas da tabela de encaminhamento;
- tratamento dado a múltiplas respostas `RREP`.

AODV.

O AODV ("Ad hoc On-demand Distance Vector") [24] usa uma tabela de encaminhamento em cada nó com o par (destino; próximo salto). Cada pacote que as camadas superiores gerarem é encapsulado em pacotes de camadas inferiores e encaminhados para o vizinho a que corresponder a entrada que o pacote tiver como destino. Se não houver nenhuma entrada para o destino pretendido é iniciado um processo de descoberta de rota: o nó faz difusão de um pacote especial, `RREQ`, que os vizinhos repetem, até que eventualmente o `RREQ` chega ao nó destino ou a um nó que já conheça uma rota. Em ambas as situações, um pacote de resposta é transmitido pela rota inversa até ao nó inquiridor. Esta rota inversa é obtida incluindo no pacote `RREQ` os saltos por onde o pacote foi sucessivamente passando. À semelhança dos protocolos de encaminhamento das redes com fios, cada nó não sabe que rota cada pacote vai ter – apenas sabe que para o destino D o próximo salto é H.

O AODV usa o conceito de número de sequência para marcar cada `RREQ`. Este conceito vem do DSDV [26], uma anterior proposta de protocolo de encaminhamento de alguns autores do AODV. O DSDV é uma aproximação do RIP (redes com fios) às redes ad hoc. O RIP, e genericamente os protocolos de encaminhamento "distance-vector", sofre do

problema de contagem-até-ao-infinito [27] (o RIP limita a 16 iterações) e é notório que a importância dos números de sequência é fulcral. A solução pode ser interpretada como um mecanismo simples, embora eficiente, de manter uma variável de estado, invariante, que mantenha a história da rede. Para o encaminhamento, a história da rede é apenas a quebra e formação de rotas (eventos discretos) como repercussão imediata da quebra e formação de ligações entre nós da rede. Os números de sequência marcam, portanto, do ponto de vista do encaminhamento, a evolução da rede, quantitativa e não qualitativa (irrelevante para o encaminhamento) e o estado actual da tabela de encaminhamento.

A tabela de encaminhamento consta de entradas com os seguintes elementos: destino; próximo salto; número de sequência do destino; tempo de validade da rota; vizinhos que usam o nó; número de saltos. Estes campos servem os seguintes propósitos:

- *escolher a melhor rota dentro das* `RREP` *recebidas*. Assume dois cuidados: a melhor escolha das rotas publicitadas (o AODV apenas permite uma entrada por destino) e a actualidade de cada rota. O uso de números de sequência permite distinguir as várias resposta de rotas, de forma a que a cada resposta chegada, a mais actual (número de sequência mais alto) seja introduzida na tabela de encaminhamento. A escolha final da rota é baseada no número de saltos (transmitido no `RREP` e copiado da `RREQ`) dentro da rota mais actual. Uma consequência deste mecanismo de escolher a rota mais actual é que nem sempre a rota mais curta é a que é utilizada.
- *purgar a tabela de rotas inválidas*. O AODV mantém em cada nó um temporizador que é usado para vários fins incluindo a eliminação de entradas da tabela de encaminhamento ao fim de um tempo pré-especificado sem uso. Desta forma, rotas que sejam inválidas, por alguma razão irrelevante para o protocolo de encaminhamento (e.g. um salto já não existente), são apagadas da tabela de encaminhamento, embora correndo-se o risco de eliminar rotas ainda válidas. Um segundo mecanismo de refrescamento da tabela de encaminhamento usa pacotes `HELLO` para conhecer (ou confirmar) os vizinhos directos. O protocolo também permite que se use a L2 para este efeito – no IEEE802.11 são as

transacções RTS/CTS/ACK. Notar que o AODV apenas mantém o próximo salto para um dado destino (e não rotas completas como o DSR, a discutir brevemente) pelo que o conhecimento explícito dos vizinhos de cada nó activo é suficiente para determinar quebras de ligações e, consequentemente, destinos não válidos. Sempre que um vizinho não responde (HELLO ou RTS/CTS), um pacote RERR é transmitido, em difusão limitada – apenas vizinhos que usam a rota em questão (campo vizinhos-que-usam-o-nó na tabela de encaminhamento). Neste ponto, o número de sequência do destino é incrementado (desactualizando as rotas anteriores). Para obter uma rota para o destino entretanto tornado inacessível, é necessária uma nova descoberta independente.

- *manter as rotas válidas*. Este mecanismo é complementar do anterior pelo que está implícito nele. Adicionalmente, cada nó que encaminhe um pacote (ou transmita) reinicia o contador de validade da entrada na tabela de encaminhamento. Notar que, se não houver tráfego para um dado destino através de um dado nó, a entrada na tabela de encaminhamento é eliminada ao fim de algum tempo, mesmo que a rota ainda seja válida, como se viu acima.

- *controlo da difusão de RREQ*. O protocolo marca cada RREQ com um número de série (RREQ-id) de forma a que o par (origem; RREQ-id) identifica cada pedido univocamente e RREQ antigos são ignorados. Cada RREQ tem associado uma condição de tempo-expirado, grande para que o pedido cubra a rede toda mas suficientemente pequeno para que a inundação se extinga por si só em caso de anomalia. Finalmente, como cada nó pode responder a um RREQ mesmo não sendo o destino (desde que possua uma rota para o destino com um número de sequência igual ou superior ao indicado na RREQ), a inundação nunca é verdadeiramente global.

- *rotas não cíclicas*. A conjugação de RREQ por inundação, temporizadores e a marcação das entradas na tabela de encaminhamento (número de sequência) de cada nó garante que as rotas não são cíclicas [25].

DSR.

Tal como o AODV, o DSR [30] é um protocolo reactivo por apenas necessitar conhecer rotas para os destinos com quem efectivamente troca tráfego de transporte num dado instante e por só as conhecer quando as necessita. Não usa de todo qualquer tipo de tabela de encaminhamento mas, em alternativa, mantém uma memória local de rotas que pode ser muito extensa, consoante o grau de proactividade desejado. O encaminhamento dos pacotes em cada nó é regulado pelo próprio pacote de dados já que a sequência dos saltos vem no próprio pacote IP.

A obtenção de rotas é muito semelhante à descrita para o AODV: um pacote `RREQ` com o destino pretendido é transmitido em difusão até chegar ao nó destino ou alcançar o limite de saltos predefinido. As respostas, `RREP`, são dadas pelo próprio nó destino, via rota invertida (a `RREQ` vai acumulando os saltos por onde passou) ou por nós que, conhecendo uma forma de chegar ao destino respondem gratuitamente (para usar um termo próximo do original "gratuitious replies"). Cada rota é colocada em memória local e na altura de transmitir um pacote é seleccionada a rota mais curta entre as possíveis para o mesmo destino. A única forma de uma rota ser eliminada da memória local é ser detectada uma ligação partida: o nó que detectar sinaliza pelo caminho inverso o nó origem. Pelo caminho, os nós vão actualizando as respectivas memória locais. Várias técnicas de optimização são usadas:

- *múltiplas rotas*. Como referido, os nós intermédios podem responder pelo nó destino se conhecerem uma rota, continuando a encaminhar o pacote. O nó destino responde a todos os pacotes `RREQ` que lhe chegarem até a um certo limite de saltos. Estes dois mecanismos podem gerar muitas rotas para cada par origem-destino (S-D).
- *salvação*. Se um nó não conseguir encaminhar para o próximo salto, faz seguir uma indicação ao nó origem (`RERR`) mas tenta, consoante a sua configuração, fazer chegar o pacote ao destino alterando-lhe a rota usando a sua própria memória local. Esta técnica tem o potencial de criar pacotes duplicados na rede mas pode optimizar a taxa de entrega de pacotes.
- *actualização parcial da memória local*. Ao receber ou encaminhar um `RERR`, cada nó apenas muda a memória local para o troço afectado pela quebra da

ligação. Ou seja, se na rota A→B→C→X→Y→D a ligação X→Y está quebrado, a rota até X continua a ser válida e é alterada para A→B→C→X em vez de ser totalmente eliminada pelo que pode continuar a ser usada entre A e X.

- *inversão trivial de rotas*. Se o caminho S→D é conhecido, o caminho D→S é trivial, bastando inverter a rota inscrita no pacote.

- *escuta de tráfego de terceiros*. Os nós podem aprender e eliminar rotas escutando tráfego alheio. Como as rotas vêm nos pacotes IP, muitas ligações L2 podem ser deduzidas. É neste ponto que o DSR pode ser considerado, dum ponto de vista conceptual, híbrido. Se se notar que, para além desde mecanismo, entre o mesmo par origem-destino podem ser conhecidas várias rotas (múltiplos RREQ), no decurso da operação, num tempo limitado e dependendo do grau de mobilidade da rede, grande parte da topologia da rede pode ser conhecida.

- *sinalização nula para redes estáticas*. Se uma rede não é móvel, as rotas descobertas são válidas indefinidamente pelo que a descoberta de rotas tende para zero[1].

- *fluxos*. A versão 08 [30] do 'draft' inclui a possibilidade de se usar o conceito de fluxo. Um fluxo é uma sequência de pacotes em que os saltos intermédios sabem, por alguma indicação no pacote (e.g., identificação do fluxo), qual o próximo salto. Mantém-se tudo igual, mas ao estabelecer-se um fluxo os nós intermédios são sinalizados para criarem uma tabela de encaminhamento, ou uma entrada nela, que faça encaminhar os pacotes com um identificador de fluxo sempre para o mesmo salto. Desta forma, o peso do encaminhamento-na-origem é praticamente eliminado, à custa de apenas sinalização. A forma de desfazer os fluxos é explicitamente (sinalização) ou por uma condição de tempo-expirado.

[1] Pode não ser nulo porque a tecnologia L2 nem sempre discerne entre quebra de ligação real (PHY) e quebra de ligação inferida (MAC). É especialmente válido para o IEEE802.11, como foi visto atrás, com a Determinação Virtual de Meio Livre. Mesmo uma rede com nós imóveis e sem quaisquer perturbações de sinal pode ter nós momentâneamente sem conectividade (gerando um pacote de routing RERR) porque o MAC não recebeu um CTS ao fim de 7 tentativas. O vizinho pode simplesmente não ter enviado o CTS porque a fila de saída não serviu o pacote a tempo de completar a transacção.

A simplicidade geral do DSR, conseguindo em simultâneo funcionalidades com grandes potencialidades, tem, contudo, algumas desvantagens:

- *rotas empatadas*. Como o DSR depende de sinalização de terceiros para eliminar rotas da memória local (RERR), se houver algum tipo de anomalia (como perda dum pacote RERR), certas rotas podem permanecer indefinidamente na memória local que pode aumentar injustificadamente. Igualmente, se um nó usar uma rota apenas durante um curto período de tempo e a rota se invalidar apenas depois de a usar, só quando voltar a usa-la a memória local será actualizada, após o RERR da rede [31].
- *peso de encaminhamento-na-origem*. Este tópico pode criar problemas de escalabilidade na medida em que, por salto, o pacote ganha 16 bytes (IPv6). As medidas tomadas para contornar este problema específico consistem em limitar o número de saltos e o uso de fluxos, explicados anteriormente.

O impacto real destas desvantagens será discutido adiante de forma mais integrada aquando da análise comparada dos protocolos. Especialmente para o problema de encaminhamento-na-origem, sugere-se que o problema pode não ser significativo em aplicações reais.

2.4.3. Outros protocolos.

Conceitos.

Numa abordagem diferente dos protocolos por-pedido ("on-demand"), os protocolos proactivos seguem a filosofia das redes com fios: usar, com mais ou menos pormenor e antecipadamente, o grafo da rede, gerar uma tabela de encaminhamento e tomar a decisão do "porto" de encaminhamento com base nela.

OLSR.

O OLSR [32] esforça-se no sentido de optimizar os algoritmos de encaminhamento "estado-da-ligação" ("link-state" – descoberta do caminho mais curto, segundo uma qualquer métrica, usando o algoritmo de Dijkstra) das redes com fios para o meio ad hoc. Por conseguinte, até toda a topologia ser conhecida e o algoritmo convergir, não pode haver tráfego (ao contrário dos protocolos "distance-vector", baseados no algoritmo Bellman-Ford, cuja acção não necessita do conhecimento de todo o grafo da rede). O OLSR é, em parte, o transporte para a camada IP de um conceito que já existe na HiPeRLAN/2. Por outro lado, parece ter-se baseado no CEDAR que propõe também que haja nós com funções de encaminhamento e outros que não, como será referido à frente[1]. No encaminhamento 'link-state', cada nó executa as seguintes tarefas:

- descoberta dos vizinhos com pacotes `HELLO`, enviando os seus próprios vizinhos;
- determinação do custo de cada ligação (1 salto), definida a métrica;
- inundação da rede com a informação dos passos anteriores, de todos para todos os encaminhadores (em redes ad hocs são todos os nós);
- algoritmo de Dijkstra para determinar os caminhos mais curtos, definida a métrica.

Reduzindo o número de nós que participam activamente, e em cada momento, no encaminhamento, a proposta do OLSR consiste em optimizar a inundação inicial e a quantidade de nós envolvida no tráfego de encaminhamento de manutenção. Esses nós, os MPR (Multi-Point Relay), são uma entidade abstracta da rede na medida em que cada nó selecciona os seus próprios nós MPR (neste sentido, não são nós especiais na rede global) e entrega-lhes todo o tráfego que pretendem transmitir: só se o nó a quem entregou o pacote for um MPR é que há retransmissão. Com mais rigor, um nó apenas retransmite um pacote se a primeira cópia desse pacote tiver vindo de um conjunto MPR.

A escolha dos nós MPR, de cada nó, segue a seguinte regra: MPRs de A são os seus vizinhos tal que cada seu vizinho a 2 saltos esteja a 1 salto de um seu MPR. Na prática, significa que todos os nós têm de estar a menos de um salto do MPR de cada nó.

[1] Refere-se que o OLSR se inspira no CEDAR e não o contrário porque o CEDAR foi tornado público em 1998 e o OLSR em 2001.

Tendo em conta que o objectivo final é que chegue a informação das ligações de cada nó a todos os nós, esta optimização elimina duplicações de pacotes na rede. É também claro que esta optimização funciona tanto melhor quanto mais densa é a rede (a mínima fracção de nós que são MPR).

2.4.4. Análise comparativa.

Protocolos reactivos e proactivos.

Em primeiro lugar, a Fig. 11 mostra a árvore genealógica de alguns protocolos de encaminhamento (baseada em [33]). "Pre-computed" são protocolos ou algoritmos que usam a topologia da rede para determinar a rota mais curta.

Nesta dissertação foca-se o AODV, o DSR e o OLSR. As métricas mais usadas para caracterizar o desempenho de um protocolo de encaminhamento em redes ad hoc são as seguintes (relacionadas com as propriedades de um protocolo de encaminhamento indicadas na secção 2.4.1):

- *rapidez* (tempo de convergência ou complexidade temporal): (i) inicial, quando a rede arranca; (ii) na manutenção de rotas.
- *peso do tráfego de encaminhamento* (complexidade de sinalização e de manutenção): (i) inicial, quando a rede arranca; (ii) na manutenção de rotas.
- *suporte*: (i) multi-rota; (ii) difusão de grupo; (iii) outros
- *complexidade de implementação*: (i) memória; (ii) complexidade algorítmica
- *qualidade e disponibilidade das rotas*: a primeira parte está intimamente relacionada com a métrica usada do protocolo de encaminhamento (normalmente é o caminho mais curto); a segunda relaciona-se com a resposta do encaminhamento quando a rota actual deixa de ser válida.

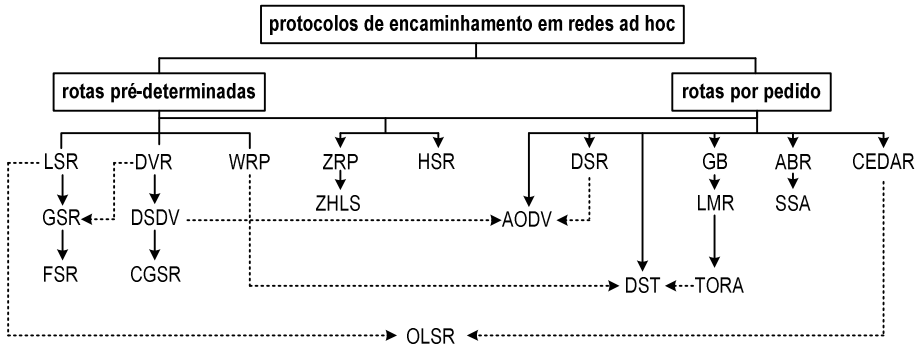


Fig. 11. *Genealogia de alguns protocolos de encaminhamento. Alguns ramos do primeiro nível são algoritmos e não protocolos formalizados: LSR: "Link-state routing"; DVR: "distance-vector routing"; GB: Gafni-Bertsekas (adaptado de [33])*

Não é fácil comparar protocolos com base apenas na sua classificação: tanto o cenário de utilização como a própria indefinição típica de uma classificação generalista inibem comentários elaborados. A perspectiva tomada é a de comparar os protocolos num cenário realista e em condições iguais, discutindo depois a capacidade de melhoramento de cada um em função dos efeitos observados. Pode contudo fazer-se alguns comentários gerais sobre certos protocolos.

De um ponto de vista da comparação entre protocolos de encaminhamento reactivos e proactivos, a única métrica clara que os distingue é o esforço de routing ("overhead"). Os protocolos proactivos exigem o conhecimento da topologia toda e em qualquer instante o que significa que, quando a rede arranca, há troca de sinalização que pode nunca ter utilidade futura. Podem também ser significativamente mais lentos a responder a pequenas alterações na topologia – porque o algoritmo de suporte o exige, ou porque só se mantém em memória local uma alternativa de rota. Por outro lado, excluindo efeitos de mobilidade, a rota nos protocolos proactivos está sempre disponível. Neste ponto, é de realçar as possibilidades do TORA [21]. Por um lado, mantém sempre mais do que uma alternativa em memória (até pode ser usado em multi-rota); por outro, pequenas alterações da topologia podem ser resolvidas localmente. Notar também que o TORA não é considerado como reactivo ou proactivo puro, em parte devido a estas características: se por um lado é reactivo quando arranca (daí estar no lado direito da Fig. 11), durante operações de manutenção cada nó tem conhecimento de parte da topologia da rede.

De resto, é difícil comparar a priori famílias de protocolos. É mais realista analisar e/ou comparar directamente protocolos formalizados. Aliás, é notória a escassez de trabalhos onde se comparam os vários protocolos existentes, em oposição à quantidade de trabalhos que descrevem os respectivos funcionamentos de um ponto de vista meramente funcional. Existem trabalhos que analisam frente-a-frente alguns protocolos mas as condições em que alguns trabalhos foram realizados não advogam em favor da generalidade dos resultados.

comparação breve dos protocolos analisados.

Em relação ao OLSR, é frequente ver-se escrito que é um protocolo especialmente adequado para redes densas e grandes. O seu autor refere-o no artigo onde apresenta o protocolo. Contudo, o autor parece mais referir-se às condições ideais de funcionamento do protocolo e menos à adequação do protocolo face a outros. Efectivamente, o protocolo tem o seu meio preferencial de utilização em redes grandes e densas. A Fig. 12 mostra que o tráfego de controlo do protocolo de encaminhamento (a métrica "número de retransmissões" é o número de trocas de tabela de vizinhos) diminui significativamente com a densidade da rede [34] quando comparado com a inundação tradicional. A justificação deste facto é o uso de MPRs. É compreensível, já que, se imaginarmos que, para uma densidade tal em que todos os nós estão a 2 saltos de distância ou menos, há apenas 1 MPR (o MPR é o nó que está entre dois extremos).

Na Fig. 12, a grandeza *q* relaciona-se directamente com a probabilidade de um nó estar ligado a todos os outros[1]. Com base nestes resultados, dir-se-ia que uma rede densa e grande, do ponto de vista do OLSR, significa $q > 0.3$ (Fig. 12a) e $N > 200$ (Fig. 12b, $q \sim 0.9$). Não são, contudo, pressupostos fáceis. Como já foi discutido, uma rede $N=200$ já é muito grande (com 1000 nós é enorme). Se $q=0.3$, então, para 200 nós, cada nó tem ligação directa com 60 outros nós (em média no tempo). Para $q=0.9$ (as condições da Fig. 12b), cada nó teria de ter conectividade directa com 180; para a Fig. 12a, cada nó, com $q=0.3$, teria conectividade com 300 nós o que significaria uma estação móvel a cada ~5 metros (assumindo um alcance de 100m para a camada física).

[1] Os autores usaram um grafo aleatório com *N* vértices e probabilidade de ligações *q*.

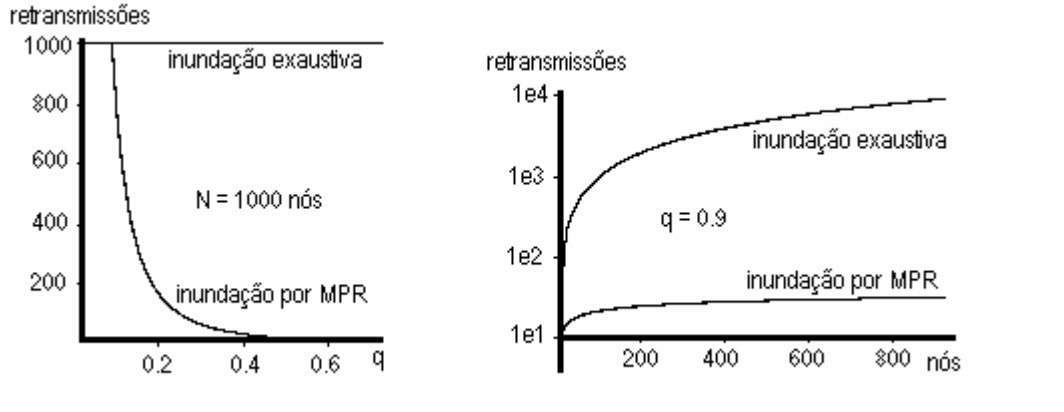


Fig. 12. *Peso do tráfego de controlo do OLSR em função da densidade da rede (q: probabilidade de um nó estar ligados a todos os outros) e do número de nós (N). Resultados analíticos.*

Em última análise, a mais importante métrica de qualquer protocolo de encaminhamento é a perda de pacotes (exclusivamente devido ao encaminhamento – é necessário ter isto sempre presente). De seguida, talvez se possa considerar a latência (neste contexto, definida como o tempo entre o pedido e obtenção de uma rota) e atraso da comunicação e da rota. Enquanto a primeira indica a rapidez com que o protocolo responde a mudanças de topologia (e início de tráfego), a segunda mostra a qualidade das rotas que fornece. O OLSR garante, pelo algoritmo subjacente (Dijkstra) que as rotas são as mais curtas – na maior parte das vezes, deve ser a melhor métrica de qualidade de uma rota, mas nem sempre será assim (mesmo excluindo protocolos específicos, como os que têm em conta a energia do nó). O AODV e o DSR não garantem a rota mais curta, mas tendem a fornece-la (todos os nós tenderão a respondem ao `RREQ` e a rota mais curta é escolhida por chegar primeiro). Por conseguinte, não é de se esperar grandes diferenças de atraso para os dois tipos de protocolos nem de perdas de pacotes – eventualmente, o OLSR tem alguma vantagem em ambientes de baixa mobilidade por garantir que é sempre a rota mais curta mas perde vantagens porque injecta tráfego periódico e persistente na rede. Com mobilidade, a latência da redescoberta de rotas ganha peso e o protocolo mais rápido e com menos tráfego de controlo terá menos atraso num intervalo de tempo grande. Para um tempo de tráfego pequeno (em comparação com a mobilidade da rede e,

consequentemente, com o tempo médio que a rede mantém a mesma topologia), o protocolo mais eficiente será o que tiver menos tráfego de controlo e o que fornecer rotas mais curtas. Tendo em conta que todos os protocolos analisados nesta secção (OLSR, AODV, DSR) usam, na prática, a rota mais curta que o algoritmo de encaminhamento lhes forneceu[1], a diferença está no tráfego de controlo gerado.

Existe um trabalho que investiga com base em simulações o desempenho comparado do AODV, DSR e OLSR [35]. Esse trabalho foi alvo de análise muito atenta sendo que o principal comentário comporta algum cepticismo face às várias conclusões que os autores retiram. Foi, contudo, decidido que uma crítica fundamentada e exaustiva sairia fora do âmbito desta dissertação, na medida em que o tema central desta dissertação não é encaminhamento.

comparação entre AODV e DSR.

Existem outros trabalhos de comparação de protocolos: [38][39][40][41]. Consideram-se dois trabalhos para comparar DSR com AODV:

- Broch'98 [38]: AODV, DSR, TORA e DSDV – as condições de simulação estão significativamente pormenorizadas, incluindo a parametrização de cada protocolo
- Perkins'00 [39]: AODV e DSR – muito semelhante a Broch'98

As condições de simulação são determinantes para a validação e inferência de conclusões. Na tab. 5 indicam-se os parâmetros das simulações subjacentes aos resultados.

Os resultados obtidos em [43] e [38] são essencialmente concordantes. No que diferem, as condições de simulação e o funcionamento distinto dos protocolos tenderão a explicar as diferenças. A referência [38] compara também o TORA e o DSDV mas não serão discutidos.

[1] O AODV usa como primeiro critério o número-de-sequência; de seguida, privilegia a rota mais curta.

	BROCH'98 [43]	**PERKINS'00 [38]**
protocolos	AODV, DSR, TORA, DSDV	AODV, DSR
simulador	ns-2, monarch	ns-2, monarch
camadas simuladas (abaixo de IP)	. sinal (waveLAN, alcance: 250m) . PHY (2 Mbps) . MAC (IEEE802.11 DCF) . ARP	mesmo que [BROCH'98]
. nós . filas de saída ("buffers")	. 50 nós . 50 pacotes	. 50 ou 100 nós . 64 pacotes
simulações (cenários diferentes)	. 900s de duração . 10 repetições	. 900 s . 5 repetições
mobilidade . modelo . geometria . velocidade máx . tempos de pausa	. RWP . 1500m x 300m . 20 m/s ou 1 m/s . 7, de 0 a 900s	. igual a [BROCH'98] . 100 nós: 2200m x 600m . mas só 20 m/s
fontes	. CBR (UDP) . 4 pacotes/s . 10, 20 e 30 fontes . pacotes[1] de 64 B	. CBR . pacotes de 512B
métricas	. perda de pacotes . tráfego não de utilizador ("overhead") . rota mais curta	. perda de pacotes . extremo-a-extremo atraso . tráfego não de utilizador ("overhead")
notas gerais	. peso do tráfego de encaminhamento não considerando ARP (varia consoante L1&L2) . determinação da rota mais curta extrasimulador	. atraso é calculado incluindo armazenamento ("buffering") dos pacotes

tab. 5. *Parâmetros de simulação de trabalhos onde se comparam protocolos de encaminhamento.*

As principais conclusões comuns para cargas baixas e médias (abaixo do limiar de congestionamento generalizado – perda de pacotes da ordem de 5%) são as seguintes:

- a fracção de pacotes perdidos é essencialmente a mesma para os dois protocolos e muito baixa, tendo o DSR ligeiramente melhores resultados principalmente para redes densas.

- o atraso também é semelhante para os dois protocolos, com o DSR a obter, em situações de maior mobilidade, resultados significativamente melhores

[1] Os autores tentaram inicialmente usar pacotes de 1024 B. Contudo, devido à implementação do modelo de propagação de sinal que usaram, verificaram que facilmente causava congestionamento. Decidiram que o melhor compromisso era usar pacotes de tamanho reduzido, 64 B, já que o artigo foca essencialmente a capacidade dos protocolos de encaminhamento de determinaram boas rotas S-D.

▪ o DSR, ao contrário do que frequentemente se afirma, é um protocolo bastante escalável. A prova é que o peso do tráfego de encaminhamento normalizado ao número de nós é praticamente constante. Ou seja, o tráfego de encaminhamento aumenta linearmente com o número de nós. Para além disso, é significativamente menor do que o tráfego de encaminhamento do AODV (chega a 500%). Contudo, é necessário ter em conta dois efeitos, um deles muito subtil, como se nota em [43] e se descreve de seguida.

O peso do DSR é maior do que o efectivamente medido quando se tem em conta o MAC. Uma contabilidade do tipo de pacotes que compõem o tráfego de encaminhamento dos dois protocolos apresentada no artigo mostra que o AODV tem quase 90% de pacotes `RREQ` e o DSR é dominado por pacotes `RREP` e `RERR`. Como pacotes `RREQ` são difusão e, portanto, sem `RTS/CTS/data/ACK`, os pacotes `RREP` e `RERR`, em muito maior proporção no DSR, têm um peso efectivo bastante maior do que `RREQ`. No artigo, os autores notam que se entrarem com este factor, os pesos compensam-se. Contudo, e acrescentando algum rigor à discussão, é necessário também atender ao facto de que, após um `RTS/CTS`, o acesso ao meio está quase garantido (são IFS muito curtos – SIFS) o que reduz significativamente o efeito. Um estudo sobre a proporção efectiva de tempo de meio adquirido entre tramas de difusão e ponto-a-ponto ("unicast") teria de ser feito para validar estes resultados. Acima de tudo, e pareceu ser a intenção dos autores, não é rigoroso falar de um factor 4:1 (compensando o ganho de 5:1 do DSR se se incluir também o peso relativo de `RREQ` em cada protocolo) apenas tendo em conta que a transacção `RTS/CTS/data/ACK` tem 4 pacotes e um de difusão apenas 1.

O segundo efeito consiste no facto de o DSR perder vantagens em termos de quantidade de bits enviados. O encaminhamento-na-origem tem esta desvantagem. Mas note-se que o principal problema das redes ad hoc baseadas em IEEE802.11 é o acesso ao meio e, assim que se inicia a transmissão, o tempo gasto em transmissão efectiva é extremamente reduzido. Exemplo: 30 ms para o atraso médio de uma rota com 4 saltos e com carga média é muito razoável. Considerando os 11 Mbps do IEEE802.11 (a maior parte das simulações são para

esta taxa de linha) e um pacote de 1000 bytes (sensivelmente o tamanho máximo do IEEE802.11), e para 4 saltos, a transmissão efectiva de bits dura 3 ms. Portanto, e na maior parte das situações, o tamanho dos pacotes não é dominante. Em segundo lugar, e de um ponto de vista de fragmentação dos pacotes IP (um acesso ao meio por fragmento), se for IPv6, a rota no pacote ocupa 16B e se uma rota média tiver 4 saltos, representa 1% do pacote máximo do IEEE802.11. Portanto, não será devido ao encaminhamento-na-origem que as tentativas de acesso ao meio ou o atraso irão aumentar drasticamente. Finalmente, existem técnicas que fazem reduzir substancialmente o peso de encaminhamento-na-origem. Uma delas, descrita anteriormente, consiste na técnica dos fluxos do DSR; outras serão apresentadas oportunamente nesta dissertação.

O desempenho de cada protocolo segundo [43], o único trabalho que apresenta resultados para o débito-entre-extremos, está globalmente descrita na Fig. 13. Estes resultados em particular são um tanto desconcertantes e difíceis de compreender. Notando que, quando o atraso médio dos pacotes sobe para um valor da ordem de grandeza de 1 segundo (rede em franco congestionamento), a zona de operação a partir dos 200 kbps é de utilidade questionável. Seria necessário saber qual a taxa de entrega de pacotes em cada ponto do gráfico. Essa informação reforçaria a qualidade das estatísticas e talvez permitisse outras conclusões que são, assim, de validade questionável.

De um ponto de vista global, e integrando toda a literatura, pode retirar-se as seguintes conclusões sobre o "melhor" protocolo de encaminhamento:

- O ambiente de simulação e de operação é determinante para as conclusões finais.
- Protocolos proactivos parecem ter pior desempenho do que os reactivos.
- AODV e DSR são, seguramente, dos protocolos com melhor desempenho.
- Entre AODV e DSR, é difícil concluir sobre a superioridade de um ou outro, principalmente porque os trabalhos de simulação deixam muitas reservas. O DSR parece levar vantagem nas condições que neste trabalho se considera ser

as mais úteis para redes ad hoc genéricas, nomeadamente, abaixo do ponto de congestionamento.

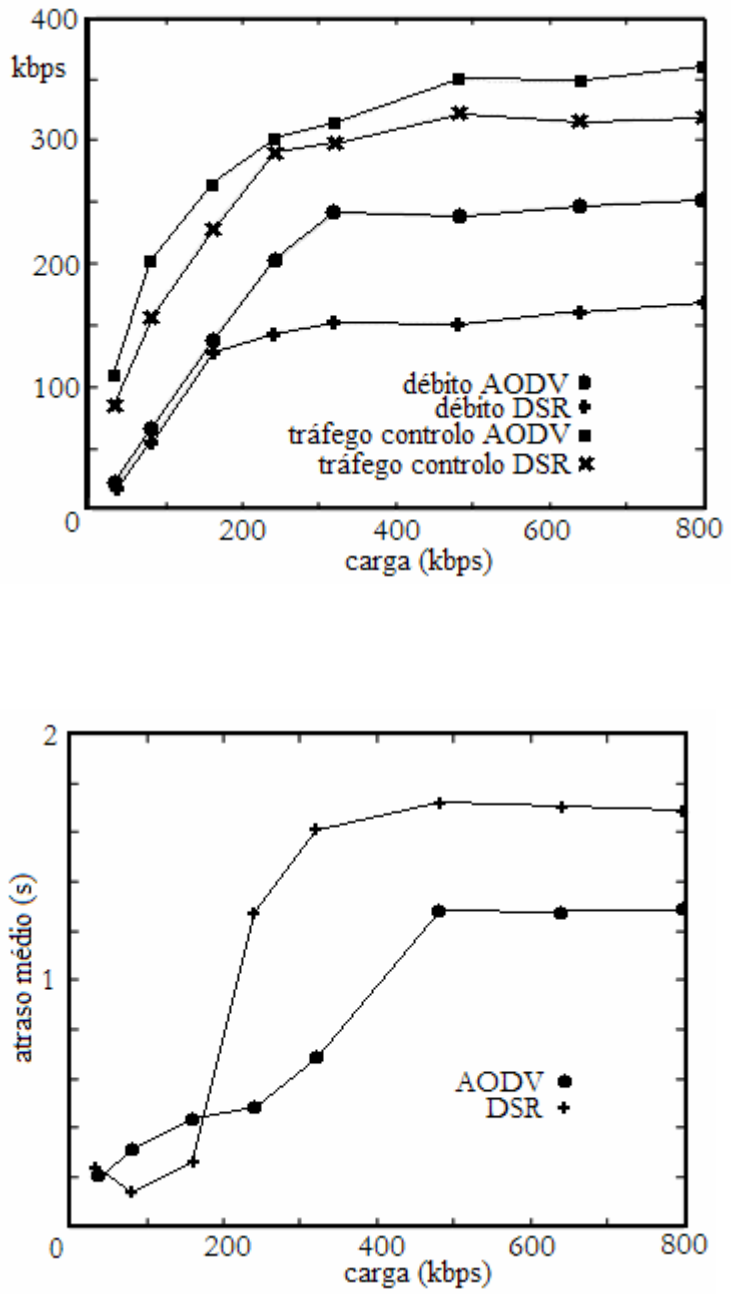


Fig. 13. *Desempenho comparado para várias cargas do AODV e DSR [43]. Condições: 100 nós, 20 m/s, tempo de pausa 0 s.*

- Começa a ser necessário haver um guia de procedimentos de simulação para normalizar as métricas obtidas em simulação ou implementação real ("testbed"). Um exemplo: até que ponto faz sentido considerar o atraso de pacotes entregue via mecanismos de salvação ("salvaging")? Estes pacotes devem entrar para a taxa de entrega de pacotes mas não deveriam contar para o cálculo da média do atraso. O atraso é uma métrica associada não à entrega de pacotes mas sim à "rapidez" de entrega quando um fluxo de transporte (UDP, TCP, SCTP, RTP, ...) está efectivamente estabelecido. Um pacote entregue com recurso intensivo à memória local com, eventualmente, mudança de rota a meio do percurso até ao destino, pode ter atrasos da ordem de vários segundos. Não é útil para praticamente nenhuma aplicação comum. Como este exemplo há outros e impõe-se, portanto, criar um espaço de discussão sobre parâmetros de simulação que permita validar uma simulação apenas referenciando-a a essa iniciativa e, mais importante, poder haver comparações directas entre protocolos.

2.4.5. Técnicas genéricas de melhoria do encaminhamento.

Introdução.

De um certo ponto de vista, a camada de encaminhamento deve apenas fornecer conectividade de rede (multi-salto) ao nó. Este deverá ser o resultado fundamental do encaminhamento. Em seguida, seguindo uma espécie de ordem de funcionalidades fundamentais, deverá ser nesta camada que todas as rotas S-D existentes são descobertas. Neste ponto, e começando a entrar no tema desta dissertação, começa a poder-se falar de QdS ao admitir ser possível que certas rotas sejam mais "favoráveis" do que outras. Por outras palavras, pode imaginar-se uma sub-função superior da camada de encaminhamento que sirva solicitações do plano[1] de QdS. Alguns tópicos deste capítulo serão revisitados ao longo da dissertação. Para começar, incluem-se algumas técnicas para optimizar o DSR e que podem ilustrar a problemática.

[1] A designação de "plano de QdS" ficará clara no próximo capítulo.

Optimização do DSR.

As principais críticas que se têm feito ao DSR consistem no tráfego de encaminhamento implicado no recurso a encaminhamento-na-origem e a validade das rotas na memória local ("buffers"). Neste trabalho de mestrado, o DSR foi objecto de especial atenção. Não são apresentados resultados detalhados, por não constituir material central do trabalho, mas duas técnicas foram concebidas, embora não validadas, que resolveriam os dois problemas acima indicados sem introduzir grande complexidade quer ao nível do processamento ou de tráfego na rede.

Em primeiro lugar, para atenuar o problema do peso do encaminhamento devido ao encaminhamento-na-origem, propõe-se dividir a camada de encaminhamento em duas. A superior é o encaminhamento em si mesmo e a inferior um processo de tradução de endereços IP em endereços especiais únicos na rede. Notar que o encaminhamento no DSR é uma camada de rede e não um processo autónomo esporadicamente chamado. Dado que todos os pacotes têm de passar pelo módulo de encaminhamento para remoção do cabeçalho, todos os pacotes são sujeitos a este processo. O mecanismo proposto foi designado por ARP2.5 por ser um processo semelhante ao ARP L2. A solução passa por atribuir a todos os nós um endereço curto. O tamanho dos endereços seria função do tamanho da rede mas, tipicamente, 8 bits (até 256 nós), seria suficiente para a esmagadora maioria das aplicações actuais.
Pormenorizadamente, cada nó seguiria o seguinte algoritmo:

1. um nó que se associasse a uma rede ad hoc, criaria o seu próprio endereço ARP2.5 baseado, de preferência, num identificador que já seja globalmente único – p.ex., o endereço MAC do equipamento.
2. o nó executaria um processo de detecção de endereços duplicados (DAD), muito à semelhança das operações de autoconfiguração do IPv6.
3. os endereços para encaminhamento-na-origem que seguem no campo 'Opções' do pacote IP seriam estes endereços.
4. cada vez que a subcamada de routing fosse chamada a tratar um pacote, os endereços locais seriam reconvertidos em endereços IPv6.

Com este esquema, os endereços de 16 bytes seriam reduzidos para 1 byte, reduzindo o peso das rotas até 16 vezes, no melhor dos casos.

O segundo problema, a existência vitalícia de rotas inválidas na memória local dos nós, ocupando memória e gerando erros na rede com o consequente tráfego de encaminhamento (RERR), seria resolvido à custa de sondas enviadas periodicamente. Cada rota seria um objecto ao qual se associaria um intervalo de tempo desde a última verificação. Um processo periódico verificaria cada rota. Se uma rota não tivesse sido verificada ao fim de Δt_{check}, o processo de verificação era invocado. Este processo consiste em enviar um pacote CHECK_ROUTE com a rota a ser verificada. Se a rota não for válida, um salto intermédio iria sinalizar a origem; senão, chegaria até ao destino, validando a rota. A origem teria conhecimento deste facto se o destino respondesse ou assumindo ao fim de algum tempo que, não havendo um RERR, a rota continua válida.

Uma outra possível optimização do DSR para resolver este problema consiste em RREQ esporádicos. O DSR apenas faz pedido de rotas se não existir nenhuma na memória local do nó: é possível que o nó destino esteja a 2 saltos[1] e o nó origem persistir em usar uma rota mais comprida. RREQ periódicos permitiria eliminar este problema, embora à custa de mais tráfego de encaminhamento na rede. Excluindo inundações periódicas, por um lado introduziria tráfego de encaminhamento pesado na rede; por outro lado, a tendência de tráfego de encaminhamento nulo para redes estáticas desapareceria. É um claro compromisso que necessita de alguma investigação.

A melhor rota.

Em primeiro lugar, havendo múltiplas rotas disponíveis para o mesmo destino, é necessário um critério de selecção. A melhor rota pode ter uma definição mais lata. Certos protocolos de encaminhamento das redes com fios (e.g. OSPF) definem a rota mais curta como aquela que apresenta uma métrica composta que não é necessariamente a rota com menos saltos: pode ser determinada por uma fórmula de cálculo ponderando saltos, largura de banda, atraso, fiabilidade heurística de cada ligação, etc. Em redes ad hoc, a

[1] A situação de o destino estar a 1 salto não participa deste problema. Antes de o nó transmitir, o DSR permite efectuar uma interrogação ARP. Caso o destino seja vizinho directo, o pacote é entregue via MAC.

primeira abordagem é escolher a rota mais curta. Isto parece decorrer de dois motivos imediatos. Em primeiro lugar, porque não há, tipicamente, forma simples de ordenar saltos por qualidade. À excepção de cenários específicos, como protocolos orientados à conservação de energia ("power-aware"), qualquer salto a priori é equivalente a outro de um ponto de vista de qualidade. A questão é como obter informação de cada salto – quando muito, é o próprio protocolo a implementar formas de análise como requisito fundamental para o problema que se propõe resolver (usando o exemplo de um protocolo power-aware, cada nó pode incluir no tráfego de controlo o seu nível de energia). Em segundo lugar, e ligado ao primeiro motivo, por uma mera questão de bom senso: na falta de informação, assume-se todos os saltos equivalentes e escolhe-se a rota mais curta por, pragmaticamente, não haver razão para escolher uma rota mais comprida. O AODV, faz-se o reparo, se tiver várias rotas à escolha, prefere usar a rota com número de sequência mais alto, correspondendo a uma rota mais recente e só em segundo lugar escolhe a que tem menos saltos.

A rota mais curta tem, a priori, menor atraso. Na generalidade, e para protocolos de transporte bem comportados como o TCP, menor atraso implica maior largura de banda; para protocolos que simplesmente fazem a multiplexagem na camada de transporte, como o UDP, pode implicar menor taxa de perda de pacotes. Contudo, é clara a orientação implícita à optimização do atraso – por si ou por influir directamente noutras métricas relevantes. Em relação à largura de banda e à taxa de perda de pacotes – eventualmente as duas outras métricas mais relevantes, em última análise – a atitude geral parece ser a de delegar confiança no atraso. Consequentemente, na rota mais curta.

De um ponto de vista de redes baseadas no IEEE802.11, é notório que a escolha da rota mais curta é a melhor quando não há informação disponível que diferencie as múltiplas rotas disponíveis, salvo quando o próprio protocolo de encaminhamento impõe medidas especiais, como acontece no AODV. Efectivamente, uma rota mais curta tem duas vantagens. A primeira é que a capacidade da rede é poupada: como foi dito atrás, cada salto adicional reduz a capacidade por nó. A segunda é que, havendo menos saltos, a taxa de acesso ao meio é menor, o que tem implicações directas numa grande zona da rede onde está o nó adicional (meio partilhado): nos nós que encaminham tráfego (cada nó transporta tráfego que não é seu), nos nós vizinhos pertencentes à rota de cada nó da rota

(enquanto o nó seguinte não transmitir, não podem transmitir) e, finalmente, em quase todos os vizinhos do nó adicional que não participam na rota[1] (meio IEEE802.11 ocupado). Portanto, e de um ponto de vista qualitativo, um nó adicional tem sempre um efeito negativo no desempenho geral da rede. A Fig. 14 [38] mostra a frequência dos protocolos com que escolhem a rota mais curta e seguintes (1 salto, 2 saltos, etc.). Infere-se imediatamente que a escolha da rota mais curta não traz, necessariamente, menor desempenho, já que o protocolo com maior desempenho não é sempre o DSR. A rota mais curta tem vantagens apenas quando se reconhece que é essencialmente um efeito local, tanto no espaço como no tempo: escolher sistematicamente a rota mais curta não é necessariamente vantajoso. Outra conclusão que se pode tirar é a seguinte. A Fig. 15 foi obtida por simples critérios geométricos e combinatórios e a proporção de rotas mais curtas não parece depender da mobilidade. Outra conclusão é que existe muitas mais rotas curtas do que rotas longas, algo compreensível de um ponto de vista de análise combinatória sobre a posição dos nós, dada uma certa mobilidade: se os nós se moverem e distribuírem uniformemente num círculo, os nós mais afastados do centro utilizarão rotas mais longas.

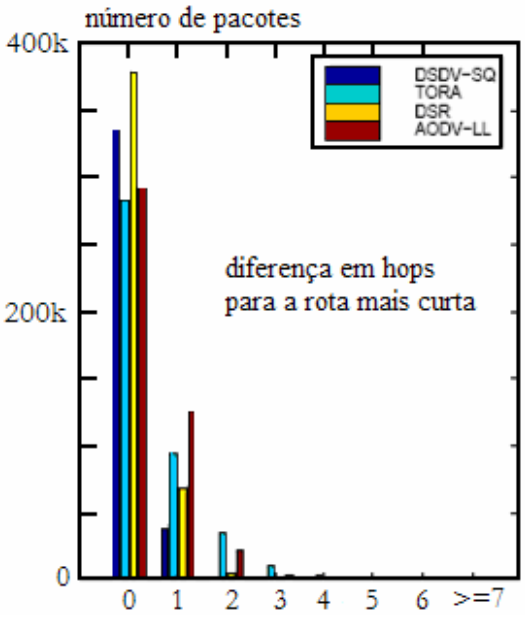


Fig. 14. *Distribuição da diferença de tamanho das rotas para a rota mais curta [38].*

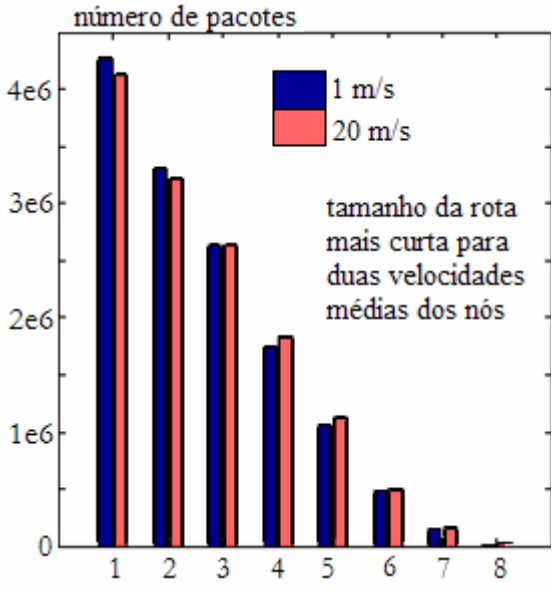


Fig. 15. *Distribuição do tamanho das rotas em relação à rota mais curta [38]*

[1] Refere-se "quase todos os vizinhos" porque, como foi dito, o IEEE802.11 tem mecanismos para, mesmo com tráfego na zona de interferência, poder transmitir para outros (NAV e RTS/CTS).

Multi-rota.

Multi-rota (MP), em contraste com caminho-único (SP), consiste em obter e usar mais do que uma rota entre um mesmo par S-D. Nas redes com fios, é tipicamente usado para fazer balanceamento de carga e multiplicar dessa forma a largura de banda disponível. O balanceamento de carga é usado no âmbito de engenharia de tráfego e da tolerância a falhas [47]. Do ponto de vista das redes ad hoc, a técnica ainda parece mais útil dada a escassez de recursos. Contudo, não é clara a real eficiência da técnica [48] [49] – primeiro dum ponto de vista de uma rede multi-salto e depois, especificamente em redes ad hoc, tendo em conta o MAC utilizado. Existem já extensões para dotar vários protocolos de encaminhamento com multi-rota como p.ex. [50] [51]. Outros protocolos usam logo de raiz MP [52] [53] [54] [55].

Neste momento, na literatura são discutidos dois problemas associados a multi-rota: como gerar rotas disjuntas e o grau dos eventuais benefícios.

A maior parte da discussão em torno dos protocolos de encaminhamento com multi-rota anda à volta da forma de se obter rotas disjuntas entre o mesmo par S-D [56] [57] [58] [59]. O problema merece atenção devido ao efeito de acoplamento de rotas [60]. Um caso flagrante será o do uso de um ponto de acesso (AP) para a rede toda. Se todos os nós se orientarem para ligações à internet via esse AP, os nós que compõem as rotas que o têm como destino (ele próprio e os nós perto dele) estarão muito carregados. Encaminhamento-na-origem tem essa capacidade automática (basta olhar para os endereços dos saltos que compõe a rota e ver se há nós em comum). Depois, como normalmente as rotas são obtidas por grosso (obtém-se o caminho e não os saltos), a forma de se seleccionar só rotas disjuntas implica, tipicamente, ginástica considerável do protocolo de encaminhamento. Atendendo a que o algoritmo tem de estar sistematicamente activo (encaminhamento sempre activo), o esforço de processamento pode ser significativo.

Contudo, este aspecto será sempre secundário já que a primeira decisão é usar ou não multi-rota. Há demonstrações analíticas, comprovadas com resultados de simulação (e.g. [49]), que constatam que multi-rota apenas compensa quando certas condições são satisfeitas. Várias métricas estão envolvidas para além da largura de banda. É necessário

considerar o benefício de multi-rota a duas dimensões: (i) mais-valia para o nó; (ii) mais-valia para a rede. Para estas duas dimensões, é necessário analisar as vantagens publicitadas para multi-rota. Consideram-se, tipicamente, as seguintes: (i) multiplicação da largura de banda; (ii) robustez de conectividade e rapidez de reacção (mais rotas disponíveis em cada instante). Em relação às desvantagens, é frequente considerar-se apenas o tráfego de encaminhamento adicional para obter mais rotas e a complexidade (obtenção de mais rotas e geração de rotas disjuntas).

Importa agora analisar MP de um ponto de vista de rede. Há duas vertentes: complexidade temporal/algorítmica (na prática, tráfego de encaminhamento) e efeitos na capacidade da rede:

- *tráfego de encaminhamento*. O tráfego de encaminhamento em si não se relaciona necessariamente com o desempenho da técnica. Quando muito, é uma componente dela que pode importar optimizar para que o desempenho em si aumente. Por isso, as técnicas têm de ser analisadas casuisticamente. A obtenção de múltiplas rotas para o mesmo destino e a complexidade algorítmica para gerar rotas disjuntas variam consoante o protocolo. Pode contudo dizer-se que pode ter efeitos positivos e efeitos negativos no tráfego de encaminhamento da rede [49]. P.ex., aumenta porque há necessidade de requisitar mais rotas mas diminui porque pode não ser necessário requisitar tantas rotas quando há quebra de ligações. Mais uma vez, a eficiência global de MP está intimamente dependente da implementação.

- *efeitos na capacidade*. Em parte, a mesma abordagem que foi feita ao tráfego de encaminhamento pode fazer-se aqui. A capacidade é um efeito global e indirecto. Demonstra-se [49] que, para efeitos de congestionamento (i.e., para efeitos do número de pacotes em espera na fila de cada nó: congestionamento se pacotes no sistema forem menores do que o tamanho das filas dos nós) e, consequentemente, apenas analisando a camada de rede, MP é eficiente se as rotas que se usarem seguirem, em média

$$L_m < \frac{\eta N_q}{(N_q+1)(\pi R^2 \delta - 1)\lambda}$$

Em que L_m: número de saltos médio das rotas, N_q: tamanho da fila de espera de cada nó, R: raio da rede ad hoc, δ: densidade de nós, λ: o tráfego gerado por cada nó, assumindo-se que são iguais; η: taxa de processamento do nó

Um argumento de pressupostos semelhantes (apenas consideram a camada de rede) é apresentado em [48], embora bastante menos optimista.

De uma forma geral, incluindo as poucas e indefinidas vantagens de MP (como a eventual redução do tráfego de encaminhamento e a disponibilidade de várias rotas em simultâneo subindo, eventualmente, a taxa de entrega de pacotes), as muitas e muito justificadas desvantagens e englobando efeitos da camada de ligação e de rede, salvo em casos muito específicos, MP parece uma técnica que deve ser encarada com muito cepticismo e precaução.

A Fig. 16 mostra resultados [56] para duas versões do DSR: caminho-único (usual) e multi-rota. O DSR multi-rota gera rotas disjuntas. É curioso que embora a fracção de pacotes entregues tenha subido com MP, o atraso aumentou mesmo quando, sistematicamente, o tráfego de encaminhamento é inferior à versão de caminho único do DSR. A entrega de pacotes pode não representar vantagens efectivas já que, se houver troca severa da ordem dos pacotes, a origem terá de os reenviar ou desistir de transmitir – o que não contribui para o débito-entre-extremos nem, em última análise, para a entrega de pacotes .

De referir igualmente que os autores não modelaram nem o MAC nem o canal apesar de referirem que, se o tivessem feito, poderiam ter obtido melhores resultados. Por um lado, é uma certeza, já que o atraso iria subir para ambos os protocolos e o peso do tráfego de encaminhamento (menor para o MP) iria beneficiar a versão MP. Mas, por outro lado, é de considerar que, com MP, o atraso global poderia aumentar. Primeiro, há menor tráfego

de encaminhamento (verificado), mas vai haver mais pacotes RERR (MP usa mais rotas e mais longas, a probabilidade de quebrarem nalgum ponto e com mais frequência é maior). Em segundo, como referido, os problemas MP que se relacionam com o MAC serão dominantes na maioria dos cenários.

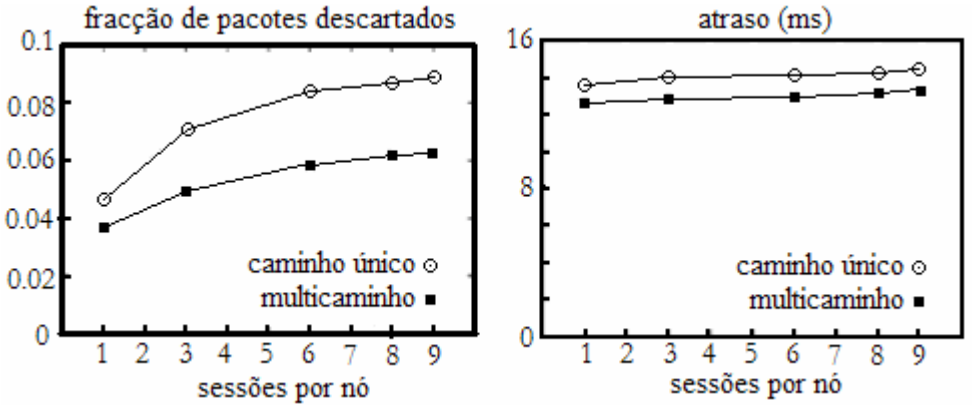


Fig. 16. C*omparação entre SP e MP usando derivados do DSR [56].*

3 Qualidade-de-Serviço.

Neste capítulo, o problema da Qualidade-de-Serviço (QdS) é discutido de um ponto de vista de estado-da-arte. Começa-se por apresentar soluções para redes com fios como o IntServ e o DiffServ. Mostra-se que não são adequadas a redes ad hoc. No final, as soluções existentes para redes ad hoc são analisadas e discutidas. O propósito é capturar lições relevantes para o problema central desta dissertação, a formular e desenvolver no capítulo seguinte.

3.1. Colocação do problema.

3.1.1. Introdução.

Qualidade-de-Serviço (QdS) é um conceito que só pode ser entendido acompanhando a história das redes de comunicação. De outra forma permanece algo nebuloso e com definições muito dependentes de contextos específicos[1]. A definição do ITU-T (E.800) é a seguinte: *[QoS is] The collective effect of service performance which determines the degree of satisfaction of a user of the service.* Adoptou-se esta definição por ser considerada muito satisfatória na medida em que inclui o que se considera ser, nesta dissertação, as componentes de QdS: o utilizador, a percepção de qualidade, o serviço, o desempenho e a cooperação das várias entidades da rede.

QdS, em última análise, é a resposta que a rede dá às solicitações de um cliente. Se uma aplicação necessitar de um atraso limitado (acima e abaixo) e de um mínimo de largura de banda, os mecanismos que implementam QdS devem fornecer, de forma transparente à aplicação, essas garantias. Claramente, QdS está intimamente relacionado com distribuição e alocação de recursos da rede. Ao afirmar-se que é um conceito melhor compreendido de

[1] Veja-se algumas definições em `http://www.cs.virginia.edu/~cs851-2/qosdefinitions.html` (acedido em Agosto de 2006).

um ponto de vista histórico, a ênfase é colocada na evolução das redes de comunicação. Ao princípio, as redes serviam essencialmente para transportar voz e uma "chamada" correspondia ao estabelecimento mecânico de um circuito físico entre quem chama e quem é chamado. Hoje já não é assim, mas a filosofia mantém-se: usam-se circuitos físicos com capacidade para várias chamadas e cada chamada corresponde a uma fracção da capacidade do circuito alocada enquanto dura a chamada entre os dois terminais. A partir do momento em que a chamada é estabelecida, a QdS é "perfeita" (na falta de melhor palavra): a largura de banda do canal está previamente definida como suficiente para a aplicação (e.g. 64kbps para chamada de voz sem compressão nas redes com fios), o atraso está perfeitamente limitado (praticamente é só o tempo de comutação nos diversos equipamentos do sinal), a variância do atraso é praticamente nula (não há filas nos nós de comutação), etc. Este tipo de rede, comutação de circuitos, comporta utilizações das ligações necessariamente abaixo do máximo: quer o circuito esteja a ser usado ou não, a ligação está afecta a apenas uma aplicação. Uma métrica muito utilizada em redes de voz é o factor de activação de voz e é bem conhecido que numa aplicação de voz comum um valor típico é 50%, correspondendo, p.ex., ao tempo da chamada dividida pelos dois extremos (raramente os dois interlocutores falam em simultâneo[1]). Ou seja, um canal de 64 kbps pode, neste cenário, transportar perfeitamente 2 chamadas, mesmo sem qualquer tipo de compressão. Há sub-utilização de recursos na comutação de circuitos.

Para aumentar a utilização dos recursos, propôs-se usar comutação de pacotes. As ligações são comuns e cada trama de dados que saia de um utilizador deve levar consigo um cabeçalho para efeitos de, pelo menos, identificação do destino. Este tipo de multiplexagem – estatística – tem claramente vantagens em termos de escalabilidade ao mesmo tempo que permite que os recursos sejam utilizados a níveis mais elevados. Contudo, para se manter a mesma qualidade da comutação de circuitos, deve agora planear-se a rede de forma diferente. Do ponto de vista de um operador, e dado que é virtualmente impossível criar infrastruturas que satisfaçam o cenário de todos os utilizadores usarem, em simultâneo, o máximo de recursos que lhes é permitido, surge agora a hipótese de ainda haver canal disponível mas estar a ser utilizado acima de um ponto máximo inicialmente previsto devido à variância do padrão de utilização. Se, em comutação de circuitos, o problema se resumia à probabilidade de bloqueio (obteve

[1] O GSM nem sequer prevê comunicação em *full-duplex* durante uma chamada de voz.

circuito ou não), agora há sempre "circuito" mas é partilhado e seu desempenho instantâneo não é garantido para todos os utilizadores subscritos. Sendo partilhado, as métricas acima indicadas (débito-entre-extremos, atraso, variância do atraso, ...) deixaram de ter valores mínimos e/ou máximos garantidos. Passamos de desempenhos determinísticos para desempenhos estocásticos. Deste ponto de vista, QdS é o resultado das técnicas que devolvem determinismo a uma rede de comutação de pacotes – nem que seja a indicação ao cliente de que a rede não pode estritamente satisfazer-lhe as necessidades[1].

Outra forma de olhar QdS é como a resposta aos recursos finitos de uma rede sobre-solicitada com a necessidade adicional de distinguir utilizadores (em sentido lato) de rede. Deste ponto de vista, QdS é um mero problema de capacidade. Com efeito, a sobreprovisão ("overprovisioning") é uma técnica de apoio a QdS – e a mais simples. Não é aplicável, simplesmente porque a necessidade de recursos pode aumentar mais rápido do que o reforço das infrastruturas. Entramos já numa questão de negócio – uma das componentes fundamentais de QdS, do ponto de vista deste trabalho. Será recorrentemente referido que um bom modelo de QdS tem de fornecer mecanismos de suporte a um modelo de negócio.

Numa classificação simplista, podemos considerar 3 grandes famílias de QdS:

(i) comutação de circuitos. A forma mais imediata de se implementar QdS é tentar emular circuitos (virtuais) sobre uma rede de pacotes. Para isso é necessário sinalização (tipicamente, criação_do_canal → manutenção → destruição_do_canal) e, adicionalmente, solicitar os recursos necessários ao longo do caminho. Do ponto de vista de cada nó, tem de haver mecanismos de admissão, controlo, reserva e diferenciação (ao nível de fluxo IP). É esta a essência do IntServ, usando o RSVP como protocolo de sinalização. No fundo é arquitecturalmente simples, claro, objectivo e tendencialmente determinístico (pelo menos enquanto há conectividade o que em redes com fios não é tão crítico como em redes ad hoc). Para além disso, tem outra grande virtude: na essência, é distribuído e não necessita de nenhum tipo de entidade central administradora. É

[1] Por vezes fala-se em Grau-de-Serviço (GdS) que representa a probabilidade de obtenção de serviço. É um conceito diferente de QdS.

um modelo genuinamente de rede, envolvendo *apenas* a rede (e as respectivas aplicações que solicitam o serviço).

(ii) orientada ao operador. Para combater o peso da máquina IntServ/RSVP, a discutir adiante, relaxou-se os requisitos e re-pensou-se QdS. Perante esta realidade, ao admitir-se que QdS é um problema de gestão dos recursos finitos de uma rede, e ao tornar consciente o termo "qualidade" (inerentemente associa percepção objectiva ou subjectiva, e.g., resp., atraso numa conversa de voz e qualidade da imagem dum vídeo) destacam-se os 3 elementos da visão DiffServ de QdS:

- a rede, na personalidade do operador ou gestor da rede privada
- o utilizador e as suas solicitações
- simplicidade

Neste sentido, o DiffServ responde ao problema com sinalização implícita e muito reduzida (pode ser simplesmente os pontos DS (DSCP) no pacote IP), não implica envolvimento de toda a rede e ao nível de cada fluxo (questões de escalabilidade), compromete o utilizador a um perfil (SLA) forçando a deslocação da utilização da rede de quem não exige para quem exige e a rede é extremamente simplificada (numa primeira aproximação) ao não envolver toda a rede mas só um certo conjunto de nós (as fronteiras). Continua a haver, como não poderia deixar de ser, controlo de tráfego (admissão, medição, escalonamento, descarte, prioritização, ...) mas esse controlo sai fora dos nós-utilizadores e implementa-se apenas em pontos estratégicos da rede sob orientação de uma entidade administrativa (o "ISP").

Pode questionar-se a elegância de envolver o utilizador no próprio modelo de serviço. O utilizador, até certo ponto, passa a ser uma entidade objectiva e activa da rede, mais do que a fonte de requisitos. Mas QdS, e tocando de novo na componente de negócio, ultrapassa a Técnica em si.

Uma implementação DiffServ é obviamente bastante mais operável e implementável pelo menos até se perceber que, o que se ganhou em simplicidade de sinalização em-banda ("in-band"), perdeu-se em sinalização de administração.

A forma de diferenciar o tráfego é ao nível do pacote IP (bits DSCP), que tem de ser confrontado com um certo SLA o que implica sinalização tipo AAAC[1]. Eventualmente, um esquema misto pode ser útil.

Uma outra forma de se distinguir IntServ de DiffServ é a seguinte: o primeiro produz sempre bons resultados porque não necessita de conhecer a rede onde está inserido; o segundo, *obrigatoriamente*, envolve o operador que tem de planear a sua rede para que, para um determinado número de utilizadores, cada um com um SLA, a capacidade da rede seja suficiente para entregar o contratado. É radicalmente diferente. A componente de planeamento da rede é crítica no DiffServ mas é a priori irrelevante para o IntServ. O DiffServ, de outro ponto de vista, desloca complexidade do plano de rede para o plano administrativo. O revés desta opção é que, deixando de envolver toda a rede (ou a parte dela relevante para a aplicação que solicita QdS), DiffServ, tipicamente, não consegue dar garantias absolutas. A Fig. 17 mostra de forma simples estes equilíbrios.

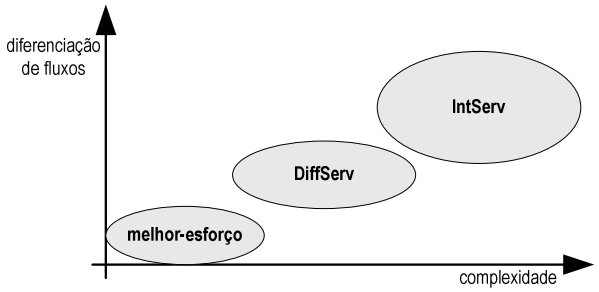


Fig. 17. *DiffServ vs IntServ.*

(iii) infraestrutural. Nesta família encaixam-se soluções de QdS intrínsecas à própria tecnologia de transporte (e.g. MPLS, ATM, SDH, 802.1p). Por si mesmas, tentam controlar a imprevisibilidade de uma rede de pacotes actuando na própria engenharia da rede (e.g. tamanho fixo/variável dos pacotes, encaminhamento/encaminhamento, ...) e/ou impondo uma camada de rede adicional. O ATM e o SDH, p.ex., têm mecanismos muito ricos de gestão da rede, nomeadamente, mecanismos internos de controlo e medição e uma granularidade

[1] AAAC: Authentication, Authorization, Accounting and Charging. Sumariamente, são as funções de conta de utilizador despoletadas pela identificação do mesmo perante a rede.

de intervenção muito fina. Adicionalmente, possuem mecanismos de tolerância a falhas que, dum certo ponto de vista, também se pode considerar suporte a QdS. O MPLS age por via de engenharia de tráfego[1] (TE) e o seu propósito é exactamente criar uma rede virtual com capacidade de QdS sobre a rede melhor-esforço. Outra característica importante do MPLS é tentar corrigir o excesso de "paquetização" do IP e aproxima-lo um pouco à comutação de circuitos como é o caso do ATM (células pequenas, de tamanho fixo e síncronas). Adicionalmente, o tamanho fixo dos pacotes tornaria as "switch fabrics" dos encaminhadores muito mais eficientes e capazes[2]. Finalmente, e muito mais importante, de um ponto de vista de redes de acesso heterogéneas, o MPLS é uma tecnologia que permite simplicidade de interconexão mantendo funcionalidades ricas de gestão e operação, muito à semelhança do carácter do IP mas para a camada de rede. De qualquer forma, são estratégias pensadas para as redes com fios pelo que não serão discutidas neste documento.

Finalmente, é necessário considerar o que é uma rede IP nos dias de hoje e o que será no futuro. Hoje as redes (no fundo, como sempre foram) caminham para a heterogeneidade pelo que qualquer modelo de QdS tem de ser suficientemente flexível para permitir interoperabilidade entre os diferentes troços. QdS exige necessariamente uma resposta extremo-a-extremo e, se apenas se considerar os troços puramente IP, os restantes troços geridos por outras tecnologias (como ATM/AAL5, porventura os mais decisivos), se não conseguirem interpretar a sinalização IP, anularão totalmente qualquer esforço para fornecer QdS. Além do mais, um modelo de QdS realista não pode sugerir modificações ao serviço BE mas simplesmente assumi-lo com mais um serviço.

Nesta dissertação serão considerados o IntServ e DiffServ por, essencialmente, um motivo principal: são universalmente considerados como referências para QdS em redes IP – seja porque absorveram experiências anteriores ou porque resultaram de um consenso. Para uma descrição pormenorizada de vários projectos, ver [61].

[1] Há sempre duas soluções para QdS: projectar a rede para excesso de carga ("overprovisioning") e TE. A primeira é colocar capacidade onde está o tráfego; a segunda é colocar o tráfego onde está a capacidade.
[2] Este argumento cada vez tem menos validade e espera-se que se torne irrelevante no futuro.

Finalmente, e por uma questão de rigor, o sumário que foi feito dos dois modelos pode levar a pensar que o IntServ e o DiffServ são duas soluções para o mesmo problema. Embora historicamente essa consideração seja válida, hoje cada vez mais se caminha no sentido de os tornar modelos *complementares*.

3.1.2. Principais modelos implantados.

IntServ / RSVP.

O IntServ consiste de três componentes: a arquitectura de rede, a sinalização e a gestão de tráfego em cada nó. A arquitectura da rede está representada na Fig. 18. A tracejado está o tráfego de sinalização.

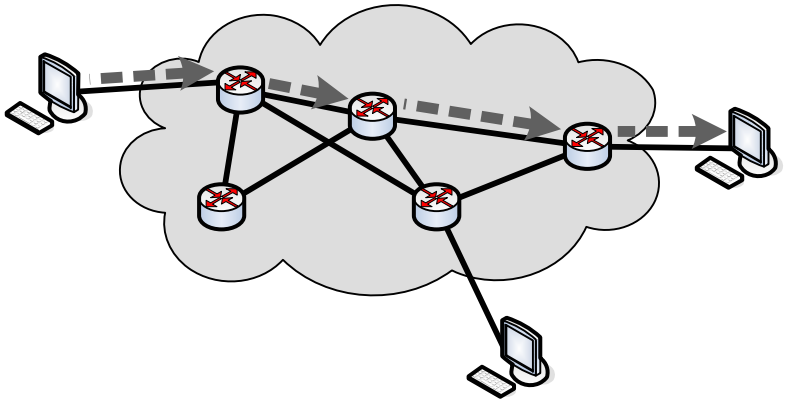


Fig. 18. *Arquitectura IntServ.*

O objectivo central no IntServ é definir mecanismos para a criação de canais virtuais com as propriedades desejadas que transporte um fluxo. Um fluxo é a granularidade mínima do IntServ e é uma sequência de pacotes originados por uma única aplicação e num único sentido (abaixo disto, só indo ao nível do pacote). Todo o caminho e todos os encaminhadores que os pacotes atravessam são envolvidos, incluindo as origens e os destinos, usando sinalização específica. A Fig. 19 mostra o modelo de referência para um nó que implemente IntServ: a parte superior do modelo relaciona-se com o processo de gestão do canal em si (estabelecimento, manutenção e terminação); a parte inferior com os

mecanismos de gestão dos fluxos durante a existência do canal, ou seja, manipula os pacotes em si.

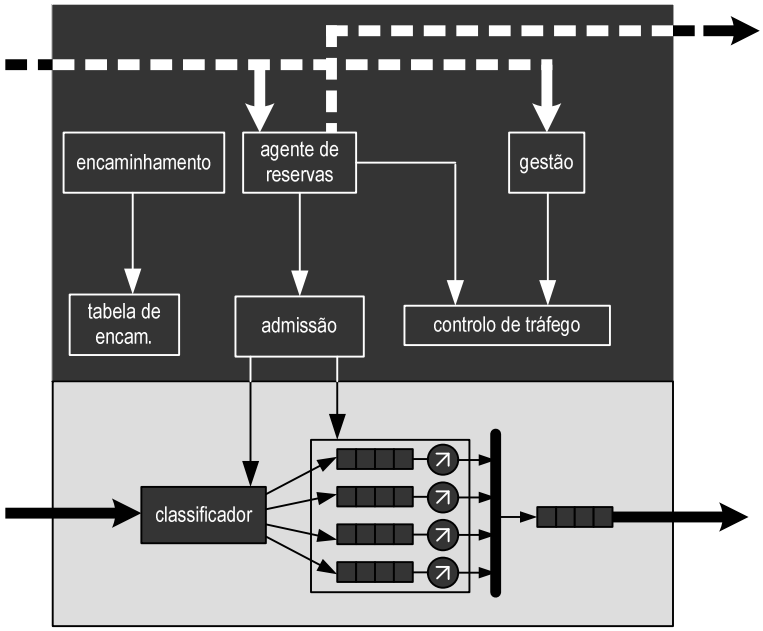


Fig. 19. *Modelo de referência para um nó que implemente IntServ (adaptado de [62]). A tracejado, o tráfego de sinalização; a contínuo, os fluxos de dados.*

As componentes são:

- *sinalização*. O IntServ, como existe actualmente [62], define um protocolo de sinalização, RSVP [63] [64], que actua em três fases: reserva, manutenção e terminação. O canal é desfeito se não houver um refrescamento activo. O RSVP não é mandatório mas, dado que a existência de vários protocolos de sinalização comprometeria seriamente a interoperabilidade das redes, é recomendado e universalmente utilizado.

- *admissão*. Cada nó verifica se tem recursos para aceitar o pedido, podendo negar. Este módulo também pode tratar de questões de AAAC.

- *classificador*. O tráfego que chega ao nó é analisado para tentar fazer corresponder a sua assinatura[1] a uma reserva anterior em memória. Se sim, coloca-o na fila da classe associada (incluindo a classe melhor-esforço). Esta assinatura pode ser

[1] Por assinatura entenda-se a a propriedade que permite distinguir pacotes ou grupos de pacotes com base nos seus campos.

transversal a todas camadas de rede – frequentemente, o túplo (end_origem; end_destino; id_fluxo). No IPv4 podem ser usados os portos TCP.

- *escalonador*. É o módulo que faz a gestão do tráfego e que, em última análise policia o tráfego em relação à especificação do fluxo propagada pelo protocolo de sinalização e aceite pelo módulo de admissão. O algoritmo de escalonamento determina a ordem pela qual os pacotes são colocados na fila de saída. Este módulo pode incluir um estimador (cálculo de estatísticas de tráfego em tempo real) e um algoritmo de descarte (incluindo o descarte-da-cauda – "tail drop").

- *fila de saída*. Tem relevância como módulo distinto na medida em que pode solicitar serviços de controlo de tráfego à camada de ligação. Por exemplo, usar o 802.3 com 802.1p e marcar pacotes com prioridades.

Uma reserva IntServ/RSVP implica propagar o perfil de tráfego (especificação do fluxo) e a respectiva assinatura. A especificação do fluxo consiste na `Tspec` (parametrização do escalonador e policiamento para o classificador) e a `Rspec` (parâmetros globais da reserva). O processo em si de reserva depende do protocolo utilizado (tipo de dados) e foi feito um esforço para haver apenas duas iterações, um em cada sentido: pedido de reserva (construção de uma configuração global) → confirmação de reserva (parâmetros definitivos). O RSVP opera da seguinte forma:

1. O originador envia uma mensagem `PATH`, indicando os pares ({si};{ri}) – o RSVP suporta de raiz difusão de grupo ("multicast") – e a `Sender-Tspec`.
2. Cada encaminhador prepara a sua máquina de estados conforme a `Sender-Tspec` e actualiza (ou inclui), na mensagem `PATH`, uma `Adspec` indicando o estado geral da reserva até ao seu salto, opcionalmente.
3. A `Adspec` não é mandatória mas flexibiliza o processo de reserva. Por exemplo, se um salto assegurar um atraso muito abaixo do solicitado, pode haver um nó que aproveite a circunstância e viabilize a reserva por assegurar um atraso acima do desejado mas globalmente respeitando a `Sender-Tspec`. A `Adspec` inclui as indicações globais da disponibilidade do caminho, válidas até ao salto que a recebeu tais como mínimo atraso acumulado, mínima largura de banda, se algum nó não

suporta RSVP (basta haver um para poder impedir a reserva), etc. Depende do serviço solicitado.

4. Quando a especificação do fluxo chega ao destino[1], este verifica a validade do `PATH`, prepara uma nova especificação do fluxo e inicia a reserva propriamente dita, enviando uma mensagem `RESV` para o originador usando o caminho inverso ao `PATH`. A `RESV` contem a especificação do fluxo composta pela receiver-`Tspec` e `Rspec`. A primeira é essencialmente igual à `Tspec` do originador; a segunda é uma optimização dos parâmetros fundamentais (mínima largura de banda e máximo atraso[2]).

5. Em cada encaminhador, cada receiver-`Tspec` é submetida ao bloco de admissão e, se passar, é usada para parametrizar definitivamente a máquina de estados que vai suportar o fluxo. Se não passar, incluindo o encaminhador anterior não suportar RSVP (bit `GlobalBreakHit`), uma mensagem de erro é enviada para trás.

6. Se não houver erro, a reserva chega até aos originadores e o tráfego começa a fluir.

O IntServ permite que se definam livremente classes de serviço mas até hoje apenas duas se fixaram: Carga Controlada ("Controlled Load", CL) e Serviço Garantido ("Guaranteed Service", GS). GS [65] é o serviço que garante um desempenho da rede significativamente determinístico. Assegura largura de banda mínima, atraso máximo e perda de pacotes nula (desde que conformem com a `Tspec`). CL [66] define um serviço sem garantias rígidas: essencialmente manipulando prioridades, permite que a aplicação veja sempre uma rede com pouca carga mesmo que efectivamente não se verifique.

A plataforma IntServ/RSVP é, sem sombra de dúvida, uma abordagem completa ao problema e que, de um ponto de vista técnico, resolve o problema da falta de garantias que uma rede IP coloca. Distingue o modelo da implementação e a sinalização da especificação do serviço. Mas é reconhecido que o IntServ é um modelo pouco prático e apenas fácil de aplicar em redes simples e pequenas (e.g. não *core*).

[1] Como dito, o RSVP foi projectado para difusão de grupo. Para simplificar o exemplo, apenas se refere um par (s, d).
[2] O atraso é fornecido não directamente mas por via do "slack time" – atraso para a largura de banda considerada e admitindo um escalonador GPS (Generalized Processor Sharing).

DiffServ.

O DiffServ [67] é uma abordagem distinta ao problema de QdS numa rede IP. A motivação inicial é a seguinte:

- *centrado no utilizador*. Ao contrário do IntServ que se centra no fluxo, o DiffServ centra os mecanismos de controlo do tráfego no "serviço". Serviço é redefinido no contexto do DiffServ*: the overall treatment of a defined subset of a customer's traffic (…)*. Ou seja, o centro deixa de ser o tráfego em si para passar a ser o cliente, no sentido administrativo do termo.
- *heterogeneidade da rede de transporte*. As entidades que asseguram o controlo do tráfego da rede foram deslocadas para certos nós especiais na rede. Tem várias implicações:
  - QdS deixa de obrigar as respectivas aplicações a terem mecanismos especiais de transporte: desaparece a sinalização por salto e ao longo da rota;
  - os encaminhadores intermédios não necessitam de suportar nenhuma funcionalidade ligada a QdS assim como as aplicações em si não necessitam de chamar procedimentos especiais;
  - a rede de transporte passa a ser heterogénea de raiz – nós com funções dedicadas a QdS são explicitamente adicionados;
- *escalabilidade, incrementabilidade e interoperabilidade*. A interoperabilidade está significativamente garantida no DiffServ ao libertar a aplicação de negociar as condições do canal. A escalabilidade é assegurada pela simplicidade da rede *core*.

Os elementos de uma rede DiffServ são os seguintes (Fig. 20):

- *domínio DS, região DS*. É a área da rede limitada pelos pontos onde o tráfego cliente entra ou sai e onde se implementa DiffServ. Um domínio DS é a parte dessa rede sob administração de uma mesma entidade; a região DS é uma de domínios DS interligados.

- *SLA, TCA, SLS* (respectivamente, *Service Level Agreement*, *Traffic Conditioning Agreement*, *Service Level Specification*). Representam os acordos que regulam o tratamento do tráfego entre os clientes e o(s) operador(es) da rede. Um SLA constitui-se entre um cliente e o operador; o TCA é inferido do SLA e define as regras a aplicar ao tráfego; um SLS é o acordo técnico, com base em SLAs entre dois operadores numa região DS.

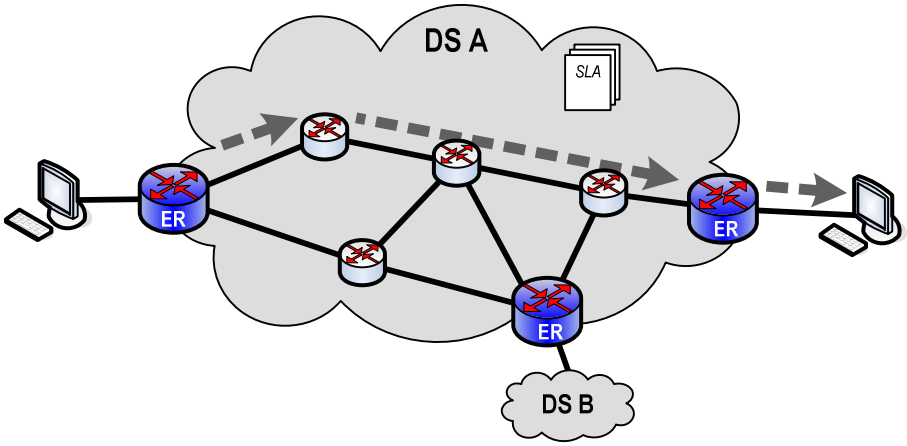


Fig. 20. *Rede DiffServ.*

- *encaminhador de fronteira* ("edge router"). Por oposição aos nós interiores (IR), o encaminhador de fronteira (ER) situa-se na fronteira do domínio DS. A Fig. 21 representa a arquitectura deste encaminhador especial. Considerando o ER mais à esquerda na Fig. 20 no sentido esquerda-direita (o DiffServ retém o 'simplex' do IntServ, ou seja, para os dois sentidos, há duas acções independentes):
  - Existe um acordo de tratamento do tráfego do cliente entre este e o administrador do domínio DS (SLA).
  - Cada pacote que entre no domínio DS é classificado e marcado pelo encaminhador ER (Fig. 21) correlacionando o SLA e a assinatura do pacote e re-preenchendo o DSCP (o conteúdo do campo DS [68] do pacote IP).

- A marcação é feita a dois níveis: segundo a classificação (função do classificador) e determinando se o tráfego do cliente particular está de acordo com o TCA (tamanho, largura do bloco de dados ("burst length"), etc. – função do medidor ("meter"). Se estiver fora do perfil, o medidor pode forçar o descarte do pacote.
- O DSCP já vem preenchido do cliente (marcação via classificação Comportamento do Agregado – "Behaviour Aggregate" ou BA) ou é inferido à custa da análise transversal (todas as camadas) do pacote de dados, à semelhança do IntServ/RSVP (marcação via classificação MF – "multi-field").
- Após formatação do fluxo à entrada do domínio DS (pode incluir descarte) o pacote entra definitivamente na rede de transporte.

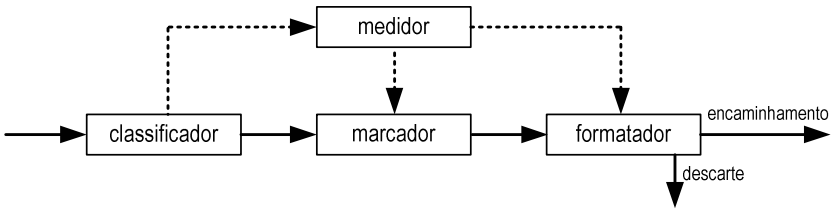


Fig. 21. *Encaminhador de fronteira no DiffServ.*

- *PHB.* Na rede de transporte, para além do encaminhamento normal há tratamento diferenciado para cada agregado DS (DS BA) – o conceito paralelo ao fluxo do IntServ.
  - Cada pacote do mesmo agregado tem o mesmo tratamento em cada encaminhador – acção-por-salto ("per-hop-behaviour" – PHB).
  - Cada IR tem de possuir mecanismos de regulação de tráfego como gestão de filas, algoritmos de descarte, escalonadores, etc. para conseguir diferenciar os agregados e providenciar a acção à saída da rede de acordo com o SLA para o agregado.

- À saída do domínio DS, o pacote vai para um dado cliente final ou segue para outro domínio DS. Entre os dois domínios DS (região DS) existe outro acordo (SLS) de tratamento de agregados.

- *sinalização*. O DiffServ não impõe nenhuma forma de sinalização mas torna-se claro que tem de haver comunicação entre a entidade na rede que gere o SLA e a configuração dos encaminhadores (classificadores e PHBs). Obviamente que há duas formas de se fazer: estática (configuração manual e nó-a-nó) ou dinâmica (via sinalização fora-de-banda). A componente dinâmica abre caminho a um novo tipo de entidades na rede, nomeadamente, os administradores-de-recursos ("bandwidth-brokers" – BB).

O DiffServ define dois PHBs gerais: o Encaminhamento Expedito ("Expedited Forwarding" – EF) e Encaminhamento Assegurado ("Assured Forwarding" – AF). O EF [69] comporta garantias mais rígidas e objectiva um serviço que garante (quase) assim que o fluxo inicia (tempo de transmissão de um pacote de tamanho máximo) uma taxa mínima através do domínio DS. Mais exactamente, cada ER policia o tráfego para que a taxa de entrada do agregado não ultrapasse a pré-determinada e para que cada IR aplique um PHB que assegura uma taxa mínima para esse agregado. O AF [70] tem garantias menos rígidas e compõe-se de várias classes (4 à data) divididas por 2 grupos: prioridade (precedência do agregado) e recursos reservados (classe do agregado). Cada IR serve primeiro o agregado com mais prioridade desde que não ultrapasse os recursos pré-alocados à classe do agregado (e haja competição entre vários agregados de classes diferentes). A ideia essencial é garantir que, a longo prazo, não haja perda de pacotes significativa (não há congestionamento) mas momentaneamente possa haver perda de pacotes (envio de "bursts" do agregado).

3.1.3. Discussão integrada.

Está fora do âmbito desta dissertação discutir em profundidade o tema de QdS em redes com fios. O interesse será apenas conhecer abordagens para redes com fios e tentar transportar a experiência adquirida para redes ad hoc.

Problemas conhecidos nas redes sem fios.

É bem conhecido que o IntServ tem problemas, todos virtualmente derivados das opções de arquitectura:

- *escalabilidade*. Cada encaminhador pode necessitar de manter informação sobre muitos fluxos (muitos nós, vários fluxos por nó). Por exemplo, cada fluxo pode ter a sua própria fila e o seu estado tem de ser actualizado periodicamente. Isto cria sérios problemas de escalabilidade.
- *modelo de negócio*. O IntServ não suporta convenientemente modelos de negócio, apesar de o RSVP poder ser usado com um protocolo de autenticação (um modelo de negócio em redes de comunicação envolve necessariamente, e em alguma fase, AAAC).
- *complexidade*. Tem duas faces. Primeiro, cada encaminhador necessita de complexos mecanismos de controlo de tráfego. Em segundo lugar, toda a rede, incluindo os utilizadores finais, tem de suportar IntServ – basta um encaminhador não suportar RSVP para as sólidas garantias CL ou GS se tornarem em garantias frágeis ou mesmo inexistentes.
- *interoperabilidade*. Dado que o RSVP é mandatório extremo-a-extremo, um fluxo ao atravessar vários domínios IntServ (cada um pertencente a diferentes administradores) pode ser tratado de forma diferente. Os vários parâmetros que cada encaminhador necessita implicam um acordo muito detalhado entre os vários administradores.

Finalmente, o próprio RSVP está algo ultrapassado, não suportando bem, p.ex., questões de mobilidade. Há iniciativas para definir novos esquemas de sinalização [71] [72].

A grande virtude do Intserv consiste no tipo de serviços que oferece que, no limite, permite realmente emular um circuito virtual extremo-a-extremo. Uma aplicação tipo VoIP poderia invocar o RSVP para construir o canal com GS durante o estabelecimento da chamada via SIP.

O DiffServ tem a virtude de deslocar a complexidade para a rede de transporte ("rede core"), criando nesta dois tipos de entidades (ER e IR). De um ponto de vista de negócio, a arquitectura é extremamente adequada aos operadores, já que tipicamente o tratamento dado aos pacotes é definido com poucos perfis permitindo até gestão por políticas. Além disto, é significativamente menos complexa do que a do IntServ. O revés é imediato: o tipo de garantias que pode oferecer não são tão rígidas como as que o IntServ pode oferecer. Pode, contudo, não ser um problema real nas redes actuais mas, academicamente, o problema existe.

Umas das aplicações que mais necessita de QdS é VoIP e basta uma rápida pesquisa no mercado para se perceber que os produtos comerciais de VoIP raramente tocam no tema[1]. A razão é que, mesmo à escala de uma rede local ("intranet"), onde o IntServ/RSVP poderia ser facilmente implementado, QdS é uma característica que não existe na prática.

Da parte dos operadores de WANs e redes metro, o problema parece ainda não se pôr nestes moldes. Até certo ponto, os operadores sempre usaram um esquema semelhante a DiffServ. Mas QdS para redes IP parece, paradoxalmente, pouco útil. Retome-se uma figura anteriormente usada para ilustrar conectividade entre redes heterogéneas e repetida na Fig. 22. Considerando que o troço 'ethernet' é uma rede de acesso xPON e que a rede do operador começa logo depois do Ponto de Acesso, a forma como o operador tipicamente actuaria para taxar o utilizador seria a seguinte. A partir de uma assinatura (p.ex., endereço L2 do equipamento na vizinhança do utilizador ou uma etiqueta VLAN) em cada pacote, o equipamento que faria o interface ethernet/SDH formataria o tráfego antes de entrar no anel SDH de acordo com o SLA aplicável. Tipicamente o pacote IP iria

[1] Tipicamente, incluem um guia de auditoria à rede. Ou seja, ainda que o problema seja reconhecido, a solução apresentada passa por saber se a rede pode suportar VoIP e não dotar a rede com funcionalidades que permitam suportar VoIP.

encapsulado num pacote da família 802.3, que seria encapsulado em GFP[1] (OAM e recuperação do contentor SDH) que, por sua vez, seria inserido num contentor SDH. No outro extremo da rede SDH, o ADM libertaria o pacote GFP, desencapsularia o pacote 'ethernet' e, deste, o pacote IP que encapsularia num pacote PPP sobre xDSL que, finalmente, entregaria ao utilizador final. Isto pode ser comparado com DiffServ, embora com pressupostos diferentes, nomeadamente, redes onde o IP apenas serve de ligação entre os utilizadores em cada extremo. Mas os aspectos essenciais mantêm-se: a rede de transporte do operador está planeada para a carga prevista com sobreprovisão (digamos que os PHB são para todos iguais e único) e cada pacote é marcado ou, pelo menos, analisada a sua proveniência, à entrada da rede.

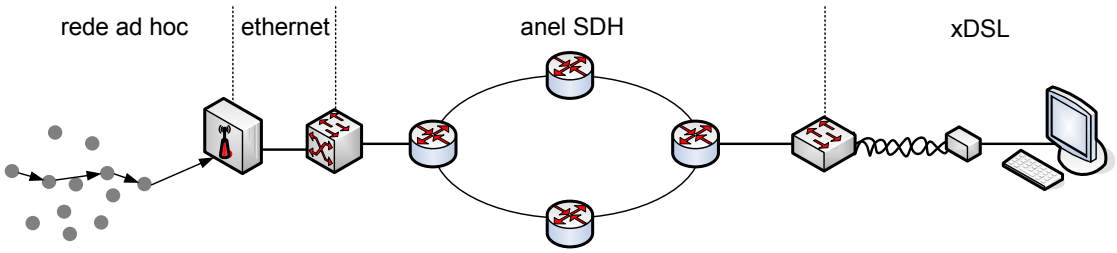


Fig. 22. *Rede heterogénea.*

A questão será agora se o utilizador quiser usar VoIP através das redes de vários operadores com garantias rígidas. Ou seja, colocando o problema de outra forma, como conseguir conjugar as virtudes do IntServ e do DiffServ. A solução pode passar por um modelo IntServ-sobre-DiffServ [73] em que o RSVP, p.ex., é usado para sinalização externa aos domínios DS e quem formata o tráfego (em sentido lato) na rede de transporte é o DiffServ.

Aplicabilidade a redes ad hoc.

Em primeiro lugar, é necessário notar o seguinte ponto não trivial. Em redes ad hoc, a conectividade de rede está muito dependente de factores externos ao nó (como a densidade e o número de nós). Assim, algo que não se leva em conta nas redes com fios pode bloquear de imediato qualquer tentativa directa de implementação destes modelos

[1] GFP: General Framing Procedure; é uma tecnologia que dota tecnologias L2 de funcionalidades de administração (e.g. detecção de falha de ligação remota) e de delimitação de tramas de comprimento variável.

em redes ad hoc. Existe contudo um trabalho que tenta implementar DiffServ em redes ad hoc [74]. Em relação ao IntServ, obrigar todos os nós a implementarem toda a complexidade que o modelo exige seria muito pouco prático. Os problemas de escalabilidade ainda tornariam o problema mais grave embora talvez menos do que o normal dada a pequena dimensão esperada das redes ad hoc no futuro próximo.

O DiffServ torna-se algo atractivo para as redes ad hoc pela sua relativa simplicidade (sinalização no limite nula, PHB podem ser simples escalonadores baseados em prioridades, etc.) e por não impor tantas condições a todos os nós.

É frequente dizer-se que o IntServ não é adequado a redes ad hoc e o DiffServ já pode ser utilizado. Contudo, esta afirmação geral e omnipresente merece um comentário. Em termos de princípio, Intserv *é* adequado às redes ad hoc e o DiffServ *não* é. Para compreender isto, é necessário ter em conta a essência básica das redes ad hoc: antes de mais, são redes de que se exigem serviços mínimos quando não administradas (i.e., *on-the-fly*). Dado que o princípio do DiffServ exige tanto uma fase de planeamento como, depois, um supervisionador (SLA/TCA, ER, PHB, etc.), só com esforço extra uma rede ad hoc pode ter DiffServ. Em termos práticos, e é essencialmente o que se diz, é mais fácil marcar os pacotes com, p.ex., prioridades e cada nó usar esse campo para gerir uma fila diferente do que exigir que cada nó implemente o RSVP. Mas só por si, isso não é DiffServ, assim como o IntServ não se reduz ao RSVP e ao modelo de encaminhador sugerido. Aliás, de um certo ponto de vista, pode até ser mais fácil implementar algo parecido com IntServ (ex: encaminhamento de QdS + sinalização + 1 bit de prioridade) do que DiffServ onde a componente de planeamento e administração (SLA/PHB) implicam um plano administrativo difícil de implementar em redes tão "anárquicas" como podem ser as redes ad hocs.

Há outro aspecto não trivial que será discutido em mais detalhe à frente. Como não se espera que haja muito tráfego numa rede ad hoc (por muitas razões), não pode haver muitos fluxos. Logo, o problema essencial da escalabilidade do Intserv/RSVP pode estar controlado por inerência.

A segunda razão pela qual estes modelos não são directamente aplicáveis a redes ad hoc prende-se com a instabilidade da rede (alta taxa de quebra e criação de ligações). Esta característica exige ginástica de qualquer protocolo de sinalização e reserva. O problema não é muito diferente do encaminhamento em si mesmo mas pode imaginar-se que uma reserva RSVP, a ser efectuada com sucesso, pouco tempo duraria. O tempo que demoraria a efectuar a reserva pode ser da ordem de grandeza da duração média de tempo de vizinhança.

Finalmente, coloca-se o problema da colaboração dos nós. Ao ser claro que cada nó não ganha nada em encaminhar o tráfego de outros, pode ser necessário incluir formas de forçar cooperação. Especialmente no campo de QdS, cada reserva para tráfego alheio significa menos recursos para si próprio.

3.2. QdS em redes ad hoc.

3.2.1. Introdução.

Uma vez explicado que as soluções para redes com fios não podem funcionar, tal como existem, em redes ad hoc, importa definir novos modelos de QdS para redes ad hoc. O trabalho nesta área está algo disperso. Alguns autores propõem modificações ao MAC, outros tentam adaptar modelos das redes com fios e outros tentam pensar no problema de raiz, incluindo a experiência das redes com fios. Propostas exclusivamente dedicadas ao MAC não serão discutidas nesta dissertação.

Parece haver uma certa tendência para particionar o problema em 3 áreas: modelo, encaminhamento de QdS e sinalização. Propostas integradas de QdS para redes ad hoc serão discutidas no fim deste capítulo.

Esta secção está dividida em três partes. Primeiro, apresentam-se abordagens que não são exclusivamente encaminhamento de QdS. Depois, protocolos que apenas servem para

encaminhamento de QdS[1]. Finalmente, faz-se uma discussão comparada das várias soluções.

3.2.2. Modelos de QdS para redes ad hoc.

FQMM.

Xiao [75] [76] propõe o FQMM. É um modelo de QdS para redes ad hoc que mistura conceitos IntServ com conceitos DiffServ, tendo em conta as características de uma rede ad hoc. Ou, dito de outra forma, tenta integrar o DiffServ no IntServ (ou v.v.), enquanto se ajusta a arquitectura às especificidades das redes ad hoc.

Na Fig. 23 representa-se o modelo de nó no FQMM. A ideia central é englobar num único modelo os seguintes elementos (tomar como referência a Fig. 24):

- sinalização em-banda e reserva com granularidade de fluxo associado a encaminhamento de QdS;
- papeis de nós de ingresso ("ingress nodes"), nós de egresso ("egress nodes") e encaminhadores interiores ("interior routers") dinâmicos e de contexto;
- nós com gestão de fluxos e agregados (no sentido IntServ e DiffServ) e gestão de recursos locais;
- DSCP e classificação de pacotes na origem.

Os blocos novos no FQMM são os seguintes:

- *discriminador*. Faz a triagem do tráfego: recebido do MAC (segue para camadas de transporte), vindo das camadas de transporte (segue para o MAC e posterior transmissão na linha) ou relativo a QdS.

[1] Chamar-lhes modelos de QdS não seria rigoroso. Daí dividir o sub-capítulo em protocolos de encaminhamento de QdS e protocolos que não o são, sob pena de enfatizar demasiado o tema de encaminhamento QdS, o que não é propositado.

▪ *formatador de tráfego adaptativo*. Formata o tráfego para respeitar o serviço que se pediu. Considera-se adaptativo porque este módulo deve reagir às condições da rede, nomeadamente à inferência do estado da ligação do nó (bloco "estado da ligação").

▪ *classificador*. Para além da função DiffServ (mapear pacotes em classes de serviço via DSCP), distribui informação a módulos no plano de controlo como a detecção da qualidade da ligação e a gestão do protocolo de encaminhamento de QdS.

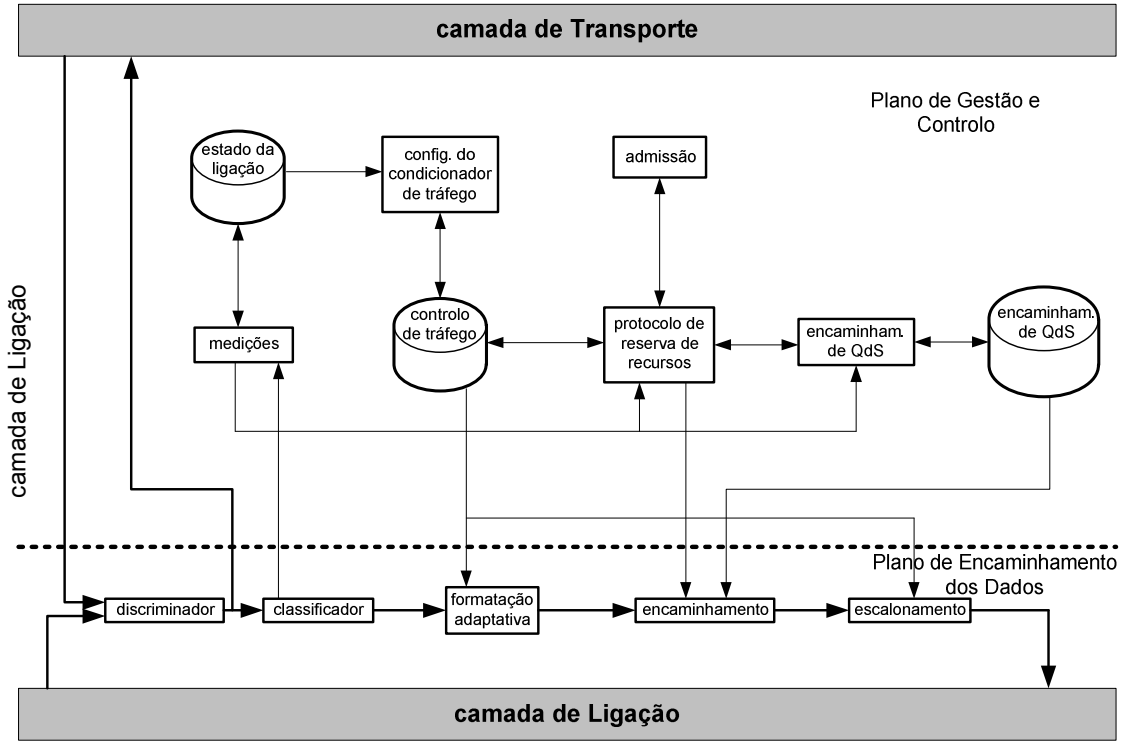


Fig. 23. *Modelo de referência de nó no FQMM [76].*

Talvez a maior dificuldade do IntServ é a reduzida escalabilidade do modelo. No FQMM considera-se que o problema é praticamente resolvido sabendo-se, primeiro, que a quantidade de nós/utilizadores é pequena e, segundo, a largura de banda será tão reduzida que o número de estados que cada nó terá de guardar e refrescar será sempre muito reduzido. É um pressuposto realista.

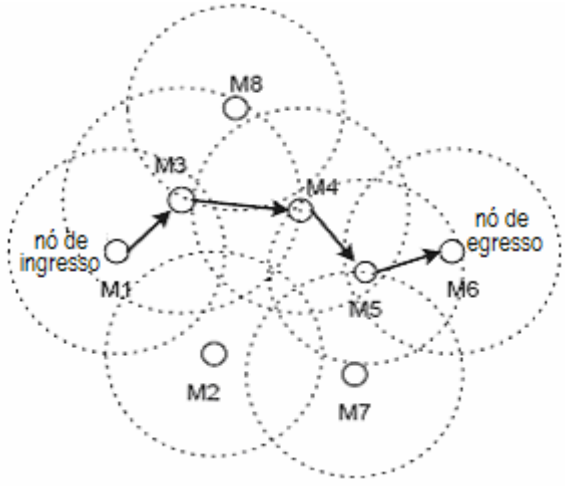


Fig. 24. P*apeis dos nós no FQMM [76].*

O tráfego é gerado por nós de ingresso e terminado por nós de egresso; nós intermédios, apenas encaminhando o tráfego, são nós interiores. Esta classificação dinâmica dos papeis de cada nó, dependente do contexto e local a cada nó, resolve a atribuição fixa de papeis no âmbito do DiffServ. Considera-se, em primeiro lugar que há dois tipos de tráfego: fluxos (IntServ) e classes (DiffServ). Para tráfego de alta importância, classificar-se-á como fluxo com a consequente activação de processos que impõem determinismo (sinalização de caminho, admissão, reserva, etc.). Para garantias mais suaves, usar-se-á agregados.

O modelo impõe a existência de um protocolo de encaminhamento de QdS para encontrar uma rota com as características desejadas para o fluxo ou agregado.

O modelo foi implementado numa rede muito simples (8 nós; 4 sessões TCP; token-bucket/RIO; DSR) obtendo os resultados da Fig. 25. A figura deve ler-se da seguinte forma. No cenário #0, nenhuma sessão tem tratamento preferencial; nos restantes, a sessão #i tem tratamento preferencial no cenário #i. Tratamento preferencial significa que a sessão tem direito a usar até 100% da capacidade da ligação; a capacidade restante não usada por essa sessão é dividida entre os restantes fluxos.

De notar, em primeiro lugar, que é clara a acção do modelo e o tratamento preferencial dado a cada sessão. Em segundo lugar, é necessário descontar o efeito do TCP (no cenário #3, a sessão #4 ainda teve mais utilização da rota do que no cenário #4). Finalmente, em termos absolutos, os resultados são modestos. Se a rede fosse muito maior, o efeito

diferenciador deveria esbater-se. As duas grandes lições a retirar e adiante reforçadas são as seguintes: (i) é perfeitamente possível usar-se um modelo que assente em diferenciação de tráfego em redes ad hoc; e (ii) para garantias minimamente aceitáveis, a solução terá de passar por algo mais.

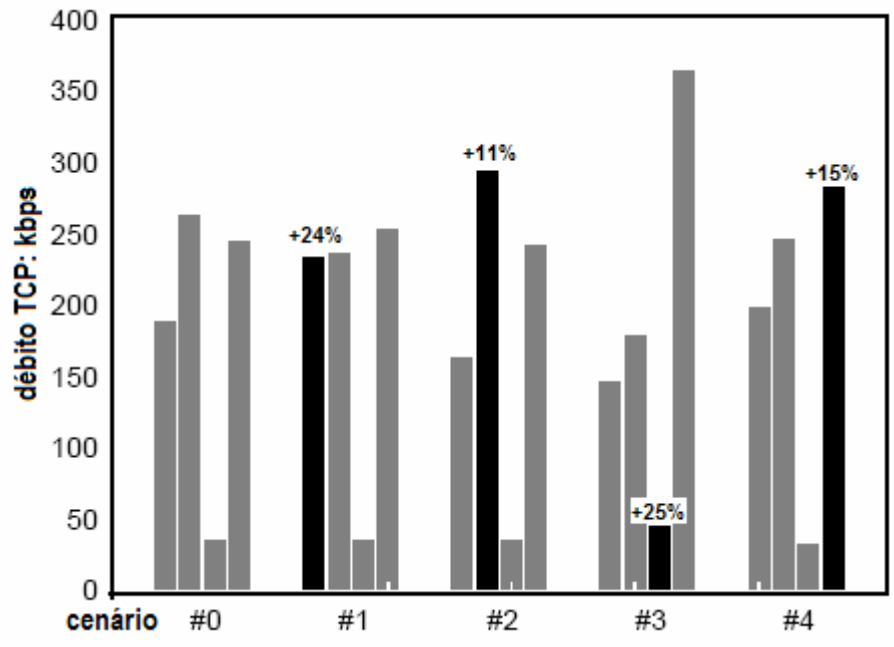


Fig. 25. *Simulação de FQMM [75].*

2LQoS.

Nikaein [77] parece usar o FQMM como ponto de partida mas usando premissas algo diferentes para elaborar o 2LQoS. Propõe igualmente um modelo de QdS por camadas (FQMM designa duas: controlo e encaminhamento) mas dá destaque à aplicação e métricas instantâneas da seguinte forma:

- a aplicação especifica métricas usando o seu próprio vocabulário
- uma camada intermédia mapeia essas métricas em métricas de camadas inferiores
- a aplicação tem de ser suficientemente adaptativa para conseguir operar com as métricas disponíveis

Define duas operações sobre rotas: geração e selecção. A essência do modelo é a seguinte (Fig. 26):

- invocar o algoritmo de encaminhamento e/ou de encaminhamento com base na para obter rotas
- rotas passam a ser objectos que incluem a especificação das suas propriedades em termos de métricas das camadas inferiores
- uma camada de interface traduz essas métricas para a aplicação
- a aplicação escolhe a rota de acordo com as métricas que acha pertinente

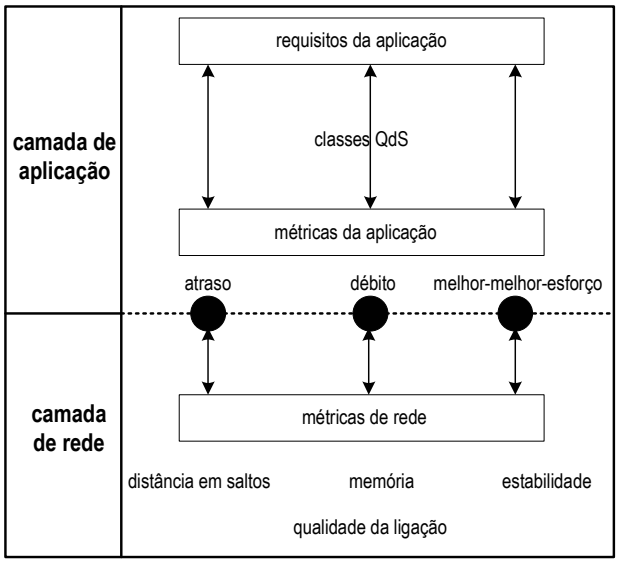


Fig. 26. 2*LQoS [77].*

Concretamente, os autores definem classes de serviço e dois tipos de métricas: de rede (NLM) e de aplicação (ALM). Dentro das métricas de rede, constróem as seguintes métricas: estabilidade (taxa de novos vizinhos), espaço livre na fila de saída e número de saltos. Cada nó faz propagar esta informação aos outros nós ou aos nós que possam estar interessados.

A camada de mapeamento NLM → ALM faz corresponder as métricas ALM às métricas NLM compostas a partir de 'número de saltos' ("hop count")', 'espaço disponível na fila' ("avaliable buffer") e 'estabilidade'. Concretamente, definem três classes de serviço embora seja claro que o modelo é significativamente mais flexível:

classe I: atraso → número de saltos + fila disponível
classe II: débito-entre-extremos → número de saltos + fila disponível
classe III: melhor-esforço melhorado → número de saltos + estabilidade

À saída de cada nó, cada pacote é marcado por DSCP o que servirá de apoio ao tratamento diferenciado dos pacotes em cada nó (prioritização, escalonamento, etc.).
Os autores apenas discutem o modelo não fornecendo nenhum tipo de implementação. A principal contribuição é fazer notar a necessidade de traduzir métricas entre camadas de rede diferentes e a participação de todas num modelo de cruzamento de camadas de QdS, avançando já com algumas ideias práticas.

DLite.

Gerhartz [78] tenta aplicar certos princípios do DiffServ. Não traz realmente nada de essencialmente novo, na medida em que apenas captura certas práticas do DiffServ e introduz a possibilidade de descarte de pacotes de tempo-real que vão atrasados e fora da especificação a meio da rota. Essencialmente, propõem filas e algoritmos de escalonamento e marcação de pacotes em cada nó. Cada nó também é responsável por não colocar excesso de tráfego na rede (o próprio nó efectua a própria "admissão" e formatação do fluxo). Para tal, é necessário um mecanismo de prevenção de congestionamento, ou uma qualquer métrica relacionada que esteja acessível ao nó. Os autores limitam-se a comentar o facto de ter de haver mas não fazem qualquer proposta.

SWAN.

SWAN [79] é uma proposta integrada que, essencialmente, se baseia num algoritmo de transporte com diferenciação de tráfego. Cria uma segunda camada de transporte em cima da existente, adopta uma série de técnicas e evita outras. Na Fig. 27 representa-se o modelo de nó. O principal objectivo do SWAN é evitar a todo o custo sobreutilização da rede, evitando activamente o congestionamento, para que pelo menos uma classe de tráfego (marcada com DSCP) tenha sempre serviço. No modelo, e não sendo uma mera decisão de projecto, como se verá à frente, definem-se duas classes de serviço: melhor-esforço ("BE") e tempo-real ("real-time" – RT). A ideia é, em nós intermédios, formatar o

serviço BE, evitar activamente congestionamento e dar prioridade ao tráfego RT. Se formatar uma classe e oferecer prioridades estritas a outra é trivial (do ponto de vista da técnica), evitar congestionamento de forma a que o atraso de uma classe RT esteja limitado e com variância do atraso pequeno já não é.

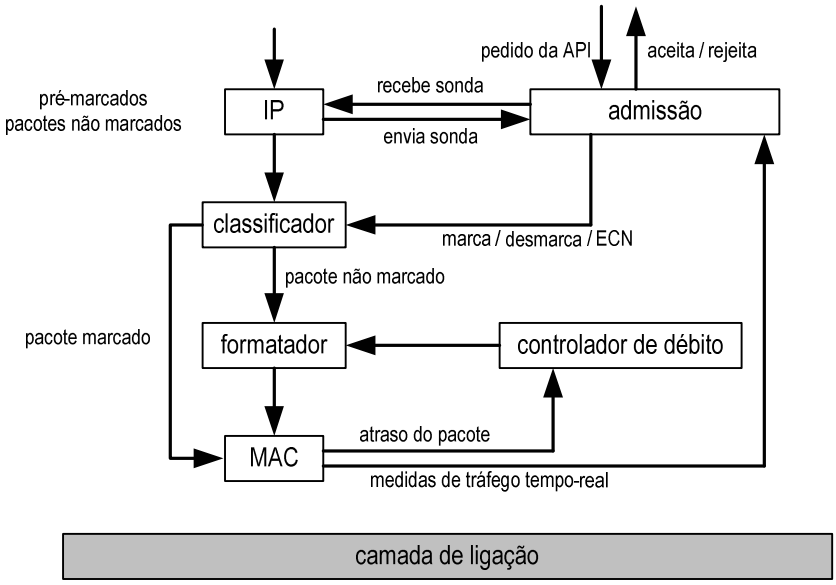


Fig. 27. *Modelo de nó do SWAN [80].*

Para tal os autores usam diversas técnicas:

- *bits ECN*. Os bits ECN[1] dos pacotes RT são marcados assim que o nó detecta as condições de congestionamento, tendo em conta o comportamento genérico de uma rede de pacotes (Fig. 28) em que o débito disponível tem quatro zonas (linear, saturação e congestão) e o atraso médio cresce rapidamente a partir da zona de saturação. O objectivo é manter sempre o nó abaixo do joelho da figura, colocando limites ao atraso. A gestão do congestionamento consiste em dotar o protocolo de transporte (essencialmente, TCP) de mecanismos adaptados às redes ad hoc.

[1] Explicit Congestion Notification

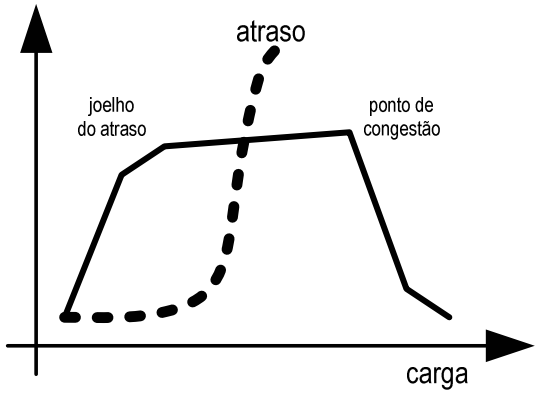


Fig. 28. *Comportamento genérico de uma rede de pacotes [80] utilizado pelo SWAN.*

- *suavização de blocos de tráfego ("bursts") RT.* o tráfego BE, usando filas com 'token bucket', amortece e suaviza o atraso do tráfego RT
- *sem reservas.* Há admissão mas não há reservas. É neste ponto que o modelo é sem-estados ("stateless"). Será discutido mais à frente.
- *sem sinalização persistente.* A excepção é feita às sondas iniciais para estimar a distância a que a rota está do ponto de congestionamento. No limite, apenas se faz uma vez por sessão RT; aumentam com a mobilidade da rede mas continuam a não ser tráfego periódico. Será discutido mais à frente.

O SWAN funciona da seguinte forma (referir à Fig. 27):

- Cada nó na rede, incluindo, naturalmente, os nós na rota do fluxo RT, vai regular o tráfego BE de forma a manter o tráfego local significativamente abaixo do limiar de congestionamento. O algoritmo do formatador de tráfego é do tipo AIMD[1] e o parâmetro de correcção é o atraso de cada pacote que sai. Tráfego RT contorna o formatador e é transmitido imediatamente.

[1] AIMD: Additive Increase Multiplicative Decrease

- Quando um nó tem tráfego RT, envia um pacote UDP para o destino (sonda) com um campo que preenche com a estimativa do débito-entre-extremos disponível que faz da sua própria ligação para o primeiro salto da rota.
- Essa estimativa baseia-se no tráfego RT que o nó ouve dos vizinhos directos via MAC. A estratégia do modelo em si é a de estimar a largura de banda disponível subtraindo a medida do meio a um valor previamente definido. Os autores referem, compreensivelmente, que não é a única forma de o fazer e que foi escolhida por ser simples. Este valor fixo e previamente escolhido deve ser conservativo (bem abaixo do joelho de congestionamento) para compensar admissões falaciosas (devido à dinâmica da rede). Há aqui uma certa componente de planeamento prévio de uma rede ad hoc que será recuperada e capitalizada no modelo que esta dissertação propõe (capítulo 4).
- A sonda corre a rota e cada salto actualiza o valor que veicula caso estime que a sua própria largura de banda disponível seja inferior ao valor actual.
- A resposta chega ao nó origem que compara a largura de banda disponível indicado pela sonda com a que necessita, efectuando ou não a admissão do tráfego RT.
- Se algum salto detectar que as condições da rota mudaram (ligação quebrada, p.ex.) e a rota está perto de congestionamento, notifica o destino (bits ECN) que, por sua vez, notifica a origem. O processo de admissão inicial é repetido.

O SWAN é, acima de tudo, um suporte à camada de transporte em redes adhoc. Pode ser visto como uma adaptação do TCP ao ambiente desfavorável das redes ad hoc. Desse ponto de vista, é difícil encara-lo como modelo de QdS. Discutir-se-á oportunamente o assunto.

Na Fig. 29 encontram-se resultados ilustrativos do excelente desempenho do SWAN ao nível do controlo do atraso. Estes resultados foram obtidos para 50 nós num espaço de 1500m x 300m, mobilidade variada e encaminhamento usando o AODV. Adicionalmente,

os autores referem que cada rota tinha entre 2 e 5 saltos (3 saltos em média). Portanto, as condições representam um cenário razoável.

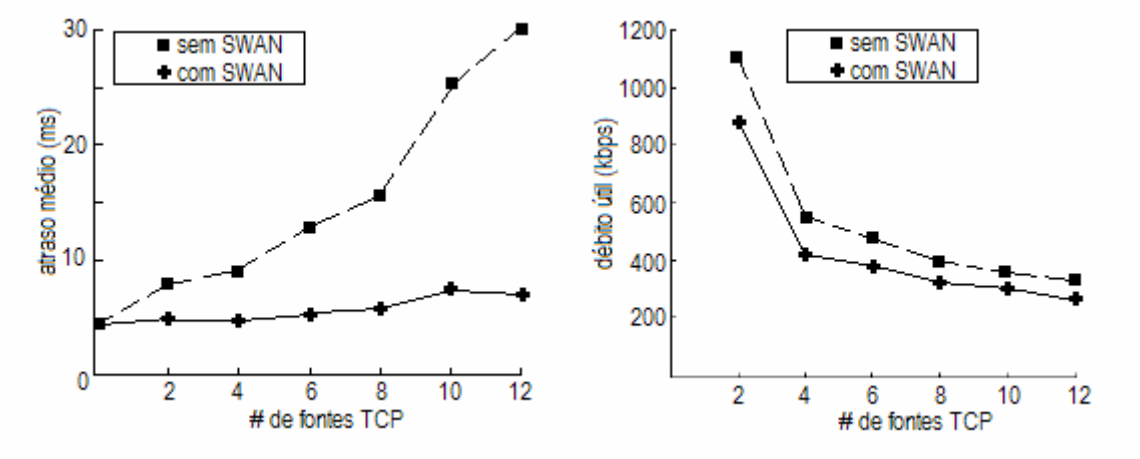


Fig. 29. Resultados do *SWAN [81].*

No primeiro gráfico, o efeito de prioritização do tráfego RT é notório e foram conseguidas garantias de atraso absolutas. Mas notar também que o débito-entre-extremos RT decresceu (segundo gráfico). A interpretação é imediata: para garantias rígidas de atraso, é necessário, antes de mais, regular o número de pacotes na rede. Para conseguir-se isto, primeiro é necessário controlar activamente (neste caso bits ECN) o tráfego que se injecta na rede. Baixou-se o débito-entre-extremos, colocou-se a rede numa zona confortável para o MAC e o atraso foi nitidamente controlado até quase ao limite da propagação do MAC&PHY (os 5 ms de atraso indicam claramente que praticamente não houve contenção de acesso ao meio). O controlo activo de congestionamento é essencial para estes resultados, algo que mais nenhum modelo até agora teve em conta. Pode até dizer-se que há dois passos essenciais no modelo: controlo de congestionamento → gestão de tráfego em cada nó.

É também claro que a aproximação ao controlo de tráfego é um mecanismo de força-bruta porque os pacotes RT têm prioridade absoluta sobre o tráfego BE. Mas notar também que neste modelo compete à origem regular o seu próprio tráfego para efeitos de admissão. Ou seja, é a própria origem que, em última análise policia o seu próprio tráfego.

Isto representa um conceito algo distinto em que ou são os nós na rota que o fazem ou são nós especiais como os ER no DiffServ. Este conceito de admissão na origem e *apenas* na origem (são dois efeitos em um) será utilizada no modelo que esta dissertação propõe.

INSIGNIA.

O INSIGNIA [82] pode ser considerado como uma tentativa de adaptar o IntServ às redes ad hoc, criando um modelo e um esquema de sinalização muito semelhantes. Na Fig. 30 representa-se o modelo de referência do nó. A reengenharia do IntServ situa-se nos seguintes pontos:

- *sinalização na-banda*. Ou seja, em vez de existir um protocolo específico para sinalização, os próprios dados transportam os requisitos de QdS. A diferenciação do serviço é ao nível do pacote.
- *admissão com base na estimativa de recursos disponíveis*. É centrado em largura de banda.
- *independência face ao encaminhamento*. Os autores fazem questão de separar sinalização, encaminhamento e reserva. É da responsabilidade do encaminhamento criar as rotas e o INSIGNIA funciona por cima do encaminhamento.

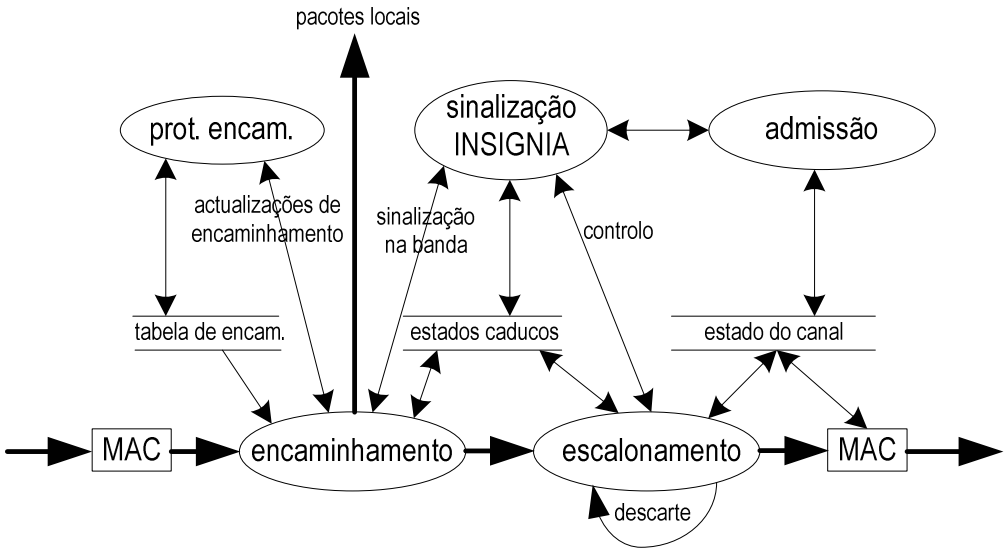


Fig. 30. *Modelo de referência de nó INSIGNIA [83]*

- *adaptatividade da aplicação*. O INSIGNIA exige alguma adaptatividade da aplicação ao exigir reservas escaladas em graus. A qualquer momento a reserva pode regredir para um nível inferior e a aplicação tem de suportar despromoções (nem que seja cancelando). Por outro lado, o INSIGNIA cria vários graus de reserva.

O INSIGNIA implementa a sinalização no campo das opções do pacote IP – Fig. 31.

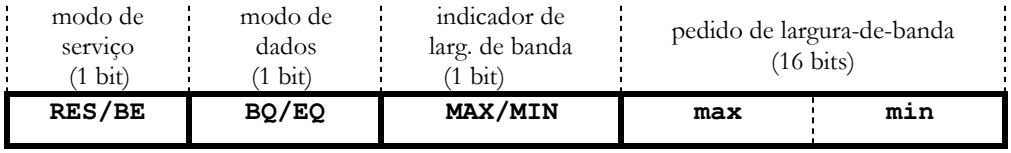


Fig. 31. *Campo INSIGNIA no pacote IP [83]*

A operação do INSIGNIA consiste nos seguintes passos:

- *melhor-esforço* ("best-effort" – BE). Se o pacote for BE não tem nenhum tratamento especial por parte dos nós intermédios.

- *reservas*. O pacote IP é transmitido com o bit `RES` solicitando o tipo de QdS (BQ/EQ). Se um nó intermédio receber um pacote destes e caso não tenha nenhuma reserva estabelecida, tenta admitir o novo fluxo consoante os recursos disponíveis e os restantes campos no pacote. A origem especifica o débito-entre-extremos que necessita (máxima e mínimo, só máximo, só mínimo).

- *relatório de QdS*. O destino envia a informação do estado da reserva ao nó origem. Os pacotes que podem chegar ao destino são:

[RES/-/MAX]	máxima largura de banda alocada ao longo de toda a rota
[RES/-/MIN]	pelo menos um nó não conseguiu reservar o máximo; a rota conseguiu alocar o mínimo
[BE/-/-]	reserva sem sucesso

Como as reservas são unidirecionais, a origem não tem forma de saber em que estado os pacotes estão a chegar ao destino. Portanto, o INSIGNIA usa relatórios de QdS para notificar e reportar alterações à origem.

▪ *alteração de serviço.* É a essência da flexibilidade do INSIGNIA. Uma reserva pode ser alterada (relatórios de QdS e autonomamente por cada nó intermédio) enquanto dura, principalmente pelos nós intermédios ao detectarem alteração no tráfego: `BQ←→EQ ; MAX←→MIN ; RES←→BE`. Notar que o campo "payload indicator" é específico da aplicação e pode servir para diferenciar fluxos da mesma aplicação[1].

O INSIGNIA consegue resultados interessantes [84] como ilustrado na Fig. 32 [38]. Para além de provar que funciona com protocolos diferentes, consegue efectivamente melhorar o atraso e o débito-entre-extremos de sessões TCP. Particularmente para o TCP, consegue suavizar a seu desempenho ao longo do tempo – Fig. 33.

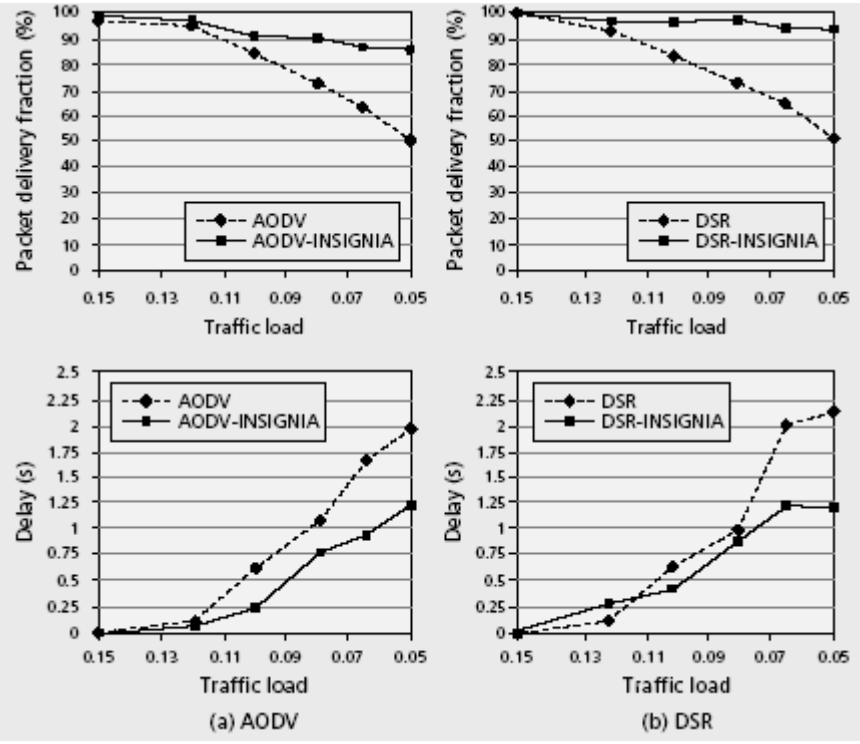


Fig. 32. *Desempenho do INSIGNIA [82]*

[1] Aliás, uma versão anterior do INSIGNIA incluia um terceiro campo que distinguia entre serviços RT e BE.

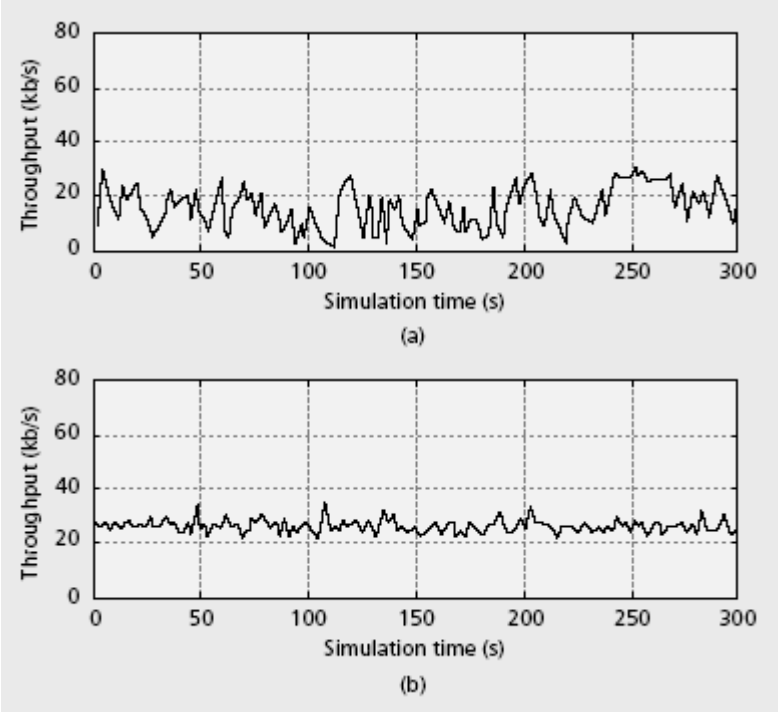


Fig. 33. *Efeito de suavização do INSIGNIA sobre o TCP [82]*

Contudo, notar que os benefícios podem não ser muito significativos, principalmente atendendo à complexidade que se introduz quer na rede quer em cada nó. Talvez o efeito mais interessante seja a persistência da taxa de entrega de pacotes. Curiosamente, sendo o INSIGNIA claramente orientado a largura-de-banda, conseguiu-se optimizar outra métrica. Por um lado, é QdS puro porque para certos pacotes serem entregues outros foram descartados e QdS pode sempre ser visto como deslocação inteligente de recursos entre fluxos que são diferenciados. Por outro lado, e mesmo havendo sempre relação entre as várias métricas, entrega de pacotes e largura de banda são algo independentes: para maior entrega de pacotes não são necessárias filas com mecanismos de controlo de largura-de-banda: bastaria para tal um simples esquema de prioritização em vez da complexidade do INSIGNIA. Por sua vez, e para o TCP em particular (Fig. 33), é notório o efeito de suavização, mas estes resultados devem ser comparados com outras experiências do TCP em meio IEEE802.11 e retiradas as conclusões devidas. Para se obter este efeito, muitas vezes basta reduzir a janela máxima do TCP. Estes efeitos foram já discutidos.

Finalmente, um resultado que está implícito é o seguinte. O INSIGNIA pode ser considerado uma versão com menor complexidade do IntServ. Até certo ponto, fica provado que o IntServ, desde que devidamente adaptado, pode funcionar sobre uma rede ad hoc. O princípio é executável e, quando muito, é a implementação do modelo que deve ser cuidadosa.

3.2.3. encaminhamento de QdS.

Encaminhamento de QdS é o mecanismo que encontra uma sequência de saltos com propriedades adequadas ao transporte dos pacotes. O INSIGNIA é também um protocolo de encaminhamento de QdS. Há sempre duas visões: (i) a de considerar que deve estar integrado no protocolo de encaminhamento base e (ii) que deve ser independente (e.g. INSIGNIA). Consideram-se nesta secção algumas propostas referentes à visão (i).

É frequente confundir-se encaminhamento de QdS com optimização, o que, se são conceitos que podem estar muito próximos, são diferentes do ponto de vista de princípio. Encaminhamento de QdS implica necessariamente um passo inicial de definição de necessidades do nó (obviamente, externo ao protocolo de encaminhamento) e o protocolo proposto apenas tenta buscar uma rota com métricas suficientes e não com as melhores métricas. Por outras palavras, encaminhamento de QdS é um procedimento de que resulta um conjunto de rotas que satisfazem QdS; optimização devolve rotas S-D orientadas e/ou ordenadas a/por determinada métrica. Esta distinção é fundamental para distinguir encaminhamento optimizado (eventualmente engenharia de tráfego) de encaminhamento de QdS. Uma consequência fundamental é a seguinte: enquanto a optimização é um processo que pode convergir sem resultar numa rota que satisfaça QdS, encaminhamento de QdS converge sempre porque devolve sempre o conjunto das rotas que satisfaz QdS (incluindo o conjunto vazio).

AODV com encaminhamento de QdS.

A literatura refere três iniciativas para dotar o AODV de mecanismos de encaminhamento de QdS [85] [86] [88]. Será apenas discutida a primeira e será designada por QAODV (actualmente vai na segunda versão).

A ideia da extensão é relativamente simples. Na altura do RREQ, o nó especifica as métricas que a rota deve satisfazer e apenas os nós que a satisfizerem, localmente ou cumulativamente (depende da métrica) participam na rota. As métricas pré-definidas (o protocolo parece dar flexibilidade) são a largura de banda (métrica local), o atraso (cada nó adiciona o atraso da ligação ao que vem no RREQ) e a variância do atraso (igualmente). Cada RREQ identifica a sessão de QdS com uma identificação (identificação da sessão – "session-id"). A tabela de encaminhamento também passa a associar a cada salto (na prática, a cada rota) métricas de QdS, incluindo uma lista de vizinhos que necessitam de garantias extra, à semelhança da lista de vizinhos para o encaminhamento de melhor-esforço.
Após o estabelecimento de uma rota QdS, o tráfego usa-a até terminar ou, se as métricas da rota se alterarem, cada nó que deixar de a satisfazer envia um QOS_LOST (ICMPv6) à origem, indicando a identificação da sessão associada, que terá de repetir o processo de procura de nova rota.

QOLSR.

Existe também uma proposta de encaminhamento de QdS para o OLSR [87]. No entanto, esta proposta dificilmente se pode considerar encaminhamento de QdS. A ideia é associar um custo a cada ligação e escolher o grupo MPR de cada nó com base nos custos. Por esta razão, QOLSR é mais uma optimização do protocolo 'link-state' original. Contudo, não parece ser impossível escolher não a melhor rota de todas mas uma que cumpra os requisitos. É nesta perspectiva que se prossegue a análise.

QOLSR tem como objectivo construir, para largura de banda, uma árvore de custos ("spanning tree") com custo mínimo entre todos os extremos – neste caso, custo é uma medida da largura de banda. Como o OLSR opera em volta do grupo MPR de cada nó, trata-se de escolher, entre os nós possíveis para serem MPR, os que oferecem as condições

exigidas. Os nós MPR são, para cada nó, o vértice da árvore e basta somar o custo vertical para se obter um grupo de MPR suficiente. A diferença entre esta versão a o OLSR original está na difusão dos vizinhos em que agora se inclui métricas para além das ligações directos.

CEDAR.

O CEDAR [88] é um protocolo de encaminhamento por si que incorpora encaminhamento de QdS. É algo complexo já que uma série de técnicas adicionais foram desenvolvidas para colmatar falhas quer do próprio algoritmo quer das redes adhoc (e.g. difusão fiável). É especialmente adequado a redes ad hoc muito grandes (os autores referem centenas de nós).

É proactivo e tem alguma similitude com o OLSR no controlo de inundação, como referido anteriormente: no OLSR constitui-se o grupo MPR e no CEDAR o MDS (o conjunto mínimo dos nós dominadores – "minimum dominating set"), embora com a diferença de o conjunto MPR ser diferente para cada nó e o MDS ser global à rede. O MDS é determinado a partir de um algoritmo proposto pelos autores para conseguir que cada nó o determine localmente. Adicionalmente, o MDS também determina as rotas para cada nó, em vez de serem os próprios nós a fazerem-no. Isto significa que uma rede virtual ("overlay") de encaminhamento é constituída por cima da rede física, que certos nós nunca participam em encaminhamento (quantos mais, melhor) e que a rede virtual pode utilizar outro protocolo de encaminhamento (e.g. DSR, AODV, TORA, …). Cada nó apenas mantém da parte a topologia da rede para chegar ao primeiro nó de transporte (uma espécie de gateway) e, tendo depois acesso à espinha dorsal da rede ("backbone"), a sequência de 'gateways', usa um protocolo de encaminhamento para criar uma rota S-D. Os nós de acesso à espinha dorsal designam-se por nós dominantes. Esta estratégia tem vantagens significativas ao nível da manutenção das rotas, do tráfego de encaminhamento exigido por inundação e em escalabilidade. Por outro lado, parece ser extremamente adequado para redes grandes.

Um aspecto interessante da proposta é a de anunciar a rede de transporte usando pacotes unicast entre vizinhos de um nó ao nível do MAC emulando difusão fiável.

A partir do momento em que a espinha dorsal está gerada e há conectividade extremo-a-extremo, há a questão de encaminhamento de QdS. Há dois passos. O primeiro consiste em propagar (usando a emulação de difusão referida) o estado e largura de banda de cada ligação em vários intervalos:

- cada nó mede pelos seus meios a largura de banda entre cada vizinho e reporta ao nó dominante esta informação; este propaga a informação pela espinha dorsal.
- cada vez que há alterações, é colocado na espinha dorsal para difusão um pacote que indica o aumento (pacote "ito") ou diminuição (pacote "dto") na largura de banda de uma dada ligação (e não a largura de banda em si). Este mecanismo é extendido a novas ligações ou ligações desfeitas, sendo um caso particular de incremento/decremento de largura de banda. Os autores referem a propagação desta informação como "ondas de incremento/decremento". Estes incrementos/decrementos estão quantizados.
- cada pacote de propagação contem um TTL; se é apenas a indicação de ligação-em-cima ou ligação-em-baixo, apenas pertinente para o nó dominante, TTL=0.
- cada mensagem tem um tratamento algo diferenciado consoante o seu tipo (ito/dto) e a largura de banda anunciada.
- os autores argumentam que o TTL da onda deve ser função monotonicamente crescente da largura de banda anunciada já que ligações mais robustas (estáveis/largas) devem ser mais utilizadas. Logo, a informação de ligações menos estáveis/largas deve ficar localizada a uma área pequena da rede para que poucos nós os usem;
- depois, cada nó dominante não propaga imediatamente o pacote mas coloca-o na fila e só envia a certos intervalos de tempo (gestão da difusão). Se uma onda for colocada para transmissão na fila do nó dominante quando outra espera transmissão, o nó dominante recalcula a variação global dos pacotes ainda não transmitidos e apenas transmite esta variação.

Em segundo lugar, há encaminhamento de QdS em si. Como dito, em cima da rede virtual qualquer protocolo de encaminhamento pode operar. O CEDAR propõe um protocolo por-pedido ("on-demand") para descobrir uma rota S-D entre o dominante de S e o de D (difusão de pedido de rota, encaminhamento-na-origem) ao longo da espinha dorsal; para descobrir uma rota que satisfaça os requisitos de largura de banda, opera de outra forma:

- descoberta de uma rota que satisfaça QdS entre o nó dominante de S e o nó dominante mais longínquo (nó T) da área de conhecimento da topologia do nó S (cada nó conhece uma parte da topologia), usando todos os nós, e não apenas a espinha dorsal. Esta rota é obtida pelo algoritmo de Dijkstra.
- T executa o mesmo processo e determina uma parte das rotas que satisfazem os requisitos.
- o processo prossegue até chegar ao destino ou até se determinar que não há nenhuma rota S-D satisfazendo QdS. Quando existe, as partes da rotas descobertas são reunidas e a rota extremo-a-extremo foi encontrada.
- a manutenção de rotas é baseado no recálculo apenas da zona que foi afectada.

Sondagem baseada em bilhetes.

Sondagem baseada em bilhetes ("Ticket-based probing") [89] consiste em enviar sondas por várias rotas S-D à procura de uma rota que satisfaça QdS. Definem dois algoritmos gerais consoante a necessidade: atraso ou largura de banda. Apoia-se nas seguintes técnicas:

- *inferência de métricas de ligações*. Cada nó caracteriza as suas ligações de acordo com a estabilidade (estacionário/transiente) e propriedades QdS. As métricas para QdS são o atraso, a largura de banda e o custo (parâmetro genérico configurável pela aplicação – pode ser o número de saltos). A primeira métrica baseia-se no tempo de duração da ligação (quando se forma é transiente, ao fim de algum tempo é promovido a estacionário).
- *gestão da imprecisão das métricas*. Para além de registarem e propagarem métricas de ligações, cada nó mantém e propaga uma estimativa de previsão da métrica.

- *sondas*. Cada nó faz enviar um dado número de pacotes de descoberta contendo as métricas desejadas. As sondas só seguem por ligações que satisfaçam QdS e cada nó é responsável por seleccionar as ligações por onde encaminhar cada sonda (baseado nas métricas de cada ligação e nas métricas de estabilidade) já que a origem limita o processo de inundação. A inundação é controlada incluindo em cada sonda uma quantidade limitada de autorizações de encaminhamento (bilhetes – "tickets").

Cada sonda chega ao destino com o valor actualizado da métrica considerada (atraso acumulado, mínimo de largura-de-banda ou operação sobre o custo). Quando todos os bilhetes chegarem (cada nó que não reencaminhar uma sonda notifica o destino), escolhe-se a melhor rota. As restantes são usadas em caso de a primeira falhar. Os autores não deram grande importância à forma como a origem é notificada das rotas encontradas; por simplicidade, escolheram encaminhamento-na-origem. O mais importante é que também se procedem a reservas de recursos ao longo da rota escolhida. Os autores não sugerem nenhum mecanismo mas é claro que assumem que cada nó tenham capacidades de gestão de tráfego e o tipo de mecanismo está obviamente dependente da condição de QdS desejada (atraso, largura-de-banda ou custo).

A grande vantagem dos bilhetes é limitar a inundação mantendo todas as funcionalidades[1] (as melhores rotas são obtidas e as métricas de qualidade incluem a estabilidade.

AQOR

Esta proposta [90] não representa nada de realmente novo no âmbito de QdS para redes ad hoc: rotas QdS por-pedido (indicando atraso máximo e largura-de-banda mínima), sinalização própria, inferência de largura-de-banda e atraso e reservas com admissão. É também um protocolo de encaminhamento por si e usam números de sequência para evitar ciclos (à semelhança do AODV). Para a inferência de recursos e consequente admissão de agregados, os autores propõem um cálculo linear que entra com o tráfego próprio (bem conhecido do nó) e o tráfego estimado por interferência. Esta parcela

[1] À altura do trabalho (1999), seria de considerar todas mas, no estado actual da investigação, é claro que muita experiência já foi considerada pelo que só se refere uma delas. Por exemplo, a questão de tornar o processo de descoberta de rotas totalmente distribuído é mandatório neste momento e já não se pode considerar uma vantagem no sentido de constituir uma opção ou tecnologia nova.

consiste essencialmente em majorar o tráfego de cada vizinho (indicado por pacotes `HELLO` e não inferido do MAC) e subtrai-lo à largura-de-banda disponível. O atraso é uma medição directa do RTT aquando do pedido de rotas.

3.2.4. Questões de colaboração.

As questões de colaboração podem ter um papel decisivo nas redes ad hoc. Desde logo se pode definir dois tipos de nós: maliciosos (desrespeitam os protocolos a correr e/ou deliberadamente atacam a rede) e egoístas (só usam a rede em acções que conduzam a benefícios para si). No primeiro caso, pelo que já foi dito, para redes baseadas em IEEE802.11, basta um nó ocupar indiscriminadamente o meio para impedir outros nós de comunicar. Sendo assim tão simples, este exemplo mostra que segurança é um conceito muito frágil – neste caso, é um problema do PHY+MAC e só aí se pode resolver. Os problemas de cooperação são os do segundo caso.

Efectivamente, é absolutamente legítimo (ou deve ser) de um ponto de vista de liberdade dos utilizadores, que alguém queira usar o seu equipamento apenas em situações que necessite. O caso mais claro talvez seja o de um utilizador a operar um equipamento que funcione a baterias. É sabido que uma placa IEEE802.11 gasta quase tanto só a ouvir o meio como a transmitir. Por isso, por questões de energia, pode querer apenas usar o seu equipamento quando necessitar dele.

Outra situação muito válida, e desta feita tocando perto o problema de QdS, é um utilizador rejeitar encaminhar tráfego de terceiros para ter mais recursos próprios (com efeito exponencial já que "limpa" o meio à sua volta). Parece-me ser igualmente legítimo alguém negar ser encaminhador com o argumento de salvaguardar recursos para si.

A cooperação é, por estas razões, um problema que pode ser decisivo numa rede real: se todos os nós, ou uma parte significativa, forem egoístas, a rede não pode funcionar devidamente. A visão do problema nesta dissertação é a de que o problema se resolve (quase) por si: a qualidade de uma rede ad hoc depende de si própria e se muitos utilizadores usarem a rede apenas quando tirarem proveito directo dela, deixam eles

próprios de a poderem usar porque, simplesmente, deixa de haver rede. Há, portanto, aqui três faixas de utilização: (i) (quase) todos cooperam; (ii) (quase) nenhum coopera; (iii) alguns não cooperam. Os primeiros casos estão resolvidos: um é desejável; o outro é necessariamente transitório porque força os utilizadores a entrarem no primeiro ou no último caso sob pena de não haver conectividade para (quase) ninguém. O 3º caso é mais complicado porque a rede funciona – simplesmente, com um desempenho sub-óptimo com alguns nós a não participar em prol dos restantes.

Mas, por uma questão de bom-senso, o 3º caso só é problemático por uma mera questão de "equidade social" e, necessariamente, pouco impacto deverá ter na funcionalidade da rede. Portanto, até certo ponto, o problema não é das redes ad hoc e sai completamente fora do âmbito da tecnologia. Mas, acima de tudo, é necessário não menosprezar o problema, até porque a tecnologia pode servir de apoio à engenharia social.

As soluções apresentadas até agora são algo frágeis na medida em que introduzem mecanismos e/ou serviços que, por si, são susceptíveis de fraude e/ou introduzem ainda mais tráfego não útil ("overhead") à rede, algo muito indesejável. Essencialmente, baseiam-se em dois grupos: moeda virtual e reputação. As soluções de moeda virtual [91] [92] [93] quantificam a vantagem de um nó em encaminhar tráfego alheio (ganha créditos que pode converter em recursos); as soluções baseadas em reputação [94] [95] [96] tentam promover colaboração ao punir nós que a rede (ou parte dela) globalmente concerte serem maliciosos.

No âmbito desta dissertação será assumido que os nós cooperam salvo menção em contrário.

4 AADQ.

Neste capítulo apresentam-se os principais contributos materiais deste trabalho. Em primeiro lugar, pormenoriza-se a visão particular de QdS para redes ad hoc e a descrição fundamentada do AADQ – o modelo de QdS derivado das reflexões ao longo desta dissertação. O TSQ (e a sua adaptação a ambientes totalmente distribuídos, o dTSQ) é a prova de conceito do modelo.

4.1. Crítica ao estado-da-arte.

Modelos.

Primeiro, viu-se já que os modelos de QdS para as redes com fios não são adequados às redes ad hoc (tal como existem) e qualquer tentativa de adaptação acaba por quebrar (pelo menos, parcialmente) o espírito dos modelos. É igualmente claro que, à data, não existem modelos universais de QdS para redes ad hoc. O único que reclama sê-lo é o FQMM mas é apenas uma adaptação minimalista dos modelos para redes com fios, tal como o D-Lite o é. Talvez o 2LQoS se aproxime mais de um modelo universal de QdS para redes ad hoc – mas peca por ser incompleto e não propor mecanismos para a sua concretização. É contudo muito útil como base de partida para projectar um genuíno modelo para redes ad hoc.

A tentação de considerar as redes ad hoc como caso particular das redes com fios tem sido grande. A excepção é o SWAN, numa abordagem de cruzamento de camadas ("cross-layer"), que combate o problema usando uma abordagem monolítica. Embora seja a *única* solução que cria real QdS nas redes ad hoc, peca por estar demasiado colada aos protocolos de transporte, limitando a versatilidade e flexibilidade da proposta.

Os resultados do INSIGNIA e do FQMM são, em termos absolutos, decepcionantes. É notório que tanto o INSIGNIA como o FQMM não conseguem fornecer garantias aceitáveis e, para a complexidade que se introduz no sistema (especialmente o

INSIGNIA), talvez seja preferível ter apenas o serviço de melhor-esforço. O INSIGNIA diminui ligeiramente o atraso (mas ainda mantendo-o crescente com a carga e com valores da ordem do segundo) e o FQMM aumenta levemente o débito-entre-extremos.

O SWAN consegue, *efectiva e absolutamente*, controlar o atraso (fica estável e contido com carga crescente) e só nesta proposta começa a haver QdS com relevância para o utilizador. Mas não é possível considerar esta proposta como modelo porque lhe falta uma série de características. Desde logo, se fossem implementadas várias classes de tráfego (algo que, por si, não parece complicado fazer, e.g. [101]), as classes com mais prioridade iriam competir e acabariam por ter um tempo de serviço divergente para carga crescente, destruindo a coerência dos resultados. Os bons resultados do SWAN provêm da prioritização do tráfego RT e da imposição de limites à congestionamento. Não existem formas simples de escalar QdS e os resultados acabam sempre por ser os melhores possíveis dadas as características da rede. Por outras palavras, não é possível definir limites para métricas (como atraso máximo) e adaptar o SWAN a requisitos particulares criando serviços mais elaborados como o CL do IntServ.

Portanto, a atitude de considerar que os modelos e técnicas das redes com fios vão servir também para redes ad hoc, necessitando "apenas" de mais ou menos engenhosas adaptações, deve ser abandonada. Mesmo levando em conta as especificidades das redes ad hoc (fragilidade das L1 e L2, nós heterogéneos, mobilidade, etc.), apenas se conseguirá realizar QdS em redes ad hoc aplicando pressupostos diferentes. Para clareza, destaco o primeiro pressuposto:

Pressuposto 1:

As redes ad hoc são uma generalização das redes com fios e não o inverso.

QdS em redes ad hoc.

Neste ponto, convém formular o conceito de QdS em redes ad hoc. Não se deve redefinir o conceito que se tenta realizar na esperança de reduzir a complexidade do problema. QdS

é um conceito já bem estabelecido, embora com subtilezas, e não pode ser diferente consoante o tipo de rede: é porventura o conceito mais perto do utilizador que, idealmente, deve ignorar que tipo de tecnologia suporta a comunicação.

Mas é claro que QdS hoje está a tomar três significados distintos. O primeiro, mais antigo e clássico, é o de emular comutação de circuitos sobre uma rede de pacotes. O segundo, muito próximo do negócio, é o de diferenciar utilizadores seja para evitar abusos ou para distinguir utilizadores via taxas suplementares. Finalmente, e apenas o refiro como "QdS" para efeitos de integração no tema, há as optimizações de técnicas já utilizadas. Em termos conceptuais, uma optimização não é QdS porque continua a não oferecer quaisquer garantias, como discutido atrás.

De uma forma ou outra, QdS continua a ter as seguintes características fundamentais:

1. garantias vincadas de serviço (atraso, largura-de-banda, pacotes perdidos, …)
2. disponível num espaço de tempo razoável (consoante a aplicação subjacente) e durante quanto tempo durar a utilização
3. cuja única métrica real é Existência-de-Serviço (sim ou não)

A segunda característica é fundamental porque se pode admitir que certa técnica garante, por exemplo, um atraso abaixo de um certo limiar mas apenas como média ao longo de um período de tempo muito grande. Algumas aplicações podem funcionar com este tipo de garantias mas não se pode considerar que seja, em termos estritos, QdS. Sumariamente: QdS é ter os recursos necessários, (quase) imediatamente disponíveis e enquanto for necessário. Rescrevendo este pressuposto num registo de linguagem sintético:

Pressuposto 2:
QdS é recursos suficientes, imediatos e enquanto for necessário.

Há, obviamente, sempre um intervalo de confiança associado. Mas não deve ser alargado arbitrariamente para reduzir a complexidade do problema. Nas redes com fios, estes

intervalos de confiança assentam essencialmente em falhas que ocorrem com probabilidades extremamente reduzidas – numa rede com fios, a periodicidade das quebras de ligação mede-se em semanas; numa rede ad hoc, pode ser ao fim de um segundo. Da mesma forma, as garantias dadas dependem essencialmente do serviço definido. O CL do IntServ ou o AF do DiffServ não garantem atrasos, p.ex., mas é de assumir que são suficientes para as aplicações mais comuns. Seja como for, e mesmo com serviços de métricas algo indefinidas, há sempre confiança na utilização da rede.

Resumindo, não deve haver uma redefinição do conceito para redes ad hoc, não importando a dificuldade do problema.

A questão de QdS pode ainda ser analisada segundo outro prisma que será muito útil à frente. QdS pode sempre ser entendido como a funcionalidade que suporta a deslocação de recursos de uns utilizadores para os outros, mais ou menos localizadamente. Para isto, torna-se imediatamente necessário algum esquema de diferenciação correspondendo à assinatura do tráfego, quer por pacote ou por fluxo. Esta noção vai contribuir para o 3º pressuposto.

Validade de encaminhamento de QdS e sinalização específica de QdS.

Encaminhamento de QdS, tal como tem sido entendido, tem validade apenas quando há recursos para tal e são estáveis. Numa rede ad hoc, dado o desconhecimento da rede, a sua mobilidade e dinâmica, é muito difícil obter conhecimento da rede em tempo real. A solução seria usar sinalização intensa – obviamente impraticável. Por outro lado, o "conhecimento" necessário da rede é de dois tipos. Primeiro há o conhecimento da estrutura da rede (tipo de nós); depois, há inferência de métricas em tempo-real. Obter tanto um como outro é impraticável. Daí que este trabalho encare com bastante pessimismo abordagens que se centram ou dependem de encaminhamento de QdS: para além de introduzirem tráfego não útil, a informação que dão é francamente limitada e, na maior parte das vezes, inútil por ter validade extremamente pequena, dada a dinâmica que se espera das redes ad hoc. Incluem-se aqui protocolos de encaminhamento que usam pacotes `HELLO` fazendo difusão de informação de encaminhamento e também de métricas acessíveis a cada nó (e.g. OLSR, Q-AODV).

De forma geral, a aplicabilidade de encaminhamento de QdS tem utilidade não para obter rotas com certas propriedades ligadas a métricas voláteis (atraso, largura-de-banda, variância do atraso, ...) mas, *apenas e só*, para se descobrir rotas com nós generosos, em termos de recursos e de colaboração. Por outras palavras, encaminhamento de QdS pode servir para eliminar uma rota que inclua um dispositivo de baixa capacidade numa rede com nós heterogéneos. Mas não deve servir para encontrar uma rota com uma dada métrica volátil.

Genericamente, mecanismos que usem recursos da rede para suportar QdS devem ser criteriosamente ponderados.

Finalmente, notar o que os dois grandes modelos de redes com fios usam para compensarem as suas lacunas. O IntServ/RSVP cria canais virtuais e reserva sempre uma fracção pequena dos recursos na rota; o DiffServ não oferece garantias rígidas mas houve um passo anterior que foi o de planear a rede de forma a que, a menos de um grau de confiança, haja sempre recursos disponíveis para servir certos agregados com SLAs mais exigentes. Há sempre, em qualquer dos casos, excesso de recursos e o propósito é sempre estabilizar a multiplexagem estatística e *nunca* criar recursos onde eles não existem. O 3º pressuposto combina esta reflexão com anteriores:

Pressuposto 3:

(a) QdS consiste em deliberada e criteriosamente deslocar recursos de uns utilizadores para outros e
(b) nunca em tentar criar recursos onde não existem.

Por muito óbvia que seja, especialmente a parte (b), há alturas em que a parca capacidade disponível, agravada pela dinâmica da rede, é totalmente ignorada esperando-se um comportamento impossível da rede.

Requisitos de um modelo de QdS.

Um modelo de QdS deve ter os seguintes requisitos:

- *granularidade*. Deve ser flexível e capaz de suportar serviços customizáveis e elaborados. É essa a definição própria de modelo. O principal problema do SWAN é essencialmente não derivar de nenhum modelo e limitar-se a aplicar uma série de técnicas – é sempre reengenharia. Por serviços elaborados, entenda-se os que são definidos à custa da combinação de várias métricas. Por exemplo, o CL do IntServ não garante nenhuma métrica, apenas que, subjectivamente, a rede se irá comportar, para um dado fluxo, como se não estivesse congestionada. No limite, um operador deve ter a flexibilidade de definir um leque de serviços diferente para cada utilizador e em tempo-real.

- *tráfego adicional pequeno*. O tráfego adicional deve sempre ser considerado em função da capacidade da rede. Uma rede adhoc facilmente tem uma capacidade por nó da ordem de 10 kbps. Dificilmente se poderá implementar QdS com sinalização persistente. Um trabalho teórico interessante seria o de estimar o mínimo de largura-de-banda necessário para sinalizar reservas. Esse mínimo pode ser da ordem de grandeza da própria capacidade da rede por nó numa rede ad hoc (mesmo de pequenas dimensões).

- *suporte a uma entidade de gestão*. Deve suportar processos de diferenciação, conjugados com a granularidade permitida, permitindo a uma entidade gestora deliberadamente alocar recursos a nós consoante o seu interesse. Pode haver duas intenções: permitir que certos nós, com um grau de importância superior aos restantes, tenham benefícios que outros não tenham e esses benefícios possam ser concedidos de forma escalável e simples a partir de mensagens de rede; e, segunda via, deslocar recursos de uns nós para outros, temporária ou permanentemente.

- *distribuição*. O modelo deve permitir operação totalmente distribuída, totalmente centralizada ou mista. Mas também não deve exigir um tipo de operação ou outro.

- *simplicidade.* Não deve exigir aos nós que, para usarem um mecanismo que podem não necessitar nunca, tenham de ter funcionalidades que limitem o seu desempenho. Deve tolerar cenários de nós muito simples e heterogéneos.

- *interoperação.* Um modelo de redes ad hoc deve possuir mecanismos para fazer o interface a modelos preexistentes como IntServ e DiffServ.

Planeamento.

A questão do planeamento considera-se essencial em redes ad hoc e costuma ser menosprezada ou mesmo ignorada. Até certo ponto, faz algum sentido na medida em que, como foi dito, o desconhecimento das características da rede ad hoc a que um nó se associa é grande e planear implica algum conhecimento prévio do cenário. Por outro lado, não faz sentido não usar de um algum planeamento no sentido de optimizar a rede para o fim desejado. Concretamente, este planeamento pode ter duas faces. Primeiro, criar ordem na rede; segundo, gerir recursos.

De notar que não há nada numa rede adhoc baseada em IP que faça a gestão de recursos. Um fluxo UDP simplesmente transmite à taxa a que a aplicação entrega os dados; um fluxo TCP é bem mais comportado mas, como foi visto, conseguir que o TCP funcione optimamente numa rede ad hoc é complicado, principalmente com vários fluxos TCP em competição na rede.

Especificamente, nada impede que um nó com uma aplicação que funcione sobre o UDP transmita ininterruptamente a uma taxa arbitrária. Um cenário simples será o de uma aplicação de vídeo em tempo-real. A ligação é feita via TCP mas a transmissão do fluxo de tempo-real ("stream") é sobre UDP. Como a ligação inicial não necessita de grande largura de banda, a ligação tipicamente é bem sucedida e o nó começa a transmitir cegamente se não houver nenhum esquema de reconhecimento ("acknowledge") do outro extremo. Neste caso, apenas está a gerar colisões na rede sem qualquer utilidade até que, p.ex., ocorram quebras de ligação forçadas pela quantidade de pacotes nas filas dos nós.

Resumindo, não existe nada que faça a gestão da capacidade livre do nó e a única forma de medir a capacidade livre da rota é transmitindo pacotes para a rede. Uma rede grande ainda mais facilmente entra em congestionamento por esta razão. De uma forma geral, o problema é pragmaticamente ignorado, confiando que alguma camada na rede irá tomar conta do sucedido. Nas redes com fios isto não acontece porque o problema está em mais ou menos largura de banda e não, simplesmente, conectividade.

4.2. AADQ.

Motivação.

Baseando-se nos pressupostos anteriores, esta dissertação propõe o AADQ: 'Adhoc Administered with DiffServ-like QoS'. Como prova de conceito, discute-se ainda o TSQ ('Time-Slotted QoS') que aplica o modelo entre outras formas possíveis de o fazer.

O modelo foi elaborado no sentido de conduzir a esquemas de utilização da rede que cumpram os pressupostos enunciados acima. São eles, agora integrados:

1. *As redes ad hoc são uma generalização das redes com fios e não o inverso.* O contributo para o AADQ consiste no seguinte: antes de se atacar o problema, e para usar a experiência das redes com fios nas redes ad hoc, há que emular primeiro uma rede com fios sobre uma rede ad hoc. Uma espécie de "wire-lization". Ao ser possível configurar e operar uma rede ad hoc como se fosse uma rede com fios, até MPLS/TE se torna possível. Do ponto de vista deste trabalho de mestrado, emular uma rede com fios sobre ad hoc resume-se a distribuir de forma controlada e explícita os recursos de largura-de-banda existentes na rede.

2. *QdS é recursos suficientes, imediatos e enquanto for necessário.* Implica, primeiro, que QdS deve ter o mesmo entendimento quer numa rede com fios, quer numa rede ad hoc; segundo, que se as garantias forem de médio ou longo prazo, deixam de o ser por definição.

3. *QdS consiste em deliberadamente deslocar recursos de uns utilizadores para outros e nunca em tentar criar recursos onde não existem.* A solução para o problema adoptada no AADQ consiste em limitar a utilização da rede de forma a que haja capacidade de manobra para assegurar QdS a certos nós. Se a rede não consegue por falta de capacidade, o cenário simplesmente não tem solução; se tem, e se houver razão para limitar até ao extremo nós não prioritários (incluindo silencia-los), que seja feito.

Torna-se claro o principal objectivo: *a essência do AADQ é o esforço de emular uma rede com fios sobre uma rede ad hoc.*

Modelo simples para a capacidade de redes ad hoc num cenário 'hotspot'.

Apresentam-se resultados por simulação para a capacidade de uma rede ad hoc tendo em conta dois modelos. O primeiro foi designado de *modelo do hop central* e o segundo de *modelo de tráfego distribuído.* Enquanto o primeiro é aplicável a redes em que há um nó que encaminha a maior parte do tráfego (por exemplo, um 'hotspot' alargado[1] em que o AP é o ponto de saída para a Internet), o segundo pretende cobrir a situação do tráfego igualmente distribuído pela rede toda, sem que existam zonas de tráfego preferidas.

Considere-se o cenário da Fig. 34. Os quatro círculos pretendem ilustrar os 4 hops existentes numa dada fracção da rede.

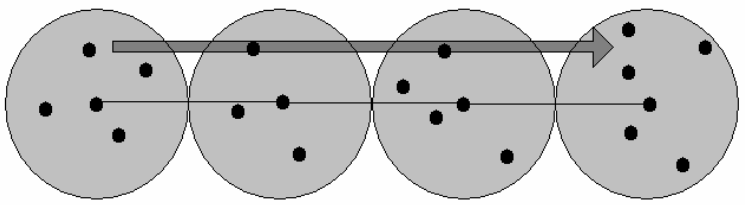


Fig. 34. *4 saltos de uma rede ad hoc.*

Admitindo que $\delta(r_0)$ é a densidade da rede ad hoc (nós/área) no ponto $r = r_0$ (simetria esférica), B é a largura de banda do meio (e.g., IEEE802.11b: 11 Mbps), βB é a fracção eficiente de largura de banda (taxa efectiva de linha) N nós, rotas de M saltos em média,

[1] "alargado" por haver nós que não têm ligação directa (L2) ao AP – caso contrário não seria uma rede ad hoc.

A_c a área do cenário, $A_{PHY} = \pi R_{PHY}^2$ a área correspondente ao alcance do PHY (antenas omnidireccionais), a capacidade do nó S na rota P vem

$$C_P = \min_j \left\{ \frac{\beta_j B}{N(r_j)} - \lambda_{in}^j - \lambda_{out}^j \right\} = \min_j \left\{ \frac{\beta_j B}{A_{PHY}\delta(r_j)} - \lambda_{in}^j - \lambda_{out}^j \right\} \quad \textbf{(eq. 2)}$$

para cada salto j em que $N(r_j)$ é o número de nós que partilha o domínio de colisões j ("salto"), λ^j é o tráfego externo que entra/sai na região de interferência do salto j e que *não* é gerado dentro do domínio de colisões j.

Notar que a primeira parcela refere-se *apenas* a tráfego gerado dentro do salto. A intenção de separar o tráfego interno de externo fica clara de seguida.

A probabilidade de existir um canal no caminho P é:

$$prob(\text{"}serviço\ \ disponível\text{"}) = prob\left[\bigcap_j \left\{ P^j : \left(C_P^j - \lambda_{in}^j - \lambda_{out}^j > R_{canal} \right) \right\} \right] \quad \textbf{(eq. 3)}$$

Esta expressão formaliza a noção de que, no limite $\lambda^j = 0$, a rede só não consegue criar o canal desejado se o caminho P não tiver capacidade intrínseca suficiente (a primeira parcela da intersecção). Havendo capacidade, reduzindo o tráfego externo até níveis convenientes, o canal é *sempre* possível. Até certo ponto, está aqui a essência do modelo de QdS proposto.

Considere-se agora o cenário da Fig. 35, para um modelo de "hop central".

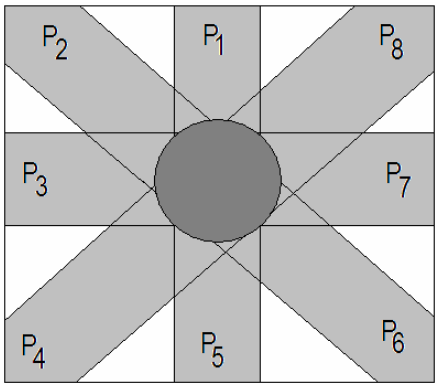


Fig. 35. *modelo de tráfego para um cálculo simples da capacidade de uma rede ad hoc (modelo 'hotspot').*

Se se admitir que a capacidade da rede e por nó (Ω) está limitada pelo salto central, aproximando a densidade a uma distribuição uniforme:

$$\Omega = \frac{\beta B}{A_{PHY}\delta(r=0)}\frac{1}{N}, \quad \text{com } \delta = \frac{N}{A_c} \qquad \textbf{(eq. 4)}$$

Se supusermos que a rede está perto da utilização máxima (sem congestionamento e admitindo que a capacidade é elástica), para densidades moderadas (<10 nós/salto) e tendo em conta os resultados de [13],

$$\beta = \frac{\tau_0}{\tau_0 + \tau_{MAC}} \sim \frac{1}{2}.$$

τ_0 é o tempo de transmissão (L/B) e τ_{MAC} corresponde ao atraso devido ao algoritmo de contenção do IEEE802.11 (desprezando o IFS).

Finalizando o raciocínio, a capacidade do canal vem:

$$\Omega = \frac{A_C B}{2A_{PHY}}\frac{1}{N^2} \qquad \textbf{(eq. 5)}$$

Este modelo foi simulado no ns-2 [97] para dois cenários: 750m X 750m e 2000m X 2000m. Para cada simulação tentou colocar-se a rede num ponto tal em que a perda global de pacotes era muito pequena (da ordem de 1%) para uma aplicação do tipo débito constante (CBR – 'Constant Bit Rate'). A mobilidade foi reduzida e apenas a suficiente para não haver problemas de simulação de propagação de sinal[1]. O protocolo de encaminhamento foi o DSR. Para garantir que cada nó possuía rotas para o destino e notando que o DSR demora algum tempo até construir rotas extremo-a-extremo, apenas se colheu estatísticas após 10s de simulação. Este período de aquecimento da simulação também elimina eventuais efeitos transitórios no início da simulação. Para não haver problemas de conectividade, o número de nós total foi de 200, claramente suficiente para preencher todo o cenário, já que o alcance de cada nó é de 250m. Contudo, o número de nós activos variava e era uma das variáveis sob investigação. As simulações eram curtas para que a mobilidade interviesse pouco:

- 45 s de tempo total
- primeiros 10s para obtenção de rotas
- período intermédio de 10s para eliminar pacotes transitórios na rede, sem transmissão de novos pacotes
- 25s de tráfego que contava para as estatísticas

Entre os 200 nós no total, o número de nós activos (que transmitiam pacotes) variou. Cada nó activo transmitia pacotes UDP a um ritmo fixo para um único nó, simulando um cenário de 'hotspot' em que todos os nós transmitem para um AP. A taxa de transmissão foi obtida, como referido, por tentativa e erro. No final, registou-se o atraso para cada pacote dos 25s finais. O tamanho dos pacotes era 256 bytes. Considerou-se este valor como médio num contexto de aplicações multimédia.

[1] Foi observado que o ns-2 tem alguns problemas para simulações de cenários com nós fixos. Estes problemas centravam-se essencialmente em quebras de ligação.

Em primeiro lugar, verificou-se se existe efectivamente uma curva de congestionamento com um "joelho" acentuado. A Fig. 36 representa-se a curva de perda de pacotes em função do débito. Num cenário 750mx750m, 20 nós transmitiam para o mesmo nó à taxa indicada pela abcissa. Fica bem claro que a taxa de pacotes não servidos pela rede tem duas zonas de funcionamento: uma com crescimento praticamente linear e outra constante de valores muito baixos.

Nas simulações subsequentes, tentou encontrar-se o valor do joelho para cada configuração. No âmbito desta secção, define-se *capacidade por nó* activo da rede ad hoc como a ordenada do joelho. No caso da figura (20 nós), a capacidade por nó activo é de cerca de 40 kbps. Notar que, em torno deste ponto, o equilíbrio é instável e a fracção de pacotes perdidos cresce rapidamente com a carga.

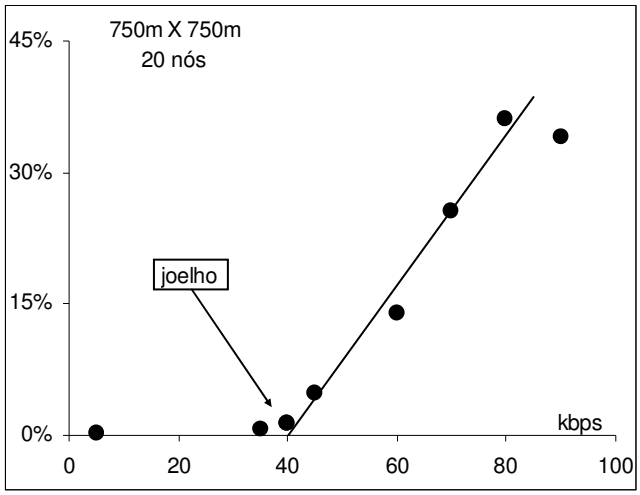


Fig. 36. *Taxa de perdas com carga crescente num cenário de 'hotspot'.*

Na Fig. 37 e Fig. 38 representa-se a variação da capacidade por nó com o número de nós activos para duas áreas de mobilidade. Notar a qualidade do ajuste: 99.5% num caso e 98% noutro. Notar também que, de acordo com estes gráficos, e notando a necessidade de débito das aplicações de hoje, uma rede ad hoc grande (no sentido de capacidade livre por nó) começa logo nos 10 nós.

Finalmente, notar que a expressão derivada anteriormente, eq. (5), onde se assumia um expoente de 2 é optimista pois para 750x750 o expoente é 1.1 e para 2000x2000 é 1.7. Notar igualmente que, para 750x750, o coeficiente de 1111.3 está uma ordem de grandeza abaixo do calculado via eq. (5) (daria 15764). A conclusão clara é que, retomando a eq. (3), o efeito do cruzamento de tráfego (λ^j) tem significativamente mais peso do que o admitido. Em resumo, a capacidade disponível por nó activo segue $\Omega \sim x^a$ com o efeito de cruzamento de tráfego a ser tão ou mais dominante do que a capacidade de um domínio de colisões.

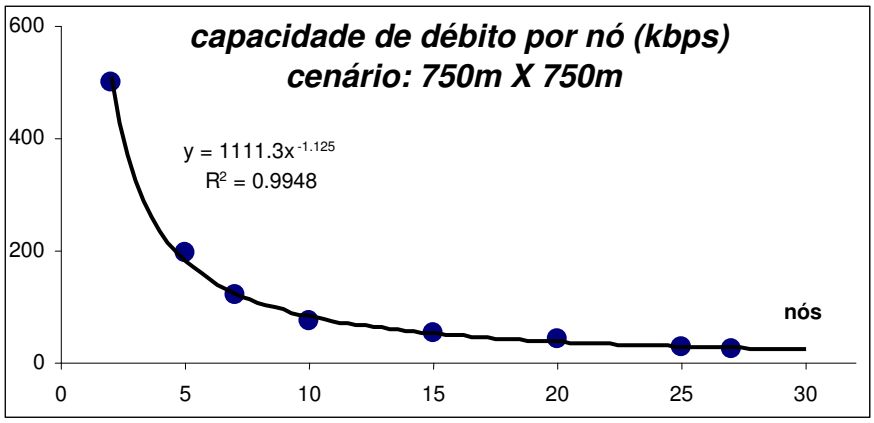


Fig. 37. *Taxa de perdas com carga crescente num cenário de 'hotspot' – área 750m X 750m.*

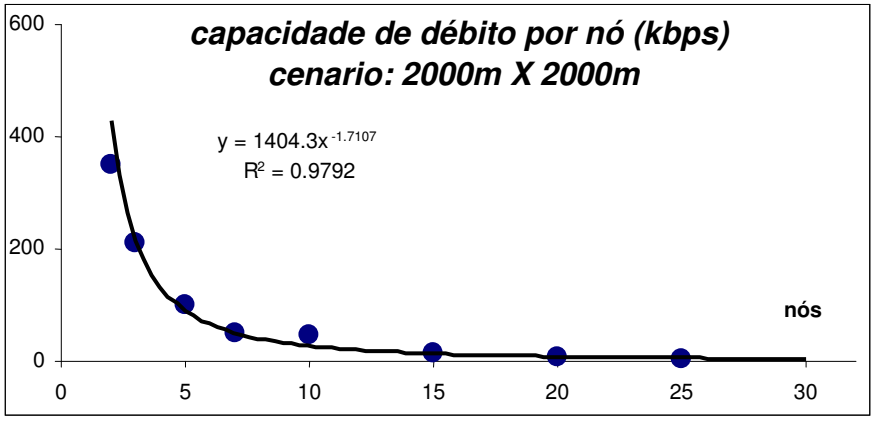


Fig. 38. *Taxa de perdas com carga crescente num cenário de 'hotspot' – área 2000m X 2000m.*

Efectuaram-se igualmente estudos sobre a distribuição do atraso. Na Fig. 39 encontra-se um histogramas com a contagem de pacotes para cada intervalo de atraso. O cenário é de

20 nós em 750m×750m e para duas utilizações da rede: imediatamente abaixo do joelho e bem acima do joelho de congestionamento.

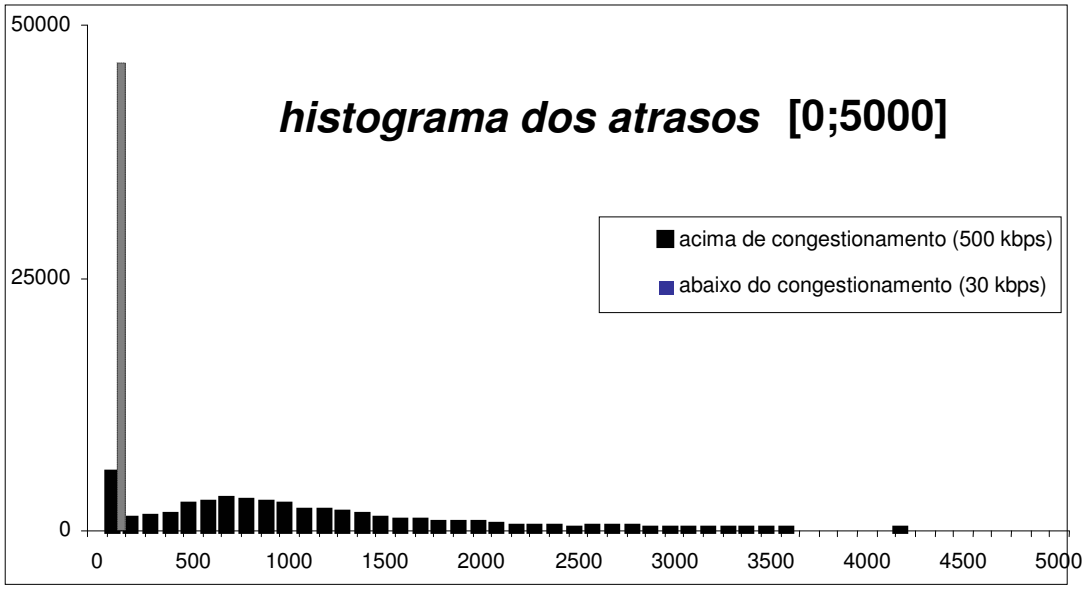
É claríssimo o compromisso largura de banda/atraso. O ponto de funcionamento da rede praticamente determina o "tipo" de atraso que cada pacote vai ter: determinístico se abaixo do joelho de congestionamento; espalhado se acima. Este resultado pode ser uma primeira validação de [100]. Notar que abaixo do joelho de congestionamento, todos os pacotes têm atraso muito baixo pelo que as ligações extremo-a-extremo se comportam como circuitos virtuais. Ou seja, no joelho de congestionamento pode assumir-se que o tráfego cruzado é nulo, pelo que a eq. (3) fica simplificada ao seguinte: para cada nó na rota, basta $C_P^j > R_{canal}$ para que o canal se estabeleça. Estes resultados serão capitalizados no modelo de QdS que esta dissertação propõe.

Finalmente, da análise da distribuição dos atrasos para duas áreas diferentes obtém-se a Fig. 40. Para áreas maiores (e mesma densidade), as rotas são maiores e, portanto, os atrasos estão mais espalhados – embora ainda significativamente contidos. Contudo, notar que para 2000m X 2000m o atraso atinge os 250 ms o que pode ser crítico para muitas aplicações. Isto pode indicar uma zona de validade no que concerne a densidade da rede normalizada ao alcance da tecnologia sem fios.

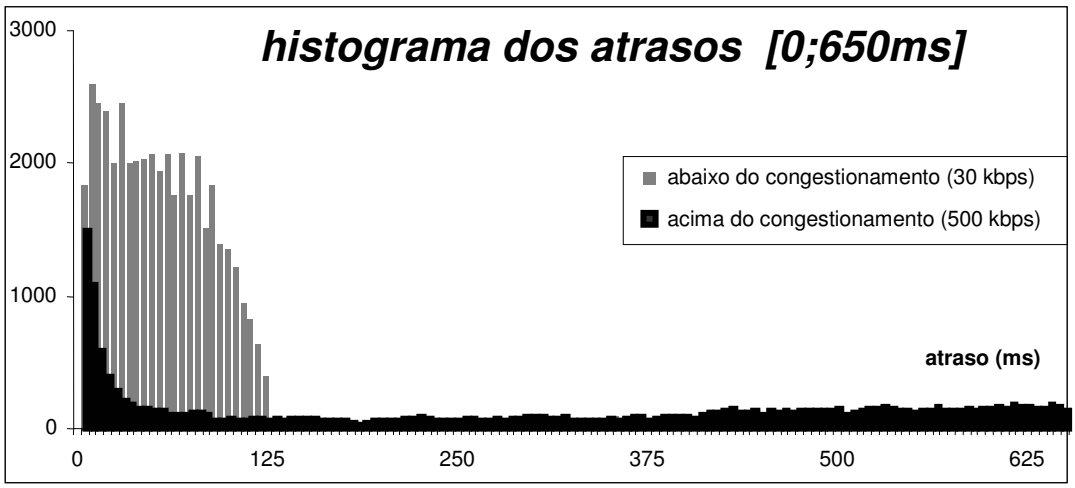
Integrando agora este estudo no contexto desta dissertação, pode extrair-se as seguintes conclusões, para um cenário de 'hotspot':

- as redes ad hoc facilmente ficam congestionadas a partir de um certo débito por nó (joelho de congestionamento)
- parece ser possível criar circuitos virtuais em redes ad hoc. Estes *canais* têm débito garantido, atraso máximo garantido e jitter controlado;
- o impacto de pouco tráfego adicional na rede pode ser muito alto, dependendo do ponto onde se está a operar a rede
- abaixo do ponto de congestionamento, a capacidade livre por nó parece ser bastante previsível

- a partir de 10 nós activos, a capacidade por nó reduz-se consideravelmente. A noção de tamanho de uma rede ad hoc pode quantificar-se a partir deste resultado: uma rede ad hoc de 10 nós é grande



(a)



(b)

Fig. 39. *Distribuição do atraso para 20 nós e área de 750m X 750m num cenário de 'hotspot'. (a) intervalo total de atrasos; (b) ampliação da zona 0—650 ms.*

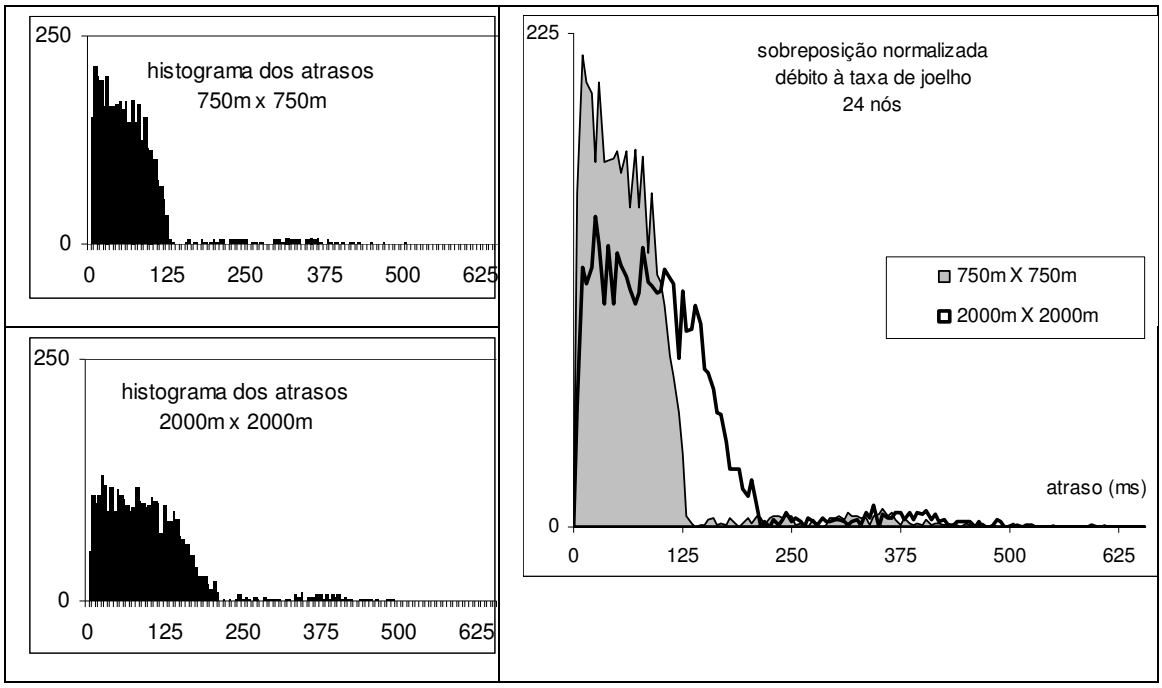


Fig. 40. *Distribuição dos atrasos para 20 nós e duas áreas diferentes.*

Componentes do AADQ.

Utilizando o resultado anterior, se C_P for a capacidade do caminho, P a probabilidade de existir um canal no caminho P vem:

$$prob(\text{"}serviço\ \ disponível\text{"}) = prob\left[\bigcap_j \left\{P^j : \left(C_P^j - \lambda_{in}^j - \lambda_{out}^j > R_{canal}\right)\right\}\right] \qquad \textbf{(eq. 6)}$$

λ^j é o tráfego externo que entra/sai na região de interferência do salto j.

Como dito, esta expressão formaliza a noção de que, no limite $\lambda^j = 0$, a rede só não consegue criar o canal desejado se o caminho P não tiver capacidade intrínseca (a primeira parcela da intersecção) suficiente. Havendo capacidade e reduzindo o tráfego externo até níveis convenientes, o canal é *sempre* possível.

As componentes do modelo que se propõe estão reunidas na expressão anterior e são as seguintes:

1. *definição de canais*. Para simplificar, propõe-se a definição de *canais*. Um canal é um fluxo coerente de dados segundo uma dada rota o qual é servido extremo-a-extremo com garantias previamente acordadas. A tab. 6 mostra alguns exemplos de canais. O canal em AADQ é o serviço de QoS por si.
2. *capacidade conhecida*. É necessário conseguir estimar a capacidade de uma dada rota e, em geral, a de uma rede ad hoc. Em secções anteriores foram descritos dois modelos para dois cenários (modelo do hop central e da capacidade distribuída). A capacidade predeterminada serve para manter a condição $C_P^j - \lambda_{in}^j - \lambda_{out}^j > R_{canal}$ possível.

serviço / canal	canal	largura-de-banda	atraso	perdas
melhor-esforço	BE	–	–	–
voz	VO	>16 kbps	<200 ms	<10%
vídeo	VI	>64 kbps	<400 ms	<10%
modem-analógico	MO	>56 kbps	–	<10%
melhor-esforço-melhorado	BBE	>16 kbps	–	–

tab. 6. *Exemplos de canais AADQ.*

3. *controlo de outras métricas*. Há duas métricas fundamentais e irredutíveis, do ponto de vista de utilização de uma rede de comunicações: o débito extremo-a-extremo e o atraso (e sua variância). Notar que a perda de pacotes significa atraso infinito. O débito extremo-a-extremo está assegurado ao garantir capacidade suficiente na rota. Quanto ao atraso, é necessário criar filas de tráfego para que certos pacotes tenham prioridade superior aos restantes ou, em termos mais genéricos, um tratamento diferenciado. Considera-se aqui que a forma de diferenciar pacotes é uma questão de implementação e *não* de modelo pelo que essas considerações serão discutidas à frente. Contudo, é de assumir ou o uso do campo DS no pacote IP ou análise transversal dos pacotes (p.ex., endereço IP da origem + porto TCP destino). O TSQ usa o campo DS.

4. *limites ao tráfego de fundo.* Havendo capacidade suficiente para criar o canal, pode ser necessário limitar o tráfego de fundo λ^j, de forma simples, distribuída e fácil de sinalizar em tempo-real.

5. *informação da qualidade do canal em tempo real.* O nó origem e as entidades de supervisão não têm forma de saber, pelos seus meios, se o canal efectivamente existe e qual o seu desempenho. Por esta razão, os nós relevantes (tipicamente, o destino do tráfego) têm de informar a rede. Mais uma vez, a forma de o fazer *não* respeita ao modelo sendo uma questão de implementação.

Nas Fig. 41 e Fig. 42 representa-se a arquitectura funcional do modelo.

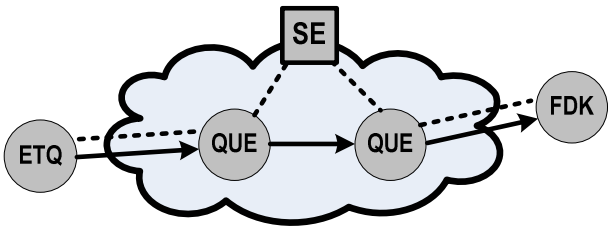


Fig. 41. *Arquitectura de rede do AADQ.*

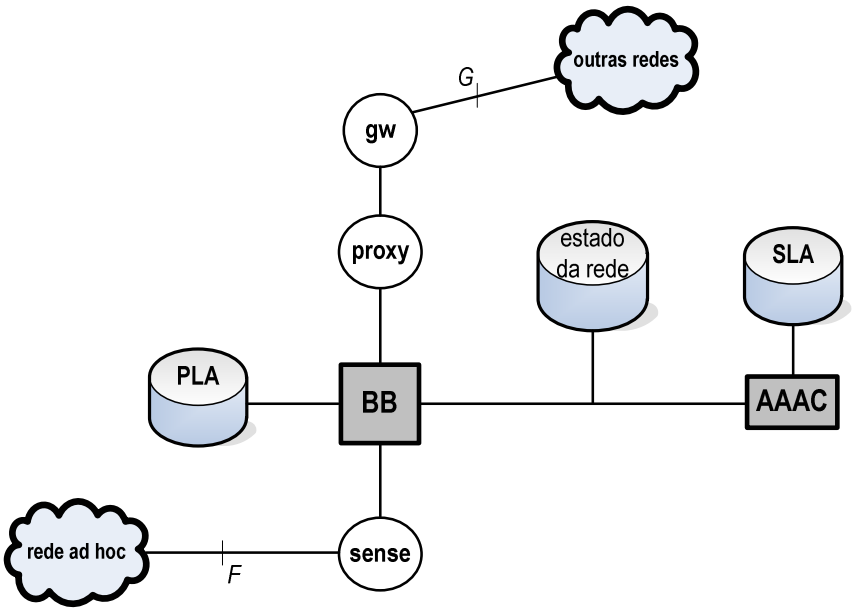


Fig. 42. *Arquitectura da entidade SE.*

Pode imaginar-se que o nó origem vê uma rede de trânsito à sua frente que transporta os pacotes até ao destino. Deste ponto de vista, existem os seguintes elementos:

- *Entidade Supervisionadora* (SE) – Fig. 42. É a entidade que planeia a rede, supervisiona e aloca recursos em tempo real. Pode também ser um nó da rede, no sentido que gera e absorve tráfego. O mais importante é que tem um conhecimento razoável dos recursos da rede (componente PLA, de "planeamento") e tenta ser justo ao atribuir homogeneamente os recursos da rede. Pode, contudo, via um SLA, permitir mais utilização da rede a certos nós. Usa sinalização (linhas a tracejado). A função pode incluir um Gestor-de-Recursos ("Bandwidth Broker"). Tomando como referência a Fig. 42, a arquitectura desta função tem as seguintes componentes:
    - *ponto F.* Interface com a rede ad hoc
    - *sense.* Para além de encaminhamento de/para a rede ad hoc, infere do tráfego o estado da rede no que concerne à capacidade disponível em termos de nós activos na rede, densidade da rede e recursos disponíveis.
    - *BB.* É o gestor geral da rede. Gere associações à rede, actualiza e lê a base de dados dos recursos disponíveis, calcula e distribui os parâmetros de gestão da rede e reserva ou liberta recursos alocados a certos utilizadores. a natureza destes recursos será discutida em pormenor adiante.
    - *AAAC.* Entidade que gere a identificação de utilizadores e processos dela desencadeados.
    - *SLA.* Base de dados que contem os contratos para cada utilizador.
    - *status.* Estado actual da rede (utilizadores, recursos, etc.)
    - *proxy.* Função que faz a interface a outros esquemas de QdS, nomeadamente conversão de sinalização.
    - *gtw e interface G.* Interface a outras redes.
- *nós encaminhadores* (QUE). São os nós que transportam o tráfego `origem` → `destino`.

- *nó destino* (FDK). Tem de informar o SE do estado do canal origem → destino. Se necessário, de acordo com os recursos previstos da rede e do SLA que origem e/ou destino mantêm, SE concede-lhes mais recursos.

- *nó origem* (ETQ). O nó origem compromete-se a regular a taxa de pacotes que coloca na rede numa sessão para a qual solicita QdS. Para outras sessões de tráfego, sujeita-se a melhor-esforço. ETQ ("etiqueta") significa que o nó, cooperante, não desobedece às instruções emitidas pelo SE.

Notar que há componentes que assumem uma rede ad hoc administrada. O modelo não obriga a que o seja, embora tenha sido pensado para esse cenário, sendo que, num cenário de total distribuição, técnicas alternativas suplementares têm de ser usadas em substituição da entidade supervisionadora da rede (SE: Supervising Entity) como veremos mais à frente. A metodologia de projecto deste modelo parte do princípio de que existe uma função de cariz central mas que pode estar implementada de forma distribuída.
Admitindo que a capacidade da rede consegue ser conhecida ou estimada (preferencialmente, por defeito), o SE atribui recursos a cada nó via sinalização. Cada nó, ao associar-se à rede, regista-se e o SE recalcula a taxa de oportunidade de transmissão para cada nó baseado no número de nós na rede, na densidade e no coeficiente de acoplamento de rotas ("route coupling"), e.g., acesso exterior via AP. Só o número de nós não é suficiente porque a capacidade da rede depende da densidade (partilha do domínio de colisões) e do número de nós (tamanho das rotas).

Numa primeira fase, o SE faz difusão para a rede, eventualmente usando um esquema semelhante ao MPR do OLSR (para minimizar a redundância de pacotes), anunciando a taxa máxima de colocação de pacotes na rede para cada nó. Há aqui uma subtileza necessária. Para que haja efectiva justiça ("fairness"), nós que usem rotas mais compridas (quase todos os protocolos de encaminhamento possuem directa ou indirectamente essa informação) podem transmitir mais já que a capacidade da rota depende do seu comprimento. O SE pode ter isso em conta. Para simplificar, pode ser considerado que há apenas um conjunto de parâmetros.

O planeamento deve incluir uma dada *margem de capacidade* não utilizada para que tráfego de sinalização seja prioritário. Contudo, o SE recalcula a parametrização da rede assim que ocorra algum evento que influencie a capacidade. Esses eventos podem ser:

- entrada e saída de nós e/ou alteração da densidade da rede (diminuição do espaço, p.ex.)
- solicitação de canais
- alterações na mobilidade da rede

Quando um nó pretende garantias da rede, solicita-as ao SE que, confrontando-as com o SLA respectivo, autoriza ou não. Ao autorizar, se houver capacidade disponível, apenas autoriza o nó; senão, coloca o nó em espera até a rede se reconfigurar (difusão dos novos parâmetros).

A ideia essencial é manter a rede abaixo do congestionamento ao admitir que a capacidade da rede é conhecida, ou pelo menos, passível de boa estimação por defeito, e ao não deixar os nós colocarem pacotes em excesso na rede. Para isso, dependente do papel actual do nó (ETQ ou QUE), o tratamento dado a cada pacote difere. Se a origem do pacote é o próprio nó, e se é tráfego genérico (para o qual não pediu tratamento especial da rede, submete-se ao SLA geral de tráfego melhor-esforço). Se é pacote especial, submete-o ao SLA aplicável, o qual teve prévia autorização do SE. Finalmente, se é tráfego não de utilizador (encaminhamento, sinalização, etc.), encaminha imediatamente (nó QUE).

Para além disso, a ideia de não deixar os nós colocarem pacotes em excesso é levada até ao limite. É comum haver apenas policiamento já depois de os pacotes serem transmitidos. O objectivo é limitar *logo na origem* a colocação de pacotes na rede – um mecanismo que, compreensivelmente, se designará por *admissão na origem*.

Desta forma, consegue-se o seguinte:

1. Não há pacotes em excesso na rede que ou seriam descartados por falta de capacidade ou por não cumprirem um SLA. Cada nó é responsável pela taxa de pacotes que coloca na rede. Por outras palavras, a admissão é um processo misto e feito a dois tempos: (1º) o próprio nó verifica se pode assegurar o fluxo de acordo

com (2º) a informação que o SE fornece e obteve antes da rede começar a operar (PLA) e ajustando-a em tempo real consoante o perfil de utilização que os nós derem dela.

2. O SE fornece os dados da rede (taxa máxima de pacotes na rede por cada nó) baseado em conhecimentos válidos da rede. Há aqui o pressuposto de cada nó se autenticar, ou pelo menos registar-se, no SE antes de usar a rede para que o SE tenha conhecimento. A capacidade intrínseca da rede depende do número de nós e da respectiva densidade (comprimento médio das rotas).

3. Não há sinalização de rota nem reservas, simplificando significativamente cada nó. O controlo de tráfego de cada nó é trivial de ser implementado. Um nó que implemente uma pilha IP facilmente implementa uma fila com controlo da taxa média de pacotes que coloca na rede.

4. A operação da rede, na figura do tráfego de rede (encaminhamento, autoconfiguração, etc.), não é perturbada. Apenas se limita o tráfego de utilizador.

5. São tolerados nós que não implementem o AADQ. Naturalmente que dificultará a operação da rede, mas é possível fazer coabitar serviços derivados do AADQ com simples conceitos de mero melhor-esforço.

É necessário definir com mais rigor dois conceitos usados livremente no texto: *parâmetros da rede* e *recursos*. *Parâmetros da rede* são dados de algum tipo que o SE distribui para a rede em difusão e que cada nó usa para configurar os seus algoritmos de policiamento de tráfego. O SE deve calcular estes parâmetros de forma a poder acomodar um nível mínimo de tráfego de fundo e reservas em curso associadas a SLAs. Desta forma, perante um pedido de recursos, o SE toma uma de três decisões:

- aceita como está e concede a reserva (rede com capacidade suficiente)
- nega (rede com capacidade insuficiente e/ou SLA do utilizador não conforme o pedido)
- coloca o utilizador em suspensão enquanto tenta libertar recursos da rede.

De um ponto de vista de interface entre outras arquitecturas de QdS, nomeadamente, para redes fixas, estes parâmetros de rede são usados como mediadores: p.ex., uma mensagem RSVP é interpretada e os parâmetros de rede são calculados e difundidos pela rede ad hoc

criando a reserva RSVP implícita tão fiel quanto possível ao serviço original. Este trabalho de interface entre redes diferentes será objecto de um estudo futuro.

Recursos no AADQ são canais extremo-a-extremo regulados pela (eq. 6). Estes canais são orientados a largura-de-banda com atraso muito baixo e constante (praticamente só o tempo de propagação dos pacotes ao longo da rota). Contudo, há pistas na literatura [100], suportadas por resultados já apresentados (ver Fig. 39 e Fig. 40), que indicam claramente que se pode converter largura-de-banda por atraso. Se forem desenvolvidas técnicas eficientes, dotar-se-á o AADQ de notável flexibilidade.

Finalmente, é notório que, em termos de camadas, é criada um plano extra de gestão, muito semelhante ao OAM&P de outras arquitecturas. Em especial, o 'P' de "provisioning" é fulcral no AADQ – Fig. 43 – pois é este processo que assegura que a rede funcione imediatamente, embora possivelmente abaixo do ponto óptimo (capacidade máxima).

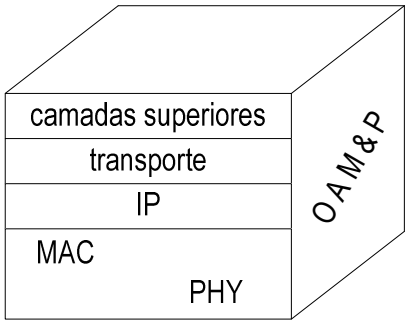


Fig. 43. *Perspectiva do AADQ segundo camadas de rede.*

Facilidade de concretização.

Resta compreender se o modelo funcional é implementável, particularmente na versão totalmente distribuída. O TSQ e o dTSQ (versão totalmente distribuída) são possíveis implementações do modelo proposto e são discutidas adiante.

O outro grande problema consiste na cooperação dos nós. A cooperação é uma questão que ultrapassa a tecnologia. Por um lado, há razões para acreditar que não será um problema fundamental. Primeiro, as redes ad hoc são frágeis e para atacar a rede há muitos

outros pontos fracos pelo que a falta de cooperação será apenas mais um. Em segundo, o SE, na versão administrada, pode oferecer contrapartidas, nomeadamente, acesso à Internet: o SE poderá ser nesse caso o ponto de acesso. Quando foi referido que o SE não transporta tráfego (embora seja um nó na rede) era de uma perspectiva de arquitectura – fisicamente o SE e o AP podem ser o mesmo dispositivo. Neste cenário, o controlo da rede ainda se torna mais fácil porque o SE pode determinar com rigor a quantidade de tráfego que entra e sai da rede e daí estimar o tráfego total na rede. Finalmente, a questão da colaboração resume-se a este ponto: ou todos os nós implementam a técnica ou não. Se sim, têm QdS; se não, ninguém tem e o mais provável é a rede ficar absolutamente intratável e transformar-se num enorme domínio de colisões. Estritamente, não é necessário todos os nós implementarem o modelo para haver QdS. Mas as garantias possíveis sobem com a taxa de adesão.

Para uma primeira observação, este modelo funcional facilita a gestão de largura de banda. Esta métrica é decisiva e, devidamente controlada, gere as restantes. Por exemplo, se o atraso apenas for composto de propagação e processamento (i.e., sem atrasos de espera em filas), está sempre limitado a tempos muito pequenos o que satisfaz qualquer aplicação de tempo-real. Mas não tem de assim ser. O atraso é facilmente controlável por uma fila simples com prioridades, algo que o modelo também prevê com filas e mecanismos de admissão em cada nó.

4.3. Prova do conceito: TSQ.

4.3.1. TSQ

Motivação.

O principal objectivo do TSQ (*Time-Slotted* QoS) é provar que os conceitos do AADQ são realizáveis. Não se pretendeu uma solução óptima, dadas as eventuais alternativas, mas tão simplesmente realizar QdS, seguindo o AADQ. QdS, mais uma vez, no sentido estrito do termo: garantias efectivas extremo-a-extremo. Desde logo surgiram vários compromissos,

nomeadamente largura-de-banda vs atraso e taxa de entrega de pacotes vs largura de banda disponível.

O TSQ concretiza 3 componentes funcionais do AADQ:

1. planeamento prévio da rede
2. sinalização SE ←→ rede
3. gestão de tráfego em cada nó e diferenciação de pacotes

Funcionamento.

Para simplificar quer a tarefa de planeamento, quer os pedidos dos nós, definiram-se os *canais* da tab. 7. A capacidade da rede e respectiva margem num dado momento deve ser definida em função da quantidade desses canais disponíveis.

serviço	canal	largura-de-banda	atraso	perdas
melhor-esforço	BE	–	–	–
voz	VO	>16 kbps	<200 ms	<10%
vídeo	VI	>64 kbps	<400 ms	<10%
melhor-esforço-melhorado	BBE	>16 kbps	–	–

tab. 7. *Canais definidos para o TSQ.*

Cada nó encaminha tráfego alheio e de encaminhamento sem restrições. Apenas há regulação do tráfego próprio. Se um nó deseja um dado serviço, envia o túplo $(n_{VO},\ n_{VI},\ n_{BBE})$ definido da seguinte forma:

$$A_{livre} = [n_{VO} \quad n_{VI} \quad n_{BBE}] * [VO \quad VI \quad BBE]^T$$

em que $(n_{VO},\ n_{VI},\ n_{BBE})$ significa o número de canais pretendido dos serviços, respectivamente, de voz, vídeo e melhor-esforço-melhorado (tab. 7)

Sendo um dos requisitos essenciais do AADQ, naturalmente que os canais definidos na tab. 7 são meros exemplos – outros poderia ser definidos.

Planeamento prévio da rede.

A capacidade de uma rede ad hoc foi já discutida anteriormente. Uma solução analítica completa e exacta não existe, por ora. Para as simulações do TSQ fixou-se um cenário e determinou-se por tentativa-e-erro a capacidade disponível por nó no joelho de congestionamento.

Gestão de tráfego na origem.

Consiste em duas partes: diferenciação de pacotes e controlo da taxa de pacotes colocados no sistema.

A diferenciação de pacotes é útil para definir prioridades do ponto de vista do escalonamento na fila de saída. Por exemplo, um pacote transportado por um canal VO deve ter mais prioridade do que um VI dado que o atraso é mais apertado. A diferenciação de pacotes propõe-se fazer-se via campo DS no pacote IP.

O controlo da taxa de pacotes colocados no sistema consiste num esquema inspirado em TDM. O nome "Time-Slotted QdS" vem daqui. Os requisitos principais deste esquema, ao mesmo tempo que serve ao AADQ, são os seguintes:

(i) ser muito simples do ponto de vista de implementação;

(ii) promover a descorrelação entre pacotes das diversas fontes para minimizar colisões no meio IEEE802.11;

(iii) ser facilmente parametrizável com sinalização mínima.

A Fig. 44 mostra o tráfego à saída e a Fig. 45 mostra o algoritmo de escalonamento geral do TSQ.

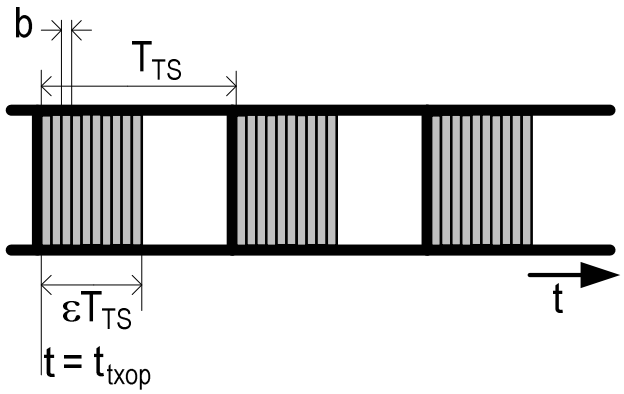


Fig. 44. *Algoritmo de escalonamento.*

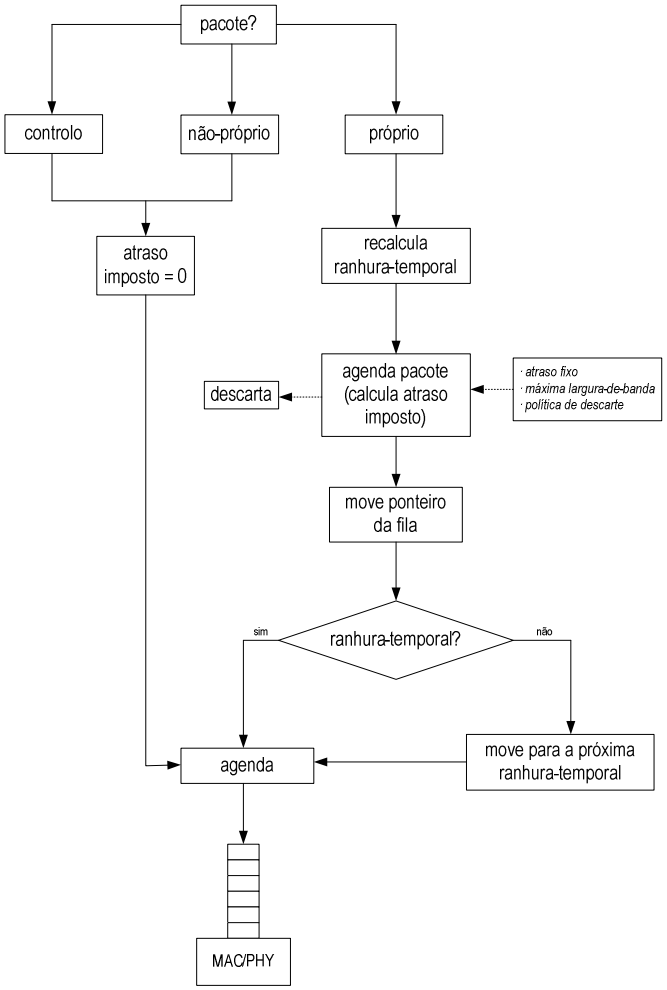


Fig. 45. *TSQ: tratamento do tráfego.*

Tal como dito, cada nó apenas regula o seu próprio tráfego podendo, contudo, dar prioridades diferentes a vários pacotes de terceiros para os quais funciona como encaminhador. A regulação do tráfego próprio é feita pela conjugação das seguintes funcionalidades:

1. *fila TSQ*. cada nó apenas pode transmitir se houver pacotes na fila TSQ.

2. *ranhuras-temporais*. Um pacote apenas pode ser agendado para transmissão no meio em intervalos de tempo definidos (ranhuras-temporais) – parâmetro ε. Notar que T_{TS} / ε não tem necessariamente de ser um número inteiro.

3. *espaçamento inter-pacote*. o intervalo de tempo entre dois pacotes consecutivos tem de ser, no mínimo, b. Este parâmetro pode ser fixo, incluindo ser nulo, ou calculado à custa de uma taxa pré-definida e do tamanho do pacote ($b = L / R$).

4. *descarte*. Um pacote pode ser descartado se só for possível agendá-lo para n ranhuras-temporais para a frente. O nó é que determina este parâmetro. P.ex., para optimizar blocos de tráfego ("bursts"), pode definir-se n ilimitado.

5. para promover a descorrelação entre transmissão de pacotes, o instante $t = t_{TXOP}$ é aleatório, definido pelo próprio nó e redeterminado ao fim de p ranhuras-temporais.

SE apenas tem de publicar os parâmetros ε, b (ou R_{TSQ}) e T_{TS} para que cada nó possa regular o seu tráfego melhor-esforço. São estes os parâmetros da rede para o TSQ. É notório que, para meramente regular a taxa de saída de pacotes, há redundância. O objectivo foi tentar construir um conjunto de parâmetros que *linearizasse* (para facilitar o projecto) a saída. Se para fluxos sobre o UDP, p.ex., é trivial (quanto maior a fracção de tempo a transmitir, mais saída há) para transportes com autoregulação, como o TCP, há mais graus de liberdade.

Análise dos parâmetros principais.

Todas as simulações efectuadas nesta secção têm a duração de tráfego de 200s e o protocolo de encaminhamento é o DSR. Os nós estão parados durante toda a simulação. As ranhuras-temporais têm a duração de 2 ms.

O gráfico da Fig. 46 mostra o efeito de regulação do TSQ sobre o TCP. Os nós estão a 2 saltos de distância e apenas existe uma sessão. É claro o comportamento assimptótico para o limite sem TSQ. O atraso na fila (parâmetro b) é nulo. Equivale a apenas averiguar se o instante actual está dentro da ranhura-temporal ou não. Se sim, é imediatamente agendado para transmissão (fila do MAC); senão, é agendado para o início da próxima ranhura-temporal. Um efeito interessante é a linearização do TCP a partir dos 70%, ou seja, há uma faixa em que o débito à saída é proporcional ao número de ranhuras-temporais alocadas.

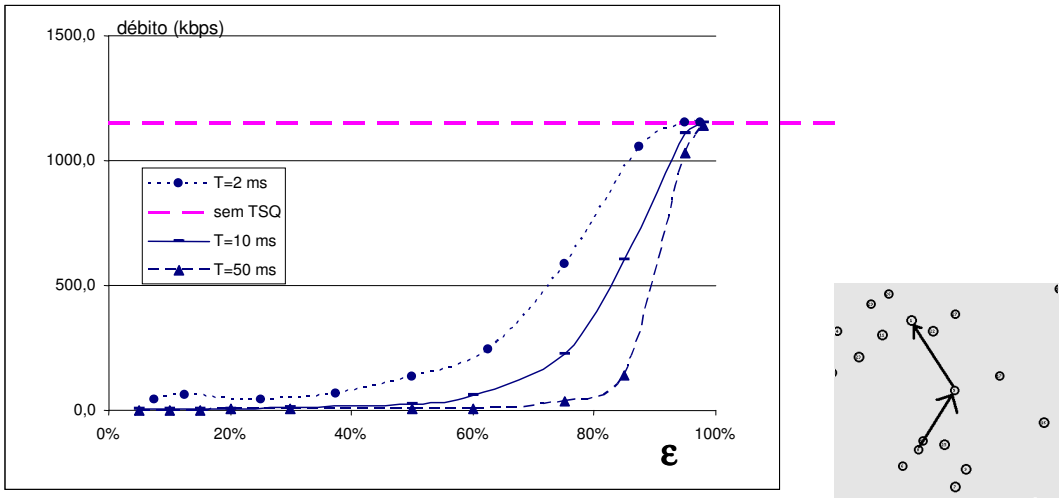


Fig. 46. R*egulação do TSQ sobre fluxos TCP – 2 saltos, sessão única.*

Na Fig. 47, mostram-se os resultados para duas sessões TCP em que há um nó em comum (na ordenada está o débito em kbps e na abcissa o parâmetro de regulação). As figuras apresentam o débito de cada nó em dois modos de funcionamento: usando o TSQ e sem usar o TSQ. Na figura à esquerda representa-se o débito isolado de cada fluxo e na figura à direita a soma dos dois débitos.

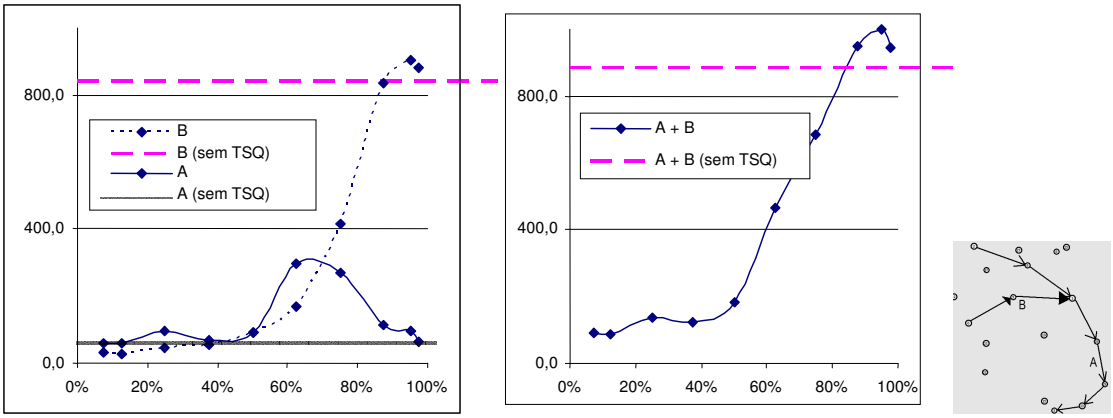


Fig. 47. *Regulação do TSQ de fluxos TCP – 2 sessões com um salto em comum.*

Talvez o efeito mais interessante é o aumento da capacidade global. Com efeito, acima dos 80% de utilização a capacidade está acima do conseguido sem usar TSQ. Na realidade, os ganhos foram conseguidos pela sessão B que está praticamente sempre acima do conseguido sem TSQ e, a partir dos 70%, à medida que o TSQ perde influência (aumento da utilização), começa a convergir para o valor sem TSQ.

4.3.2. dTSQ

Total distribuição do TSQ.

Para tornar o TSQ totalmente distribuído, é necessário implementar as funções do SE de forma distribuída. Para tal, propõe-se os seguintes passos.

1. *modo TSQ*. Um nó que pretenda QdS faz difusão de um pacote especial (primitiva `SWITCH_TO_TSQ`) e fica à escuta de tráfego. Cada nó retransmite aos seus vizinhos. Para evitar inundação descontrolada ("flood storm"), um nó que já tenha retransmitido não volta a retransmitir[1]. Se não houver tráfego ao fim do tempo T_a, o nó que solicitou TSQ considera que todos os nós entraram no modo TSQ. T_a depende naturalmente do tamanho da rede. Deve ser suficiente para que a difusão chegue aos extremos da rede. Se o tráfego persistir, repete a operação P_a vezes. Se

[1] Pode também ser usado um esquema de inundação semelhante ao do OLSR (conjunto MPR) mas implica um passo inicial de definição do conjunto MPR para cada nó.

ao fim das repetições o tráfego ainda persistir, o nó conclui que os nós ou não aceitam entrar em modo TSQ ou não o implementam. *Desta forma, vários tipos de nós podem coabitar não sendo mandatório que todos os nós o implementem.* O nó desiste e submete-se a melhor-esforço.

2. *registo.* Quando a rede entra em TSQ, os nós começam a trocar mensagens de difusão requisitando informação sobre a densidade da rede e o seu tamanho. É simples de obter, à custa de pacotes `HELLO`, já que cada nó tem conhecimento dos vizinhos. Relembre-se que *densidade* é o número de nós a 1 salto e que, portanto, partilham o mesmo espaço de colisões.

3. *tamanho da rede.* É um parâmetro complicado de se obter, ao contrário da densidade. Contudo, pode assumir-se um tamanho *médio* fixo das rotas de M saltos: $3 \geq M \geq 6$

4. *parâmetros de escalonamento.* Com a informação anterior, a capacidade é inferida, a menos de uma margem de segurança $\alpha_{capacidade}$. Cada nó consultará uma tabela que mapeia o tamanho e a densidade da rede aos parâmetros do TSQ.

5. *capacidade disponível.* Cada nó sabe os canais que transporta e sabe quais não estão a ser utilizados nas rotas que o incluem como salto (DSCP). Se houver algum livre, utiliza-o; senão, espera que esteja livre. Notar que a capacidade é essencialmente relevante na rota do nó. Logo, analisando o seu próprio tráfego e ouvindo tráfego alheio (o possível), será possível estimar com bastante precisão (a analisar futuramente) os canais disponíveis.

6. *papel do destino.* O nó destino informará o nó origem da disponibilidade do canal, a partir da taxa de pacotes que estiver a receber e o seu atraso. O atraso em si é difícil de medir nestas condições mas a variância do atraso não. Além disso, quase todos os protocolos de serialização ("streaming") possuem algum mecanismo que numera os pacotes. Aqui o ónus acaba por ser da aplicação que determina ou não se o canal tem qualidade ou não. Além disso, o destino apenas sinaliza que o canal não está disponível, sendo que é assumido pela origem que o canal está livre na falta de indicação remota.

7. *resolução de colisões.* Um nó que receba a indicação do destino como canal não disponível (baixa entrega de pacotes e/ou alto atraso) – primitiva `NO_CHANNEL` –, mesmo que a origem julgue estar, pode tomar duas decisões. Se o canal está a ser utilizado por si há bastante tempo, persiste enviando tráfego normalmente, implicitamente sinalizando o nó que se tenta apoderar do canal (que concluiu ter o canal livre), já que o seu destino indicar-lhe-á que o canal não está disponível – primitiva `NO_CHANNEL`. Ao fim de algum tempo, o nó que há mais tempo estiver a usar o canal sairá vencedor e o nó que tentou começar a usa-lo desistirá e esperará um tempo T_{hold} até tentar outra vez. Este tempo deverá ser aleatório ou sujeito a um regime de contenção para facilitar a resolução de colisões e potenciar justeza.

O dTSQ será futuramente validado.

4.3.3. Simulações.

Objectivos.

O objectivo central e que define o sucesso ou não do modelo e a respectiva implementação é conseguir criar canais. Para o efeito, as simulações consistem numa rede ad hoc com vários nós sobre a qual correm vários fluxos TCP, que representam tráfego de fundo melhor-esforço e, em simultâneo várias sessões CBR correspondendo a chamadas de voz ou vídeo prioritárias. A definição dos canais está na tab. 8. Cada cenário consiste na geração de nós aleatoriamente distribuídos pela área disponível e com percursos e movimentos independentes de acordo com o modelo de mobilidade utilizado RWP [36].

canal	suporte	débito-entre-extremos	atraso	entrega de pacotes
voz (VO)	UDP	> 16 kbps	< 200 ms	> 90%
vídeo (VI)	UDP	> 64 kbps	< 400 ms	> 90%

tab. 8. *Definição dos canais usados na simulação.*

Para reduzir o tempo de simulação, usaram-se aglomerados de canais de 16 kbps. Mais concretamente, usaram-se 2 sessões de 64 kbps e 1 de 32 kbps, a que correspondem 6

canais VO e 1 canal VI. Nas estatísticas, consideram-se pacotes dentro-do-perfil e pacotes fora-do-perfil (OOP). Pacotes OOP são determinados pela soma dos pacotes perdidos (transmitidos subtraídos dos recebidos) e pacotes com atraso superior ao especificado (>200 ms ou >400 ms, para um canal VO ou VI, respectivamente).

Parametrização das simulações.

Na tab. 9 estão indicados os valores usados na simulação do TSQ. Foi usado o ns-2 [97] com um módulo desenvolvido que implementa o TSQ. Notar que estes valores foram o resultado de iterações: foi preciso afinar o algoritmo de regulação do TSQ para que a capacidade desejada estivesse disponível.

Nos primeiros 2 segundos de cada simulação existe uma sessão CBR entre os pares S-D para que o protocolo de encaminhamento pudesse antecipadamente encontrar rotas e o atraso final não tivesse em conta esta latência. É uma situação real já que não é frequente haver aplicações que começam imediatamente a transmitir pacotes para o destino. O mais comum é haver o estabelecimento de uma sessão via TCP e só depois começar a haver fluxos RT. O propósito do tráfego inicial era simular este efeito. Contudo, os atrasos adicionais devido a re-descobertas de rotas foram incluídos no atraso final dos pacotes.

Em cada cenário, correm em paralelo 6 fontes de tráfego VO (ou 2 fontes de tráfego VI e 4 canais VO) e 4 sessões FTP de fundo entre pares colocados aleatoriamente. Considera-se que o canal foi criado com sucesso se, ao longo de toda a simulação, mais de 90% dos pacotes chegaram ao destino com atraso inferior a 200 ms (canais VO) ou 400 ms (canais VI). Para cada sessão foram determinados o número total de pacotes transmitidos, o número total de pacotes recebidos e, destes, o respectivo atraso. Pacotes OOP são aqueles que ou foram perdidos em filas de nós intermédios ou chegaram mas com um atraso superior ao desejado: para canais VO, são os pacotes com atraso superior a 200 ms e para canais VI, os pacotes com atraso superior a 400 ms). A percentagem de pacotes OOP é calculada da seguinte forma:

$$\eta_{OOP} = \frac{N_{atrasados} + N_{perdidos}}{R_{CBR} \cdot \Delta t_{tráfego}}.$$

R_{CBR} é a taxa de saída do fluxo CBR e $\Delta t_{tráfego}$ é a duração efectiva do fluxo.

Desta forma, o número de pacotes esperados a priori é confrontado com os pacotes OOP. Naturalmente, deve verificar-se $R_{CBR} \cdot \Delta t_{tráfego} = N_{chegados} + N_{perdidos}$.

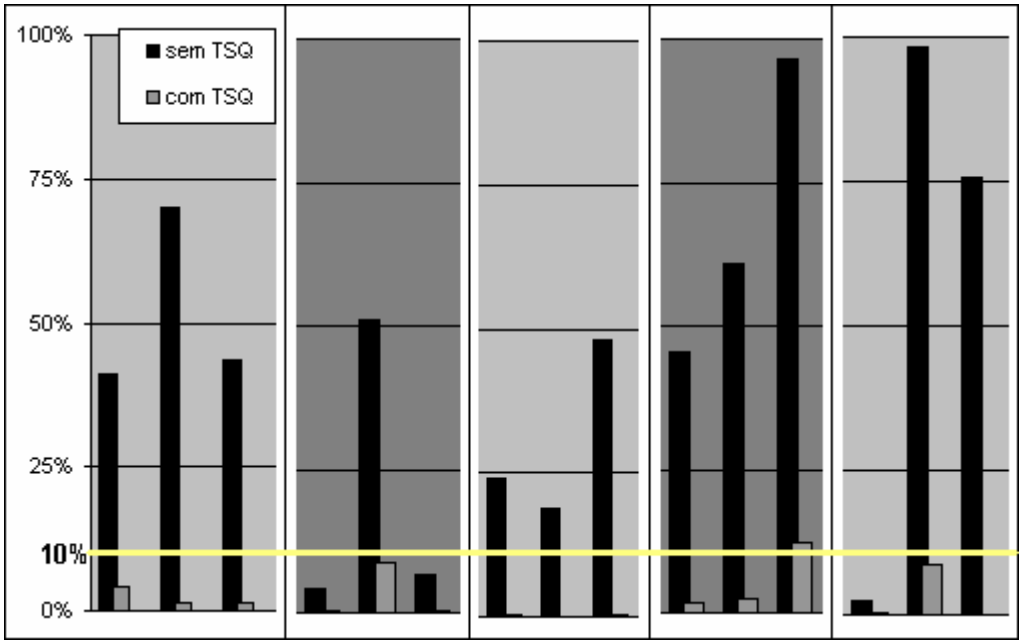
O protocolo de encaminhamento utilizado foi o DSR mas, como deverá ser claro neste ponto, o TSQ é independente do protocolo de encaminhamento. O AADQ obriga a que o seja.

período da ranhura-temporal	2	ms
fracção de utilização	0.8	ms/ms
atraso fixo	0,5	ms
simulação	305	s
tráfego	300	s
comprimento dos pacotes (CBR)	256	B
modelo de mobilidade	RWP	
área do cenário	1000 × 1000	m × m
velocidade	3	m/s
tempo de pausas	0	s
MAC+PHY	IEEE802.11b	
tamanho da fila do nó	50	pacotes
taxa de linha	11	Mbps
alcance do nó	250	m
propagação	"2rayground"	
protocolo de encaminhamento	DSR	
salvação ("salvaging")	sim	
espera máxima de salvação até descarte	30 s	
respostas gratuitas ("gratuitious replies")	sim	
memória local secundária ("secondary cache")	sim	
tempo mínimo entre `RREQ`	10	s
variância do atraso entre `RREQ`	0.5	s
fluxos	não	
espera por rota do pacote na fila	30	s

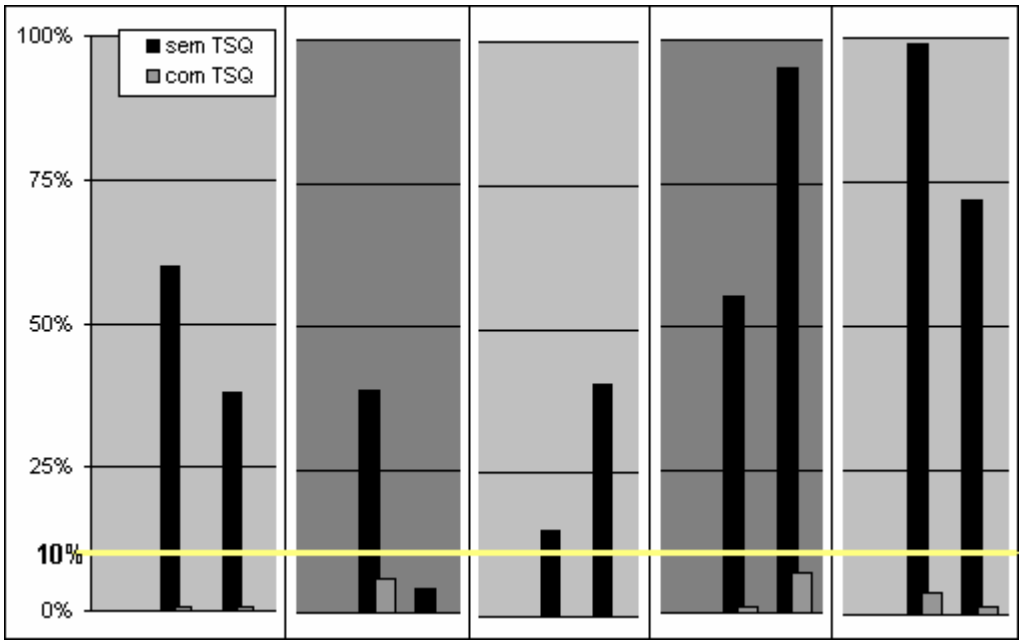
tab. 9. *Parâmetros da simulação.*

Análise dos resultados.

Na Fig. 48 estão representados os resultados. Dos 5 cenários gerados, apenas 1 sessão em 15 teve um valor acima de 10% de pacotes OOP (12.3%). Em contrapartida, sem o TSQ apenas 2 sessões em 15 foram estabelecidas com sucesso.



(a)



(b)

Fig. 48. *Resultados do TSQ. (a) estatísticas para os canais VO. (b) estatísticas para os canais VI. Há 5 cenários diferentes gerados (sem repetições).*

A análise do cenário da sessão que falhou em particular mostrou que a mobilidade criou severas dificuldades ao protocolo de encaminhamento durante um período pequeno mas significativo da simulação (perda de conectividade durante alguns segundos). Com menor

mobilidade (p.ex., um tempo de pausa superior), estes efeitos eram esbatidos. Notar que foi usado um tempo de pausa nulo mas velocidades não muito altas, num cenário 'worst-case'. A única solução possível é optimizar o protocolo de encaminhamento para que seja restabelecida conectividade o mais rapidamente possível – concretamente, num tempo que não afecte a aplicação em causa.

Notar que são valores medidos numa filosofia de *snapshot*, na medida em que não houve qualquer repetição para cada cenário. É QdS em tempo-real e absoluto. Exceptuando a sessão referida com 12.5% de pacotes OOP, todas as outras sessões de QdS foram estabelecidas imediatamente e durante todo o período de simulação.

Na tab. 10 estão os valores de débito. Na linha "todas FTP+CBR", o débito-entre-extremos foi calculado da seguinte forma: para TSQ é a soma da linha anterior e o débito de todas as sessões CBR; para colunas "sem TSQ", é a soma da linha anterior com o débito das sessões CBR pesado à fracção de entrega de pacotes para as 3 sessões do mesmo cenário (gráfico da fig. 51).

	cenário 1		**cenário 2**		**cenário 3**		**cenário 4**		**cenário 5**	
(kbps)	**sem TSQ**	**TSQ**	**sem TSQ**	**TSQ**	**sem TSQ**	**TSQ**	**sem TSQ**	**TSQ**	**sem TSQ**	**TSQ**
FTP 1	176	110	248	137	178	118	30	122	555	170
FTP 2	112	104	245	124	40	89	793	131	543	169
FTP 3	55	82	60	122	481	129	146	115	52	112
FTP 4	27	82	353	159	126	129	103	76	105	145
todas FTP	**369**	**378**	**907**	**542**	**824**	**466**	**1071**	**444**	**1256**	**597**
todas FTP + CBR	**446**	**538**	**1033**	**702**	**936**	**626**	**1122**	**604**	**1322**	**757**

tab. 10. *Resultados do TSQ – sessões FTP de fundo nas simulações.*

Em primeiro lugar, notar que o TCP ficou muito mais justo, fundamentalmente porque ficou limitado. Em segundo lugar, como não poderia deixar de ser, parte da capacidade da rede foi deslocada para as sessões UDP. Curiosamente, no primeiro cenário, enquanto se mantém QdS, o débito-entre-extremos global acabou por aumentar. O resultado final é

que, para o cenário descrito, seriam possíveis (mais de) *10 canais de voz* e (mais de) *2 canais de vídeo* a 64 kbps.

5 Conclusões.

5.1. Conclusões principais.

Esta dissertação centrou-se no fornecimento de QdS em redes ad hoc. Mostrou-se que, se o tema de QdS, genericamente, é vasto e transversal, particularmente para as redes ad hoc, pode assumir contornos de extrema complexidade. Esta complexidade deriva essencialmente das especificidades destas redes tais como os efeitos cruzados das camadas de rede e o ambiente de mobilidade.

Ao contrário da maioria das propostas para QdS em redes ad hoc, o AADQ/TSQ *efectivamente* garante QdS em redes ad hoc, ao mesmo tempo que adquire o estatuto de modelo de QdS, dada a sua generalidade de aplicação. Além disso, as garantias são absolutas, instantâneas, escaláveis, incrementáveis e manipuláveis por uma entidade supervisionadora, abrindo caminho a um modelo de negócio. A versão distribuída, que não foi testada, parece permitir distribuição.

Nesta dissertação associou-se a noção de *planeamento* às redes ad hoc. Com efeito, se uma qualquer outra rede deve ser minimamente planeada antecipadamente, o planeamento das redes ad hoc é sistematicamente ignorado. Um dos contributos desta dissertação é o propor esse passo inicial: planeamento da rede orientado à capacidade, tendo em conta que o tráfego melhor-esforço deve conviver com tráfego que necessita de outros serviços. Este planeamento é depois utilizado para aspectos de controlo de admissão.

Um estudo sobre a capacidade das redes ad hoc num cenário de 'hotspot' foi apresentado. Para além de mostrar que é possível criar QdS implicitamente, quantificou conceitos como "tamanho" de uma rede ad hoc e a capacidade disponível para cada nó.

Além disso, criaram-se mecanismos para emular uma rede com fios sobre uma rede ad hoc (do ponto de vista de rede). O grande problema é conseguir capacidade suficiente nas

redes ad hoc, algo que é assumido sem grande reflexão numa rede com fios e que foi provado (para um cenário de 'hotspot') não ser válido para uma rede ad hoc. A partir do momento em que a capacidade é conhecida, a convergência torna-se possível – há agora uma fronteira clara entre os dois tipos de redes e quantifica-se na capacidade disponível por nó. O modelo AADQ desenvolvido atinge estes objectivos.

Foi também desenvolvida uma ferramenta (o algoritmo de regulação de débito-entre-extremos do TSQ) que lineariza o débito do TCP, evitando o comportamento extremamente agressivo de captura de recursos ao mesmo tempo que cria justiça na competição por recursos entre vários nós.

5.2. Direcções Futuras.

Ao longo da dissertação foi-se dando indicações para investigação futura. Sumariamente, podem dividir-se os contributos em dois tipos: não específicos do AADQ e específicos do AADQ.

Aspectos não específicos do AADQ:

1. Seria muito interessante analisar o comportamento do *TCP* se a principal métrica que usa para regular a janela de transmissão fosse não o RTT mas *o número de pacotes ao longo da rota*. Por exemplo, se se assumisse que cada nó teria espaço para pacotes e o nó mais congestionado tivesse, no instante t_0, $N_k(t_0)$ pacotes, a diferença $N^{\max} - N_k(t_0)$ poderia servir para regular a janela de transmissão. Poder-se-ia, adicionalmente, tal como o SWAN, usar-se mecanismos explícitos de sinalização de congestionamento (ao invés de deixar o TCP assumi-lo por si), como o ECN.

2. As *optimizações e extensões ao DSR* propostas merecem alguma investigação. Não só dotariam o DSR de características úteis como também (e agora com carácter opinativo) se levava mais a sério o DSR como o melhor protocolo (genérico) de

encaminhamento para redes ad hoc. As razões que o levam, na maior parte das vezes, a ser preterido em relação ao AODV seriam colocadas em causa. Essas optimizações e extensões são as seguintes:

a. ARP2.5, para reduzir drasticamente o peso nos pacotes devido ao encaminhamento-na-origem
b. sondas para manter uma memória local reduzida ao mínimo sem perda significativa de eficiência
c. rotas tendencialmente mais curtas à custa de algum tráfego de encaminhamento adicional na rede com `RREQ` esporádicas
d. extensão para encaminhamento de QdS

3. Um trabalho teórico interessante seria o de estimar o mínimo de largura-de-banda necessário para sinalizar reservas. A julgar pelos resultados da simulação do modelo do 'hotspot', esse mínimo é da ordem de grandeza da própria capacidade da rede por nó numa rede ad hoc (mesmo de pequenas dimensões). Se a rede for operada significativamente abaixo do joelho da capacidade, a rede está sub-optimizada. Para pontos de operação acima do joelho, um pequeno acréscimo de pacotes na rede gera um grande aumento na taxa de perdas.

4. Dum ponto de vista de investigação, seria interessante não descartar totalmente a possibilidade de encaminhamento assistido por uma entidade administrativa. Seria interessante quantificar até que ponto se consegue deslocar complexidade dos nós e da rede para uma entidade de rede, simplificando os nós. Por exemplo, pode ser vantajoso um nó especial estar estrategicamente colocado num ponto em que, pela informação a que tem acesso, possa fornecer rotas optimizadas a alguma métrica. Estas técnicas poderiam dar um auxílio precioso no contesto de redes de sensores, já que os nós que as compõe podem ter requisitos muito apertado de tamanho e poder de processamento.

Contributos específicos do AADQ:

1. O *dTSQ* precisa de ser demonstrado. A sua relativa complexidade precisa de ser simulada para encontrar falhas no algoritmo. Em especial, o algoritmo de resolução de colisões (competição por canais), a avaliação distribuída dos parâmetros da rede (tamanho e densidade) e a distribuição dos parâmetros do TSQ (difusão básica ou emulada sobre ponto-a-ponto) podem não funcionar como o esperado.

2. O AADQ/TSQ está seguramente sub-optimizado. Seria interessante dotá-lo de maior adaptatividade que permita adequar o controlo de débito das fontes dinamicamente, ora aumentando ora diminuindo consoante o nível de serviço detectado. A capacidade seria aumentada.
3. O TSQ é também um algoritmo de escalonamento. Durante a optimização do algoritmo ficou claro que, para regular convenientemente o TCP, o período de contenção não pode ser muito longo (abaixo de cerca de 50 ms). Isto coloca sérios entraves a uma implementação do TSQ na medida em que sugere implementação por 'hardware'. Deve analisar-se o efeito de se usar outros algoritmos de escalonamento, mais simples de se usarem por 'software' com a devida criação de novos parâmetros de rede.

4. Seria útil expandir o AADQ/TSQ para implementar *interoperação entre uma rede ad hoc e uma rede com fios* – particularmente, interoperar o AADQ/TSQ com o DiffServ ou o IntServ. Por exemplo, com base no TSQ, é necessário investigar como se converte uma reserva CL do RSVP nos parâmetros de rede do TSQ.

5. É de enorme importância desenvolver técnicas eficientes para se poder trocar atraso e débito-entre-extremos. Se for possível, a noção de canal do AADQ ganha enorme flexibilidade ao mesmo tempo que optimiza a rede. Em particular, a relação atraso-débito deve ser investigada.

6. O TSQ proposto, muito simples, usa o facto de a rede oferecer um atraso baixo quando a rede é operada abaixo do joelho de congestionamento.

Claramente, este serviço não fornece garantias robustas mas é similar ao serviço CL do IntServ. Se dotarmos os nós de funcionalidades que permitam gerir o tráfego de forma mais inteligente (como prioridades) e de um protocolo de sinalização semelhante ao RSVP (devidamente simplificado), é possível tornar estas garantias arbitrariamente robustas e definir-se serviços arbitrariamente elaborados.

Referências.

[1] IETF, RFC2501: Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations, 1999

[2] Filipe Cunha, Vítor Silva, "Segurança em redes IP", trabalho para a cadeira de mestrado IGRS, Univ. Aveiro, 2002

[3] http://www.car-to-car.org/

[4] http://www.bedd.com/

[5] Angela Doufexi, et al., *A Comparison of the HIPERLAN/2 and IEEE802.11a Wireless LAN Norms*, IEEE Communications Magazine, May 2002, pp. 172-180.

[6] Rollet, R.; Mangin, C.;*IEEE802.11a, IEEE802.11e and HiPeRLAN/2 Goodput Performance Comparison in Real Radio Conditions* Global Telecommunications Conference, 2003. GLOBECOM '03. IEEE , Volume: 2 , 1-5 Dec.2003 Pages:724 – 728

[7] Habetha, Jörg and Dutar, R. and Wiegert, J. *Performance evaluation of HiPeRLAN/2 multi--hop ad hoc networks*, In Proc. European Wireless, p.p. 25-31, Florenz, 02/2002,

[8] Konstantinos Oikonomou, Athanasios Vaios, Sebastien Simoens, Pietro Pellati and Ioannis Stavrakakis, *Centralized Ad-Hoc Network Architecture (CANA) Based on an Enhanced HiPeRLAN/2 System*, Personal Indoor Mobile Radio Communications (IEEE PIMRC 2003), 7-10 September, Beijing, China

[9] M. Veeraraghavan, N. Cocker, T. Moors, *Support of voice services in IEEE802.11 wireless LANs*, IEEE Infocom 2001, April 23-26, 2001, Anchorage, Alaska.

[10] J. Y. Li, C. Blake, D. S. J. De Couto, H. I. Lee, and R. Morris, *Capacity of Ad Hoc Wireless Networks,* MobiCom'01, pp. 61-69, Rome, Italy, Jul. 2001.

[11] P. Gupta and P.R. Kumar, *The Capacity of Wireless Networks*, IEEE Trans. on Information Theory, Vol. 46, pp. 388-404, Mar. 2000.

[12] M. Grossglauser and D.N.C. Tse, *Mobility Increases the Capacity of Ad Hoc Wireless Networks*, IEEE/ACM Trans. on Networking, Vol. 10, pp. 477-486, Aug. 2002.

[13] M. M. Carvalho and J.J. Garcia-Luna-Aceves, *Delay Analysis of IEEE802.11 in Single-Hop Networks*, in Proceedings of ICASP 2003

[14] M. Gerla, K. Tang and R. Bagrodia, *TCP performance in wireless multi--hop networks*, 1998

[15] M. Gerla, R. Bagrodia, L. Zhang, K. Tang, and L. Wang, *TCP over Wireless Multi--hop Protocols: Simulations and Experiments*, IEEE ICC, 1999

[16] Shugong Xu and Tarek Saadawi. *Does the IEEE802.11 MAC protocol work well in multi--hop wireless ad hoc networks?* IEEE Communications Magazine, 39(6):130-137, June 2001.

[17] T.E. Klein, K.K. Leung, R. Parkinson and L.G. Samuel *Avoiding Spurious TCP Condição de tempo-expirados in Wireless Networks by Delay Injection*, , Proceedings of IEEE Globecom 2004 (2004)

[18] Vikas Kawadia and P.~R.~Kumar, *Experimental Investigations into TCP Desempenho over Wireless Multi--hop Networks*, April 15, 2005.

[19] K. Chandran, S. Raghunathan, S. Venkatesan, and R. Prakash, *A feedback based scheme for improving TCP performance in ad-hoc wireless networks*," in Proc. of Int'l Conf. on Distributed Computing Systems (ICDCS'98), pp. 472-479, 1998.

[20] Barraca, JP, Girão, J, *MANET – Mobile Ad-hoc Networks – introducing MANET*", relatório interno, DET/UA

[21] V. Park and M. Corson. *A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks*. IEEE Infocom, 1997.

[22] S. Ni, Y. Tseng, Y. Chen, and J. Sheu, *The broadcast storm problem in a mobile ad hoc network*, in Proceedings of ACM/IEEE International Conference of Mobile Computing and Networking (MOBICOM'99), Sept. 1999.

[23] H.Zhang, Z.P.Jiang, *Performance Analysis of Broadcasting Schemes in Mobile Ad Hoc Networks*, IEEE Communication Letters, vol. 8, no. 12, pp. 718-720, Dec. 2004.

[24] IETF, RFC3561: Ad hoc On-Demand Distance Vector (AODV) Routing, 2003

[25] Charles E. Perkins and Elizabeth Royer *Ad-hoc On-Demand Distance Vector Routing* , Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, New Orleans, LA, February 1999, pp. 90-100.

[26] C. Perkins and P. Bhagwat. *Routing over Multi--hop Wireless Network of Mobile Computers*. SIGCOMM '94 : Computer Communications Review, 24(4):234-244 , Oct. 1994

[27] Janne Lindqvist: *Counting to infinity*, documento capturado na internet

[28] Chakeres, I, et al., *Dynamic MANET On-demand Routing Protocol (DYMO),* disponível em http://moment.cs.ucsb.edu/pub/draft-ietf-manet-dymo-00.txt

[29] Sumit Gwalani, Elizabeth M. Belding-Royer and Charles E. Perkins. *AODV-PA: AODV with Path Accumulation.*" Next Generation Internet Symposium, held in conjunction with ICC, Anchorage, Alaska, May 2003.

[30] Johnson, D, et al., *The Dynamic Source Routing Protocol*, disponível em http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-10.txt, July 2004

[31] S. R. Das, C. E. Perkins, and E. M. Royer, *Performance comparison of two on-demand routing protocols for ad hoc networks*, in IEEE Infocom 2000, Mar. 2000.

[32] IETF, RFC3626: Optimized Link State Routing Protocol (OLSR), 2003

[33] X. Zou, B. Ramamurthy, S. Magliveras *Routing Techniques for Wireless Ad Hoc Networks-Classification and Comparison*, Proceedings of the Sixth World Multi-conference on Systemics, Cybernetics, and Informatics--SCI, July 2002

[34] P. Jacquet, A. Laouiti, *Analysis of Mobile Ad-hoc Network routing protocols in random graph models*, relatório nº3835, INRIA, 1999

[35] Thomas Heide Clausen, Philippe Jacquet, Laurent Viennot *Comparative Study of Routing Protocols for Mobile Ad-hoc NETworks*, IFIP Med-Hoc-Ned 2002, in proceedings.

[36] Christian Bettstetter and Christian Wagner. *The Spatial Node Distribution of the Random Waypoint Mobility Model.* In Proc. of the 1st German Workshop on Mobile Ad-Hoc Networks (WMAN'02), Ulm, Germany, March 25-26, 2002.

[37] Jiang, W., and Schulzrinne, H. *Comparisons of FEC and codec robustness on VoIP quality and bandwidth efficiency*. In Proceedings of ICN 2002 (Atlanta, GA, August 2002).

[38] J. Broch, D.A. Maltz, D. B. Johnson, Y-C. Hu, and J. Jetcheva. *Performance comparison of Multi--hop wireless ad-hoc networking routing protocols*. In Proceedings of the 4th International Conference on Mobile Computing and Networking (ACM MOBICOM '98), October 1998, pages 85-97.

[39] S. R. Das, C. E. Perkins, and E. M. Royer, *Performance comparison of two on-demand routing protocols for ad hoc networks*, in IEEE Infocom 2000, Mar. 2000.

[40] Hong Jiang; Garcia-Luna-Aceves, J.J. *Performance comparison of three routing protocols for ad hoc networks*; Computer Communications and Networks, 2001. Proceedings. Tenth International Conference on, 2001; Page(s): 547 –554

[41] Boukerche, A *Performance comparison and analysis of ad hoc routing algorithms*";.; Performance, Computing, and Communications, 2001. IEEE International Conference on, 2001; Page(s): 171 – 178

[42] J. J. Garcia-Luna-Aceves and M. Spohn, *Source-Tree Routing in Wireless Networks*, Proceedings of 7th International Conference on Network Protocols, 1999.

[43] Charles Perkins, Elizabeth Royer, Samir Das, and Mahesh Marina. *Performance of two on-demand Routing Protocols for Ad-hoc Networks*. IEEE Personal Communications, February 2001, pages 16-28.

[44] J. Boleng, W. Navidi, and T. Camp, *Metrics to Enable Adaptive Protocols for Mobile Ad Hoc Networks*, Proceedings of the International Conference on Wireless Networks (ICWN '02), pp.293-298, 2002.

[45] Tan, H. X. and Winston K. G. Seah, *Dynamically Adapting Mobile Ad Hoc Routing Protocols to Improve Scalability – The AODV Example*, Proceedings of IASTED International Conference on Communication Systems and Networks (CSN2004), Sept 1-3, 2004, Marbella, Spain.

[46] R.S. Prasad, M. Murray, C. Dovrolis, and K.C. Claffy. *Bandwidth Estimation: Metrics, Measurement Techniques, and Tools*, IEEE Network, November-December 2003.

[47] I. Cidon, R. Rom, and Y. Shavitt, *Analysis of multi--path routing*, IEEE/ACM transaction on Networking , vol.7, no 6, pp.885-896, Dec. 1999

[48] Yashar Ganjali, Abtin Keshavarzian *Load Balancing in Ad Hoc Networks: Single-path Routing vs. Multi--path Routing*, (Stanford University)

[49] Peter P. Pham, Sylvie Perreau, *Performance analysis of reactive shortest path and multi--path routing mechanism with load balance*, IEEE INFOCOM 2003 - The Conference on Computer Communications, vol. 22, no. 1, Mar 2003 pp. 251-259

[50] M. K. Marina and S. R. Das*, On-demand Multi-path Distance Vector Routing in Ad Hoc Networks,* Proceedings of IEEE ICNP, November 2001

[51] Asis Nasipuri, Robert Castaneda, and Samir R. Das. *Performance of multi-path routing for on-demand protocols in ad hoc networks*. ACM/Kluwer Mobile Networks and Applications (MONET) Journal, 6(4):339--349, 2001.

[52] J.J. Garcia-Luna-Aceves, S. Vutukury, and W.T. Zaumen, *A Practical Approach to Minimizing Delays in Internet Routing Protocols*," Proc. IEEE ICC '99, Vancouver, Canada, June 6--10, 1999.

[53] S.J.Lee and M.Gerla*, Split Multi-path Routing with Maximally Disjoint Paths in Ad Hoc Networks*, In Proceedings of the IEEE ICC, pages 3201--3205, 2001.

[54] Peter P. Pham, Sylvie Perreau, *Performance analysis of reactive shortest path and multi--path routing mechanism with load balance*, IEEE INFOCOM 2003 - The Conference on Computer Communications, vol. 22, no. 1, Mar 2003 pp. 251-259

[55] M.R. Pearlman, Z.J. Haas, P. Sholander, and S.S. Tabrizi *On the Impact of Alternate Path Routing for Load Balancing in Mobile Ad Hoc Networks*. Proceedings of IEEE/ACM Mobihoc 2000.

[56] Asis Nasipuri, Robert Castaneda, and Samir R. Das. *Performance of multi-path routing for on-demand protocols in ad hoc networks*. ACM/Kluwer Mobile Networks and Applications (MONET) Journal, 6(4):339--349, 2001.

[57] Aristotelis Tsirigos Zygmunt J. Haas, Siamak S. Tabrizi , *Multi--path Routing in mobile ad hoc networks or how to route in the presence of frequent topology changes* , MILCOM 2001.

[58] K. Wu and J. Harms, *On-demand multi-path routing for mobile ad hoc networks*, Proceedings of 4th European Personal Mobile Communication Conference (EPMCC 01), Vienna, Austria, Feb. 2001, pp.1-7.

[59] C. Lee, X.-H. Lin, and Y.-K. Kwok, *A multi-path ad hoc routing approach to combat wireless link insecurity,* in Proceedings of IEEE International Conference on Communications (ICC), vol. 1, April 2003, pp. 448--452.

[60] P.R. Pearlman, Z.J. Haas, P. Scholander, and S.S. Tabrizi, *Alternate Path Routing in Mobile Ad Hoc Networks,* IEEE MILCOM'2000, Los Angeles, CA, October 22-25, 2000

[61] C. Aurrecoechea, A. Cambell, and L. Hauw, *A survey of QoS architectures*, in 4th IFIP International Conference on Quality of Service, Paris, France, March 1996.

[62] IETF, RFC1633: Integrated Services in the Internet Architecture: an Overview, 1994

[63] IETF, RFC2205: Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification, 1997

[64] IETF, RFC2210: The Use of RSVP with IETF Integrated Services, 1997

[65] IETF, RFC2212: Specification of Guaranteed Quality of Service, 1997

[66] IETF, RFC2211: Specification of the Controlled-Load Network Element Service, 1997

[67] IETF, RFC2475: An Architecture for Differentiated Services, 1998

[68] IETF, RFC2474: Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, 1998

[69] IETF, RFC2598: An Expedited Forwarding PHB, 1999

[70] IETF, RFC2597: Assured Forwarding PHB Group, 1999

[71] IETF, RFC2990: Next Steps for the IP QoS Architecture, 2000

[72] http://www.ietf-nsis.org

[73] IETF, RFC2998, A Framework for Integrated Services Operation over Diffserv Networks, 2000

[74] Harpreet S. Arora and Harish Sethu *A Simulation Study of the Feasibility of Differentiated Services Framework for QoS in Mobile Ad Hoc Networks*, , Proceedings of the Applied Telecommunications Symposium (part of Advanced Simulation Technologies Conference), April 2002, San Diego, California, USA.

[75] H. Xiao , W.K.G. Seah, A. Lo, and K. C. Chua, *A Flexible Quality of Service Model for Mobile Ad-Hoc Networks,* IEEE VTC

[76] H, Xiao, K Chua, W Seah, A *Quality of Service Model for Ad Hoc Wireless Networks* Handbook of Ad hoc Wireless Networks

[77] Nikaein, Navid; Bonnet, Christian; Moret, Yan; Rai, Idris A. (France): *2Lqos- Two-Layered Quality-of-Service Model for Routing in Mobile Ad Hoc Networks*

[78] Michael Gerharz, Christian de Waal, Matthias Frank and Paul James (Nokia Research Center Bochum),*A Practical View on Quality-of-Service Support in Wireless Ad Hoc Networks*,Proc. of the 3rd IEEE Workshop on Applications and Services in Wireless Networks (ASWN), pp. 185-196, Berne, Switzerland, July 2003

[79] G.H. Ahn, A. T. Campbell, A. Veres, and L. H. Sun, *SWAN: Service Differentiation in Stateless Wireless Ad Hoc Networks*, IEEE INFOCOM'2002

[80] Ahn G-S, Campbell AT, Veres A, Sun LH. *Supporting service differentiation for real-time and best-effort traffic in stateless wireless ad hoc networks* (SWAN). IEEE Trans. on Mobile Computing, 2002,1(3):192~207.

[81] G.H. Ahn, A. T. Campbell, A. Veres, and L. H. Sun, *SWAN: Service Differentiation in Stateless Wireless Ad Hoc Networks*, IEEE INFOCOM'2002

[82] S. B. Lee, G. S Ahn, and A. T. Campbell. *Improving UDP and TCP performance in mobile ad hoc networks*, IEEE Communication Magazine. June, 2001

[83] S. Lee, A. Gahng-Seop, X. Zhang, A. Campbell, *INSIGNIA: An IP-Based Quality of Service,* "Framework for Mobile Ad Hoc Networks". Journal of Parallel and Distributed Computing (Academic Press), Special issue on Wireless and Mobile Computing and Communications, Vol. 60, Number 4, April, 2000, pp. 374-406.

[84] S. B. Lee, G. S Ahn, and A. T. Campbell. *Improving UDP and TCP performance in mobile ad hoc networks*, IEEE Communication Magazine. June, 2001

[85] Lohier Stephane, Senouci Sidi-Mohammed, Ghamri Doudane Yacine, Pujolle Guy *A reactive QoS Routing Protocol for Ad Hoc Networks*, , European Symposium on Ambient Intelligence (EUSAI'2003), Eindhoven, Netherlands

[86] Y Zhang, *Quality of service for ad hoc on-demand distance vector routing*, MSc thesis, 2004

[87] A. El-Gamal, J. Mammen, B. Prabhakar, and D. Shah, *Throughput-delay trade-off in wireless networks*, in Proc. of IEEE Infocom, March 2004.

[88] Ying Ge *Quality of Service Routing in Ad-Hoc Networks Using OLSR*, Communications Research Centre, Thomas Kunz, Carleton University, Louise Lamont, Communications Research Centre

[89] P. Sinha, R. Sivakumar, and V. Bharghavan. *CEDAR: core extraction distributed ad hoc routing.* In Proc. of IEEE INFOCOMM '99, 1999

[90] S. Chen and K. Nahrstedt, *Distributed Quality-of-Service Routing in Ad Hoc Networks*, IEEE J. Sel. Areas in Comm., vol. 17(8), pp. 1488-1505, Aug. 1999.

[91] Qi Xue, Aura Ganz: *Ad hoc QdS on-demand routing (AQOR) in mobile ad hoc networks*. J. Parallel Distrib. Comput. 63(2): 154-165 (2003)

[92] Levente Buttyan and Jean-Pierre Hubaux, *Enforcing Service Availability in Mobile Ad-Hoc WANs*, 1st IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing (MobiHOC)

[93] Buttyan, L., and Hubaux, J.-P. *Stimulating Cooperation in Self-Organizing Mobile Ad Hoc Networks*. Tech. Rep. DSC

[94] Sheng Zhong, Jiang Chen, and Yang Richard Yang. *Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks*. In Proceedings of IEEE Infocom '03, San Francisco, CA, April 2003.

[95] S. Buchegger and J. Le Boudec, *Performance analysis of the CONFIDANT protocol: cooperation of nodes - fairness in distributed ad-hoc networks*, IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing (MobiHOC), Lausanne, Switzerland, June 2002.

[96] P. Michiardi and R. Molva, *CORE: A COllaborative REputation mechanism to enforce node cooperation in Mobile Ad Hoc Networks*, Communication and Multi-media Security 2002

[97] Miranda, H., Rodrigues, L.: *Friends and foes: Preventing selfishness in open mobile ad hoc networks*. In: Proc. of the First Intl. Workshop on Mobile Distributed Computing (MDC'03), Providence, RI, USA (2003)

[98] http://www.isi.edu/nsnam/ns/

[99] J.L. Sobrinho, A.S. Krishnakumar. *Quality-of-Service in ad hoc carrier sense multi-ple access networks*. IEEE J. Selec. Areas Commun., 17(8):1353--1368, August 1999.

[100] A. E. Gamal, J. Mammen, B. Prabhakar, and D. Shah, *Throughput-Delay Trade-Off in Wireless Networks*, IEEE INFOCOM'04, Mar. 2004

[101] CORE 3.1, *revista do "Computer History Museum"*, Fevereiro de 2002 (disponível em http://www.computerhistory.org/publications/core/3.1/)

[102] S. Crisóstomo, S. Sargento, M. Natkaniec, N. Vicari, *A QoS Architecture Integrating Mobile Ad-Hoc and Infrastructure Networks. I*n Proceedings of the Workshop on Internet Compatible QoS in Ad hoc Wireless Networks (IC-QAWN), co-located with The 3rd ACS/IEEE International Conference on Computer Systems and Applications (AICCSA-05), Cairo, Egypt, January, 2005.

[103] João Paulo Barraca, Susana Sargento, Rui Aguiar, *The Polynomial-assysted Ad-hoc Charging Protocol*, IEEE International Symposion on Computers and Communications - ISCC2005,Cartagena, Spain, 2005 .