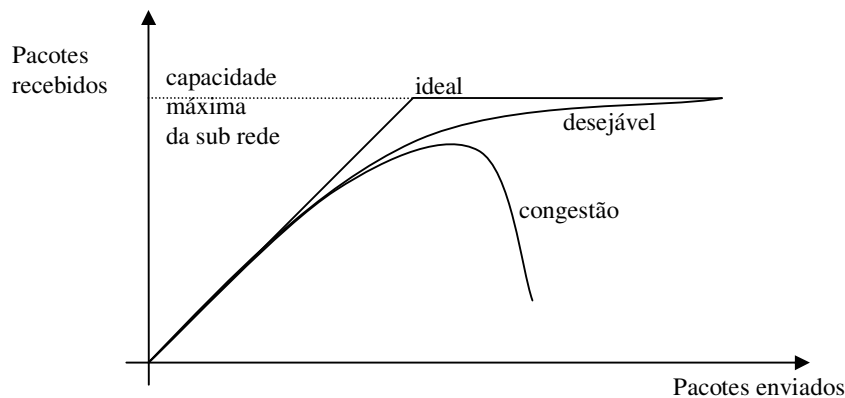


CONGESTÃO

O encaminhar da informação tal como o tráfego numa grande cidade leva a condições de congestionamento quando muitos pacotes estão presentes em sub redes. As redes têm uma determinada capacidade de escoamento e a partir desse valor a comunicação começa a degradar-se. No caso das redes começam-se a perder ou eliminar pacotes. Isto faz com que eles voltem a ser repetidos e chega ao ponto de não ser possível a comunicação.



São diversos os factores que causam a congestão e de difícil solução. Por exemplo se chegarem muitos pacotes de várias entradas para uma mesma saída os pacotes formam uma fila de espera muito grande. Caso não haja memória suficiente, há pacotes que se perderão. Por outro lado, se admitirmos que a memória possa ser infinita faz com que a fila de espera aumente e o tempo de atraso de cada pacote faça com que no equipamento fonte os mecanismos de timeout disparem e haja nova repetição de pacotes. Isto faz com que a fila ainda aumente mais. Tudo isto irá também afectar os nós que estão a jusante.

Outro aspecto que pode provocar o congestionamento é a velocidade de processamento. Mesmo que haja memória e largura de banda suficiente para as saídas, o avolumar de pacotes dá-se neste caso nas entradas.

No caso de haver uma largura de banda insuficiente para o envio dos pacotes estes começam a avolumar-se nas filas de saída.

De certa forma estes aspectos, quantidade de tráfego, memória, velocidade de processamento e largura de banda precisam de estar em equilíbrio, pois qualquer desajuste irá provocar um engarrafamento algures na sub rede.

O controlo de fluxo que possa existir entre o equipamento fonte e o de destino limita a quantidade de informação que transita entre eles controlado pelo destinatário.

O controlo de congestão, que se pretende agora concretizar, tem como objectivo garantir que a sub rede possa transportar o tráfego pretendido. Ou seja, evitar que se entre num estado de congestão total.

Há algoritmos de controlo de congestão que podem provocar o controlo de fluxo como forma de minimizar a quantidade de pacotes a transmitir.

Do ponto de vista do controlo de sistemas podemos encarar o controlo em malha aberta ou malha fechada. Em malha aberta significa que não há qualquer tipo de controlo da entrada em função da actividade da saída. No caso da malha fechada a entrada está condicionada pela influência da saída.

Em malha aberta, o controlo de congestão deve ser evitado utilizando restrições internas sem se saber o estado da rede, e procura-se assim em cada nó ter um bom funcionamento. Decide quando aceitar novamente tráfego, quando eliminar pacotes e quais é que se eliminam, e por onde redireccionar o tráfego.

Em malha fechada é necessário ter alguma medida do desempenho da rede. Assim é necessário monitorizar o sistema para detectar quando e onde é que a congestão ocorreu. Transmitir essa informação para os locais onde se terá de tomar uma decisão. Ajustar o sistema convenientemente de modo a mantê-lo funcional.

Há um conjunto de métricas que podem ser utilizadas para dar uma indicação de congestão da rede, como por exemplo a percentagem de pacotes eliminados por falta de espaço, o tempo médio de atraso das filas de espera, o número de retransmissões devido aos mecanismos de timeout, o tempo médio de trânsito dos pacotes e o desvio padrão desse atraso. Em qualquer um destes casos, o aumento de qualquer um dos valores indica um avanço na direcção da congestão.

Este conjunto de informação necessita de ser enviado através da rede para os locais onde se fazem decisões acerca do tráfego na rede, o que irá aumentar a quantidade de tráfego e aproximar a rede da congestão.

Um nó pode não enviar a informação indicando a quantidade de congestionamento em que está mas pode colocar um bit nos pacotes que envia indicando que está a entrar em congestão.

Também se pode preferir o envio de pacotes que inquiram directamente os nós acerca do seu estado de congestão e assim evitar o congestionamento com o redireccionar da informação.

Um dos aspectos que pode ocorrer na rede é esta entrar em oscilação provocada por parâmetros inadequados. Os nós ora enviam ora deixam de enviar pacotes.

Podemos enquadrar os algoritmos de congestão do seguinte modo:

Malha aberta

- actuam na emissão
- actuam na recepção

Malha fechada

- explicit feedback – realimentação explícita
- implicit feedback – realimentação implícita

A realimentação diz-se explícita no caso de serem enviados pacotes com dados acerca do congestionamento. No caso de ser implícita o estado de congestionamento é inferido pelos pacotes que passam através do nó e que vêm em sentido contrário, como é o caso dos acknowledges. Também temos uma noção deste tipo de congestionamento implícito quando no tráfego de uma cidade olhamos para a faixa contrária e avaliamos o que possa estar a acontecer para o local para onde nos dirigimos.

A congestão dá-se quando a carga é superior aos recursos disponíveis. Nesta situação podemos, aumentar os recursos, diminuir a carga ou actuar em ambos. Aumentar os recursos pode ser através do uso de linhas secundárias, ou colocar em funcionamento equipamento redundante para o caso de avarias.

O aumento da largura de banda também pode ser conseguido utilizando outras saídas do nó, em alternativa ou em paralelo como é o caso do multipath routing.

O aumento dos recursos leva também a um aumento da carga pois a rede ficando mais desimpedida torna-se mais atractiva para o aumento da comunicação. Reduzir a carga é das opções a que reduz mais o congestionamento. Isso consegue-se com a recusa de serviço a alguns utilizadores, ou no degradar da comunicação de alguns utilizadores, ou diferir para outras alturas o tráfego dos utilizadores. Isto consegue-se para situações bem determinadas do ponto de vista de tráfego na rede através de campanhas de utilização da rede, assinaturas fixas, e outros modos de marketing para aliciar os utilizadores para fatias de tempo mais vazias do ponto de vista de tráfego. O modo mais fácil de controlar a carga é através dos circuitos virtuais porque o canal está bem determinado.

Malha aberta

Este tipo de sistemas é desenhado para minimizar a congestão pois é para eles mais difícil agir em situações de congestão ou de degradação da comunicação. Tentam assim evitar sair fora de certas restrições. Para isso utilizam-se políticas de congestionamento a diversos níveis:

Nível	Política
Transporte	Retransmissão Pacotes fora de ordem Confirmações – ack Controlo de fluxo Temporização – timeout
Network	Circuitos virtuais ou datagramas Filas de espera de pacotes Eliminar pacotes Algoritmos de encaminhamento Tempo de vida dos pacotes
Data Link	Retransmissão Sequência das tramas Confirmações – ack Controlo de fluxo

Data Link

A retransmissão causada pelo fim da temporização provoca um aumento de tramas na rede. O algoritmo Go-Back-n nesta situação provoca o envio de mais tramas do que o selective repeat. A utilização de timeouts pequenos pode provocar um aumento de tramas excessivo.

No caso de se perderem tramas na rede e estas tenham de ser reenviadas, o Go-Back-n provoca também mais tramas do que o Selective Repeat porque este tem uma janela de recepção e pode guardar as que chegam fora de ordem enquanto que o Go-Back-n só aceita as tramas pela ordem correcta.

O envio dos acknowledges confirmando a boa recepção da informação aumenta o tráfego na rede. Uma forma de minimizar esse tráfego é o de enviar os acknowledges em piggy-back e esperar a recepção de mais de uma trama para depois enviar apenas um acknowledge caso seja necessário. Nesta situação o Go-Back-n já tem vantagens relativamente ao Selective Repeat que necessita de um acknowledge por cada trama.

A dimensão da janela utilizada para o controlo de fluxo indica o número de tramas que são enviadas sem que se tenha recebido qualquer confirmação do destinatário. Quanto maior for esta janela maior é a sua influência nas causas do congestionamento. A janela pode ter a dimensão 1 e nesse caso estamos perante o send & wait.

Network

Os circuitos virtuais utilizam os mesmos caminhos para enviar os pacotes, por isso os algoritmos de congestão funcionam melhor com este tipo de comunicação do que com o tráfego dos datagramas que são mais fluidos e seguem caminhos distintos.

As filas de espera nos nós podem situar-se nas linhas de entrada ou nas linhas de saída ou em ambas as linhas. O desempenho dos algoritmos de congestão também depende de como são usadas estas filas de espera. Se todos os pacotes são despachados na ordem com que são recebidos ou se há algum tipo de prioridade de modo a fazer com que certos pacotes tenham o menor atraso possível dentro do nó. Isto é importante se a informação necessária aos algoritmos de congestão possa ter prioridade nessas situações.

Quando um nó tem mais pacotes do que aqueles que pode suportar terá de eliminar alguns. O modo como se eliminam determina o desempenho da rede. Há uma grande diversidade no modo de encarar os pacotes, por idade, por tamanho, por prioridade, por importância como é o caso daqueles que têm um bit a dizer que se podem deitar fora caso seja necessário. Uma boa política de eliminação de pacotes pode melhorar o efeito de congestão.

O algoritmo de encaminhamento também necessita de ter uma noção do tráfego das linhas para que possa redireccionar o tráfego de modo a utilizar linhas que estejam a um nível de utilização inferior como é o caso do Multipath Routing.

O prazo de validade dos pacotes também influencia o tráfego. No caso de se utilizar como prazo o número de saltos (Hop) o pacote é eliminado quando esse valor chega a zero. Outro caso é o de conter um tempo que é decrementado pelo tempo que esteve retido num nó. Prazos de validade muito grandes adensam a congestão porque permanecem na rede quando já não seria necessário, e prazos de validade pequenos fazem com que os pacotes desapareçam antes de chegarem ao destino o que provocará retransmissões; ou seja refazer todo o trajecto que já tinha sido conseguido.

Transport

A camada de Transporte funciona como uma Data Link entre os dois pontos terminais da comunicação em que toda a rede é o “meio físico”. Assim tudo o que foi dito para a política de congestão para a Data Link é também válido aqui para esta camada. Contudo a maior dificuldade está em determinar os valores para os timeouts uma vez que o tempo de trânsito entre os dois pontos terminais nunca é o mesmo, é imprevisível, e não é como a ligação ponto a ponto entre dois nós. Se o timeout é demasiado curto provoca retransmissões desnecessárias o que aumenta o estado de congestionamento, no caso de ser demasiado longo já não afecta o estado de congestionamento, até contribui para o reduzir, mas no caso de se perder algum pacote a retransmissão e a recuperação da comunicação é mais lenta; o tempo de resposta diminui.

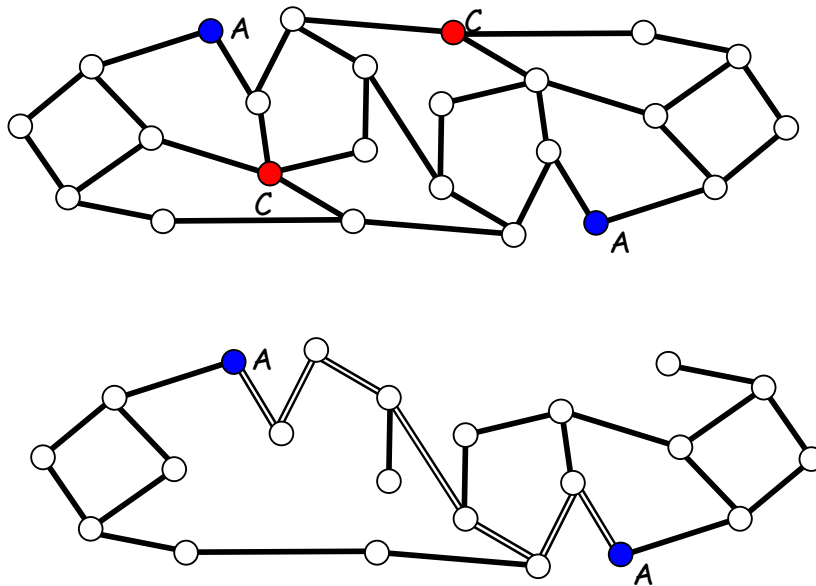
Malha fechada

Este tipo de sistemas é desenhado para agir em situações de congestão ou de degradação da rede e não evitar a congestão como no caso da malha aberta.

Admissão Reservada (Direito de Admissão Reservado)

Um modo de evitar o agravamento da congestão é o de restringir o acesso à criação de mais circuitos virtuais. Assim que se entra no estado de congestão mais nenhum nó permite a criação de qualquer circuito virtual por parte da camada de Transporte até que a congestão se tenha dissipado.

Um outro modo menos radical consiste em determinar exactamente onde se situa o problema da congestão e tentar contornar esses nós e linhas, como se a parte congestionada não existisse. Podemos verificar esta situação como mostra a figura em que os nós A pretendem estabelecer um circuito virtual e os nós C estão em congestão. Retirando os nós da topologia da rede e as suas ligações correspondentes, obtém-se uma nova topologia onde se refazem os percursos de encaminhamento. O caminho que dantes percorreria um dos nós C passa agora a utilizar outro caminho um pouco mais longo mas atravessando nós descongestionados.



Marcar a Reserva

Um modo de evitar a deterioração da ligação em caso de congestão é o de reservar em cada nó os recursos necessários à comunicação quando se utilizam circuitos virtuais. Este tipo de estratégia está mais de acordo com o tipo malha aberta. Mas a reserva dos recursos significa a sua não utilização por parte de outros que o poderiam fazer quando não há comunicação pelo circuito virtual que fez a reserva. Este tipo de contracto pode ser estabelecido em função do volume e ritmo do tráfego e da qualidade de serviço entre outros parâmetros. Nesta situação a congestão não ocorre nos circuitos assim criados. Porém, quando um nó entra em congestão, todos os circuitos virtuais que o atravessam podem sentir essa influência.

Este método de reserva pode ser feito em qualquer situação, o que acaba por desperdiçar recursos, ou então apenas em situações de congestão, em que se garante o mínimo de fluxo. Neste caso, como há uma adaptação, considera-se este tipo de estratégia como de malha fechada.

Datagramas

O uso de datagramas torna a rede imprevisível na repercussão da congestão por causa da sua fluidez. Embora o que se apresenta de seguida também possa ser feito para os circuitos virtuais as estratégias dirigem-se essencialmente para o uso de datagramas.

Utiliza-se um valor que determine o uso de uma determinada linha. O seu valor instantâneo apenas dá uma indicação pontual, e não durante o tempo. Se a utilização se mantiver durante muito tempo no máximo pode ser indicação de uma possível congestão. Para aferir melhor a condição de utilização da linha utiliza-se um facto que determine a influência passada da linha.

$$uso_{actual} = peso * uso_{antigo} + (1-peso)* valor_instantâneo$$

O peso varia entre 0 e 1, e o valor_instântaneo entre 0 e 100%. O uso_{actual} encontra-se entre 0 e 100%. Colocando um limite por exemplo a 90% de utilização, indica que acima desse valor entra-se em congestão. No caso de uma linha se manter por exemplo a 10% durante muito tempo e depois

instantaneamente vir a 100% e depois voltar a 10%, não afecta em nada o estado de congestão.

Quando se entra em estado de congestão, podem utilizar-se algumas das seguintes acções

Bit de Aviso

O equipamento terminal ao enviar os pacotes pode colocar alguns nós da rede em congestão. Para diminuir o ritmo de envio é necessário avisar esse equipamento do estado da rede. Um modo de o fazer é o de colocar um bit de aviso dessa condição de congestionamento nos pacotes de acknowledge que vêm em sentido contrário. Para isso cada nó congestionado activa o bit de aviso de congestão em cada pacote que ele encaminha, seja de acknowledge ou não. Como os datagramas podem passar por caminhos diferentes, é da responsabilidade do nó terminal de destino de copiar esse bit nos pacotes de acknowledge, e desse modo garantir que o equipamento fonte reduz o fluxo.

Os nós apenas activa o bit, e não o desactivam, de modo que se garante que a condição de congestão só é dissipada quando nenhum nó por onde passam os datagramas esteja congestionado.

Pacotes Choke

O método anterior usa uma forma indirecta de avisar o equipamento terminal fonte de que está a congestionar a rede. Outro método é o de o avisar directamente utilizando um pacote Choke (de sufoco, falta de ar). Isto é fácil porque o datagrama contém o endereço fonte. Para evitar que o datagrama ao passar por diversos nós cause um grande número de pacotes Choke a serem transmitidos, é colocado no cabeçalho do datagrama um bit a indicar que o Choke já foi enviado. Como cada datagrama diferente irá também provocar o envio de outro Choke, já terá de ser o equipamento terminal fonte a resolver essa situação. Assim, ao receber o primeiro Choke, diminui o fluxo de um determinado valor, e depois ignora os Chokes que vierem de seguida dentro de um intervalo de tempo fixo. Se depois desse intervalo de tempo ainda receber outro Choke, reduz ainda mais o fluxo e age do modo anterior.

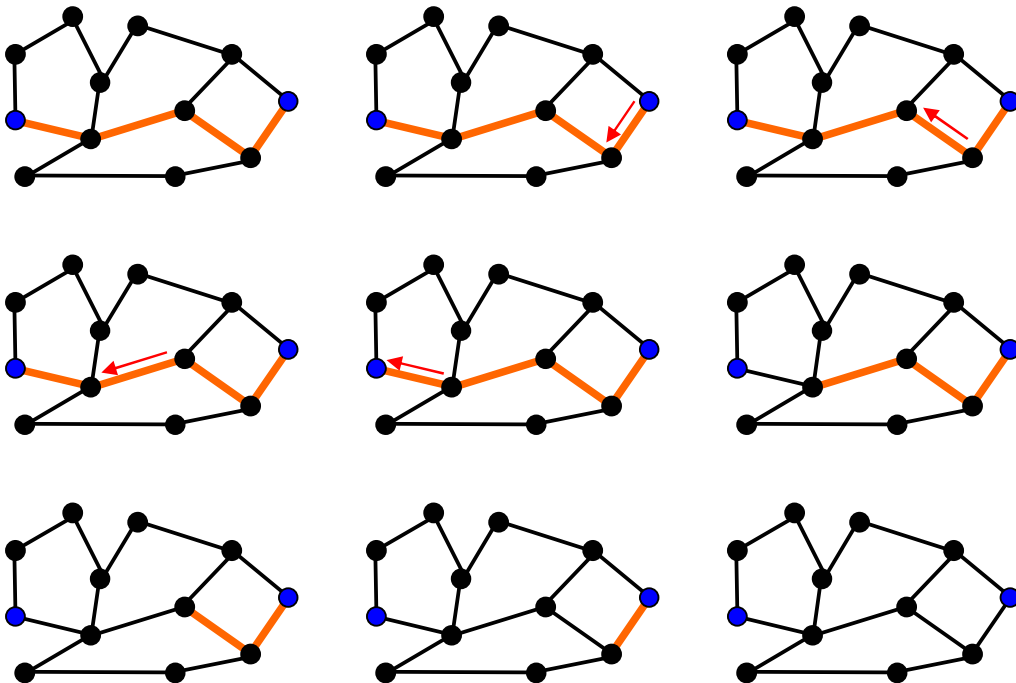
No caso de não aparecerem mais pacotes de Choke, durante um intervalo próprio de escuta, o equipamento terminal fonte aumenta o fluxo.

Para evitar uma oscilação no equipamento fonte de diminuir, aumentar, diminuir, ... o fluxo, pode utilizar-se uma taxa diferente, por exemplo, ao reduzir fá-lo sempre para metade do que estava, mas para aumentar aumenta apenas um quarto ou um quinto, até chegar a um valor óptimo.

O equipamento pode implementar qualquer outro tipo de estratégia a utilizar.

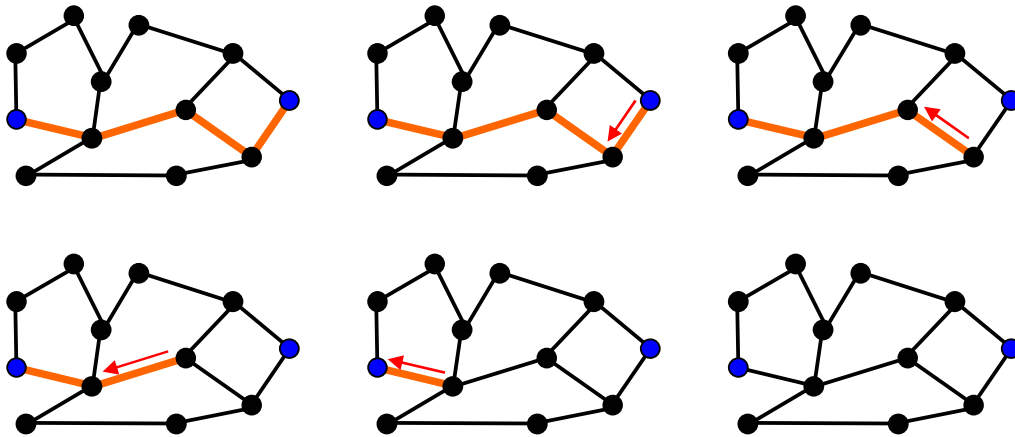
Pacotes Choke de Hop em Hop

Utilizando a técnica dos pacotes Choke consegue-se reduzir o congestionamento contudo a rede demora a reagir porque a informação de que há congestão necessita de ir até ao equipamento terminal e depois deste diminuir a quantidade de tráfego, ainda demora que esta diminuição seja propagada por toda a rede. Assim ainda se mantém a congestão no nó, apenas causado por um equipamento terminal, no dobro do tempo do nó a esse equipamento.



Outra forma de reduzir o tráfego de um modo mais significativo é a interpretação dos pacotes Choke por parte dos próprios nós, que para além de o transmitirem no sentido contrário ao do fluxo dos pacotes que causam o

congestionamento, também eles mesmos reduzem a quantidade de tráfego por essas saídas.



Este método como que empurra a congestão em direcção ao nó terminal que está a congestionar a rede. Em princípio não se perdem pacotes porque ficam guardados nas filas de espera dos diversos nós e que vão escoando a carga lentamente.

Ceifa de pacotes

No caso da congestão se manter chega o momento de se ter de eliminar pacotes. O modo como se escolhem os pacotes a eliminar tem repercussões no modo como toda a rede reagirá porque embora esses pacotes desapareçam, mais cedo ou mais tarde eles terão de ser repetidos pelos equipamentos terminais, e se a rede alivia um pouco, de seguida entrará novamente em congestão, e desse modo oscilará entre estes dois estados. No caso da transmissão de um ficheiro é preferível assegurar os pacotes mais antigos do que os recentes porque tendo de haver repetições, minimizam-se (no caso do Go-Back-n) se apenas se perderem os últimos pacotes, pois as repetições começam sempre a partir do pacote perdido mais antigo. No caso de se estar a transmitir um sinal de multimédia, é preferível desperdiçar os pacotes mais antigos e garantir a recepção dos mais recentes, porque a repetição pode não ser necessária. Estas duas estratégias são conhecidas como vinho e leite, a primeira porque promove a mais antiga, e a segunda porque promove os pacotes mais recentes.

Há outro tipo de pacotes, por exemplo na transmissão de vídeo diferencial, é enviado um pacote com valores absolutos e de seguida pacotes de tamanho inferior que apenas contêm a diferença para esse absoluto. Neste caso, os pacotes diferenciais de nada valem sem aquele que tomam como referência. É assim preferível manter um pacote grande como o de valores absolutos e eliminar os que contêm a informação diferencial.

Outro caso, por exemplo na transmissão de texto com imagem, é preferível perder um pacote de dados da imagem, que se traduzirá numa pequena linha, por exemplo, do que um pacote com uma linha de texto.

De modo a ajudar os algoritmos a tornarem mais eficiente a recuperação da congestão é necessário que os equipamentos terminais marquem os seus pacotes com o tipo de conteúdo ou com um valor de importância para que possam ser rejeitados ou não em situações extremas.

Como é difícil garantir que as aplicações que correm nos equipamentos terminais o façam, é necessário colocar incentivos como o preço da ligação sobre qualidades de serviço distintas. Ou cobrar apenas o que será de excesso sobre o valor acordado nos contratos. Ou por exemplo admitir que todo o tráfego excedente seja encarado com uma prioridade muito baixa e por isso passível de ser perdida.

Randomly Early Detection

Como é preferível prevenir do que remediar, o mesmo se atribui no caso das redes. O algoritmo Randomly Early Detection (RED), procura libertar pacotes antes de se ficar sem espaço nos nós, evitando a congestão. Há algoritmos que quando perdem pacotes assumem que isso é devido ao tráfego e não ao erro nas linhas, e o TCP/IP é um desses. Como o TCP/IP foi desenhado para usar linhas fiáveis como é o caso das redes, os pacotes que se perdem são à partida devido a problemas de falta de espaço, e isso pode ser usado para aliviar a congestão.

Ao evitar a sobrecarga da memória, este algoritmo consegue atrasar a entrada em congestão permitindo assim que ela possa ser resolvida. Os pacotes começam a ser eliminados a partir de um determinado tamanho médio das filas de espera. Como não se sabe qual o equipamento terminal fonte que está a provocar o congestionamento, retira-se um pacote aleatoriamente da fila

de espera que está maior. É de esperar que a probabilidade desse pacote pertencer a um equipamento terminal muito obsessivo é maior do que a de um outro com menos tráfego.

Este método apenas faz isso e não envia qualquer pacote de Choke pois teria de utilizar largura de banda para o fazer. Actua por isso no pressuposto de que o equipamento terminal irá por sua iniciativa reduzir o fluxo de pacotes.

Jitter Control

Há tipos de comunicação em que mais do que se perder um pacote, o importante é o de manter um tempo constante de atraso dentro da rede. A flutuação desse tempo de atraso, de trânsito no circuito virtual, tem o nome de jitter. No caso de se estar a transmitir vídeo ou áudio, é imprescindível que o tempo de trânsito se situe dentro dos limites pré-determinados. Assim os pacotes podem ser atrasados caso necessário ou então colocados prioritariamente nas filas de saída. Deste modo atrasam-se os pacotes que vão adiantados e despacham-se os que estão atrasados. Para isso é adicionado ao pacote informação correspondente dos tempos de trânsito que tem e o do esperado nesse circuito virtual.

Também se pode utilizar uma bufferização adicional na saída de modo a servir de tampão à comunicação e assim aumentar artificialmente o tempo de trânsito da rede. Mas no caso de vídeo-conferência ou outra interacção, esta bufferização pode ser prejudicial.