



República Federativa do Brasil
Ministério da Economia
Instituto Nacional da Propriedade Industrial

(11) PI 0617705-0 B1



(22) Data do Depósito: 05/10/2006

(45) Data de Concessão: 16/07/2019

(54) Título: CONTROLE DE BANDA ADAPTATIVO

(51) Int.Cl.: H04L 12/911; H04L 12/26; H04L 12/851; H04L 12/807; H04L 12/841.

(52) CPC: H04L 47/821; H04L 43/0864; H04L 47/12; H04L 47/2433; H04L 47/27; (...).

(30) Prioridade Unionista: 21/10/2005 US 11/256,259.

(73) Titular(es): INTERNATIONAL BUSINESS MACHINES CORPORATION.

(72) Inventor(es): STEVEN ELLIOTT; CHRISTOPHER VICTOR LAZZARO; THANH TRAN.

(86) Pedido PCT: PCT EP2006067094 de 05/10/2006

(87) Publicação PCT: WO 2007/045561 de 26/04/2007

(85) Data do Início da Fase Nacional: 22/04/2008

(57) Resumo: CONTROLE DE BANDA ADAPTATIVO Um método implementado por computador, um aparelho e código de programa utilizável em computador para receber dados de uma fonte (302) a uma pluralidade de gateways (304, 306, 308) para distribuição usando uma prioridade selecionada. Os dados são transmitidos a partir da pluralidade de gateways para uma pluralidade de receptores (310, 320, 330), utilizando a prioridade selecionada. Cada gateway na pluralidade de gateways tem um processo de controle de banda adaptativo e um respectivo conjunto de parâmetros para controlar o processo de controle de banda adaptativo para o envio de dados na prioridade selecionada. A transmissão dos dados a partir de cada gateway para a prioridade selecionada tem um impacto diferente em outro tráfego em diferentes gateways na pluralidade gateways para a prioridade selecionada quando diferentes valores são fixados para o conjunto de parâmetros para os diferentes gateways.

Relatório Descritivo da Patente de Invenção para: "**CONTROLE DE BANDA ADAPTATIVO**".

Referência Cruzada a Pedidos Relacionados

A presente invenção está relacionada com os seguintes 5 pedidos de patente com título: Método e Aparelho para o controle de largura de banda adaptativo com Configurações de usuário. Método e aparelho para controle de banda adaptável com controle de largura de banda atribuído ao mesmo cessionário, e aqui incorporada por referência.

10 Antecedentes Da Invenção

Campo da Invenção

A presente invenção refere-se genericamente a um sistema aperfeiçoado para processamento de dados e, em particular, a um método implementado por computador e 15 aparelho para a transferência de dados. Ainda mais particularmente, a presente invenção refere-se a um método implementado por computador, aparelho, e código de programa utilizável por computador para adaptativamente controlar a largura de banda utilizada para transferir dados.

20 Descrição da Técnica Relacionada

Com o uso comum das redes e da Internet, as comunicações no comércio foram revolucionada. Redes são comumente usadas para transferir dados. Muitas aplicações distribuídas fazem uso de transferências de grande plano de

fundo para melhorar a qualidade do serviço. Com estes tipos de transferências de plano de fundo, os usuários não são obrigados a esperar a conclusão destas transferências antes de executar outras ações. Uma ampla gama de aplicações e

5 serviços, incluindo, por exemplo, backup de dados, prefetching, distribuição de dados de empresa, distribuição de conteúdo de Internet, e armazenamento ponto a ponto empregam transferências de plano de fundo. Estes e outros tipos de aplicativos aumentam o consumo de banda de rede.

10 Alguns destes serviços têm demandas de largura de banda potencialmente ilimitadas, em que o uso da largura de banda de forma incremental mais presta serviço progressivamente melhor.

Um problema com estes tipos de aplicações é que a

15 maioria das redes tem apenas uma quantidade limitada de largura de banda disponível para a transferência de dados. Algumas aplicações realizam funções críticas, enquanto outras são não-críticas.

Normalmente, as transferências de plano de fundo são

20 não-críticas e podem usar toda a banda disponível abrandar o tráfego de rede crítico.

Controles de largura de banda adaptativos têm sido empregados para se adaptar automaticamente às condições da rede para reduzir o impacto à rede. Atualmente, diferentes

processos de largura de banda adaptativos e algoritmos de controle são utilizados para controlar a quantidade de largura de banda usada por aplicativos diferentes para evitar o congestionamento. Atualmente disponíveis os 5 processos de controle adaptativo de largura de banda levam em conta as condições da rede ao nível da interface local, seja no servidor ou computador cliente e são incapazes de levar em conta as condições outras que possam existir na transferência de dados.

10 Portanto, seria vantajoso ter um método implementado por computador melhorado, aparelho, e código de programa de computador utilizável para controlar o uso de largura de banda adaptativamente na transferência de dados.

15 Sumário da Invenção

A presente invenção proporciona um método implementado por computador, aparelho, e código de programa de computador utilizável para receber dados a partir de uma fonte de uma pluralidade de passagens para distribuição 20 usando uma prioridade selecionada. Os dados são transmitidos a partir da pluralidade de portas de entrada para uma pluralidade de receptores que utilizam a prioridade selecionada. Cada gateway na pluralidade de gateways tem um processo de controle adaptativo de largura

de banda e um respectivo conjunto de parâmetros para controlar o processo de controle adaptativo da largura de banda para enviar os dados na prioridade selecionada. A transmissão dos dados de cada gateway para a prioridade selecionada tem um impacto diferente sobre outro tráfego em gateways diferentes na pluralidade de gateways para a prioridade selecionada, quando os valores são estabelecidos para o conjunto de parâmetros para gateways diferentes.

O conjunto de parâmetros pode compreender um limiar, beta, e um tempo de espera máximo. O conjunto de parâmetros pode ser configurado para um gateway particular na pluralidade de gateways, em que o conjunto de parâmetros para a gateway particular pode ser diferente do conjunto de parâmetros para um outro gateway na pluralidade de gateways. O processo de controle adaptativo de banda leva em conta o congestionamento ao longo de um caminho de um gateway na pluralidade de gateways para um receptor da distribuição. Além disso, o respectivo conjunto de parâmetros pode ser diferentes para diferentes gateways na pluralidade de gateways. Nos exemplos ilustrativos, os dados compreendem um arquivo de dados, uma atualização do aplicativo, e um patch de vírus.

Breve Descrição dos Desenhos

A concretização da invenção será agora descrita, a título de exemplo apenas, e com referência aos desenhos anexos, em que:

A Figura 1 é uma representação pictórica de uma rede de sistemas de processamento de dados em que os aspectos da presente invenção podem ser aplicados;

A Figura 2 é um diagrama de blocos de um sistema de processamento de dados em que os aspectos da presente invenção pode ser implementado;

A Figura 3 é um diagrama que ilustra um exemplo de sistema de processamento de dados em rede em que as distribuições podem ser feitas utilizando diferentes prioridades de acordo com uma concretização ilustrativa da presente invenção;

A Figura 4 é um diagrama que ilustra os componentes utilizados no controle de largura de banda adaptativa, em conformidade com uma concretização ilustrativa da presente invenção;

A Figura 5 é um diagrama de um controle de transmissão / Protocolo de Internet (TCP / IP) e protocolos similares, em conformidade com uma concretização ilustrativa da presente invenção;

A Figura 6 é um diagrama que ilustra processos de software e componentes utilizados no fornecimento de

controle de banda adaptativa, em conformidade com uma concretização ilustrativa da presente invenção;

A Figura 7 é um diagrama que ilustra uma rede para o envio de uma distribuição de um remetente adaptável para 5 receptores que utilizam prioridades de acordo com uma concretização ilustrativa da presente invenção;

A Figura 8 é um diagrama que ilustra o uso de banda de uma rede de acordo com uma concretização ilustrativa da presente invenção;

10 A Figura 9 é um fluxograma de um processo para a configuração de definições de usuário para os parâmetros de acordo com uma concretização ilustrativa da presente invenção;

15 A Figura 10 é um fluxograma de um processo para um thread de envio de acordo com uma concretização ilustrativa da presente invenção;

A Figura 11 é um fluxograma de um processo para um pacote *sniffing* a tread de acordo com uma concretização ilustrativa da presente invenção;

20 A Figura 12 é um fluxograma de um processo para um thread de controle de congestionamento, de acordo com uma concretização ilustrativa da presente invenção;

A Figura 13 é um fluxograma de um processo para a realização de controle de banda adaptativa, em conformidade com uma concretização ilustrativa da presente invenção; e

A Figura 14 é um fluxograma de um processo para 5 concretizar o envio de distribuições para os receptores para um cliente de acordo com uma concretização ilustrativa da presente invenção.

Descrição Pormenorizada da Concretização Preferida

As Figuras 1-2 são fornecidas como diagramas 10 exemplares de ambientes de processamento de dados em que concretizações da presente invenção podem ser implementadas. Deve ser apreciado que as Figuras 1-2 são apenas exemplares e não se destinam a assegurar ou implicar qualquer limitação em relação aos ambientes em que aspectos 15 ou concretizações da presente invenção podem ser implementados. Muitas modificações para os ambientes descritos podem ser feitas sem se afastar do espírito e do escopo da presente invenção.

Com referência agora às figuras, a Figura 1 mostra uma 20 representação pictórica de uma rede de sistemas de processamento de dados, em que os aspectos da presente invenção podem ser aplicados. O sistema de processamento de dados em rede 100 é uma rede de computadores em que as concretizações da presente invenção podem ser

implementadas. O sistema de processamento de dados em rede 100 contém a rede 102, que é o meio utilizado para proporcionar links de comunicações entre os vários dispositivos e computadores ligados entre si dentro sistema 5 de processamento de dados em rede 100. A rede 102 pode incluir conexões, tais como fios, links de comunicação sem fio, ou cabos de fibra óptica.

No exemplo representado, o servidor 104 e servidor 106 conectam-se à rede 102, juntamente com unidade de 10 armazenamento 108. Além disso, os clientes 110, 112 e 114 conectam-se à rede 102. Estes clientes 110, 112 e 114 podem ser, por exemplo, computadores pessoais ou computadores de rede. No exemplo descrito, o servidor 104 fornece dados, como arquivos de inicialização, imagens do sistema 15 operacional e aplicativos para clientes 110, 112 e 114. Os clientes 110, 112, e 114 são clientes para o servidor 104 neste exemplo. O Sistema de processamento de dados em rede 100 pode incluir servidores adicionais, clientes e outros dispositivos não mostrados.

20 No exemplo descrito, o sistema de processamento de dados em rede 100 é a Internet com a rede 102 que representa uma coleção mundial de redes e gateways que usam suítes de protocolos de *Transmission Control Protocol / Internet Protocol* (TCP / IP) para se comunicar uns com os

outros. No coração da Internet está um backbone de linhas de comunicação de dados de alta velocidade entre nós principais ou computadores host, consistindo em milhares de sistemas de computador governamental, comercial, 5 educacional e outros sistemas de computador que roteiam dados e mensagens. Claro que, sistema de processamento de dados em rede 100 também pode ser implementado como um número de diferentes tipos de redes, tais como por exemplo, uma Intranet, uma rede de área local (LAN), ou uma rede de 10 longa distância (WAN). A Figura 1 destina-se como um exemplo, e não como uma limitação da arquitetura para diferentes concretizações da presente invenção.

Com referência agora à Figura 2, um diagrama de blocos de um sistema de processamento de dados é mostrado na qual 15 os aspectos da presente invenção podem ser implementados. O sistema de processamento de dados 200 é um exemplo de um computador, tal como servidor 104 ou cliente 110 na Figura 1, no qual o código utilizável por computador ou instruções de execução dos processos das concretizações da presente 20 invenção podem estar localizados.

No exemplo descrito, o sistema de processamento de dados 200 emprega uma arquitetura de hub incluindo hub controlador de memória e ponte norte (MCH) 202 e hub controlador de E/S e ponte sul (ICH) 204. A unidade de

processamento 206, a memória principal 208 e o processador gráfico 210 estão ligados ao hub controlador de memória e ponte norte 202. O processador gráfico 210 pode ser conectado ao hub controlador de memória e ponte norte 202 5 através de uma porta aceleradora gráfica (AGP).

No exemplo descrito, o adaptador de rede local (LAN) 212 conecta a hub controlador de E/S e ponte sul 204. O adaptador de áudio 216, adaptador de teclado e mouse 220, modem 222, memória apenas para leitura (ROM) 224, disco 10 rígido (HDD) 226, drive de CD-ROM 230, barramento serial universal (USB) e outras portas de comunicação 232 e dispositivos PCI / PCIe 234 para se conectar ao hub controlador de E/S e ponte sul 204 através do barramento 238 e barramento 240. Os dispositivos PCI / PCIe podem 15 incluir, por exemplo, adaptadores de Ethernet, placas de expansão e placas de PC para computadores portáteis. PCI utiliza um controlador de barramento da placa, enquanto PCIe não. A ROM 224 pode ser, por exemplo, um sistema de entrada / saída binário flash (BIOS).

20 A Unidade de disco rígido 226 e o CD-ROM 230 conectam a hub controlador de E/S e ponte sul 204 através de barramento 240. A unidade de disco rígido 226 e CD-ROM 230 pode utilizar, por exemplo, um controlador eletrônico integrado (IDE) ou interface de fixação de tecnologia

avanhada serial (SATA). O dispositivo de E/S Super (SIO) 236 pode ser ligado ao hub controlador de E/S e ponte sul 204.

Um sistema operacional é executado em unidade de processamento 206 e coordena e proporciona um controle de vários componentes de dados dentro do sistema de processamento 200 na Figura 2. Como cliente, o sistema operacional pode ser um sistema operacional disponível comercialmente, como o Microsoft® do Windows® XP (Microsoft 10 e Windows são marcas registradas da Microsoft Corporation nos Estados Unidos, outros países, ou ambos).

Um sistema de programação orientada a objetos, tal como o sistema de programação Java™, pode ser executado em conjunto com o sistema operacional e oferece chamadas para o sistema operacional a partir de programas Java ou aplicações em execução em sistema de processamento de dados 200 (Java é uma marca comercial da Sun Microsystems, Inc. nos Estados Unidos, em outros países, ou ambos).

Como um servidor, sistema de processamento de dados 200 pode ser, por exemplo, um IBM eServer™ pSeries ® sistema de computador, rodando o sistema operacional Advanced interactive Executive (AIX ®) ou sistema operacional LINUX (eServer, pSeries e AIX são marcas registradas da International

Business Machines Corporation nos Estados Unidos, outros países, ou ambos, enquanto o Linux é marca registrada da Linus Torvalds nos Estados Unidos, outros países, ou ambos). Sistema de processamento de dados 200 podem ser um sistema de multiprocessador simétrico (PMS), incluindo uma pluralidade de processadores em unidade de processamento 206.

Alternativamente, um sistema de processador único pode ser empregado.

10 Instruções para o sistema operacional, o sistema de programação orientada a objetos, e aplicações ou programas estão localizados em dispositivos de armazenamento, tais como unidade de disco rígido 226, e podem ser carregados na memória principal 208 para a execução pela unidade de processamento 206. Os processos para concretizações da 15 presente invenção são realizados por unidade de processamento 206 usando o código de programa utilizável por computador, o qual pode estar localizado numa memória, tal como, por exemplo, a memória principal 208, memória só de leitura 224, ou em um ou mais dispositivos periféricos 20 226 e 230.

Aqueles de conhecimento ordinário na técnica irão apreciar que o hardware nas Figuras 1-2 pode variar dependendo da aplicação. Outro hardware interno ou

dispositivos periféricos, tais como a memória flash, equivalente à memória não volátil, ou unidades de disco óptico e semelhantes, podem ser utilizadas além ou no lugar do hardware representado nas Figuras 1-2. Além disso, os 5 processos da presente invenção podem ser aplicados a um sistema de processamento de dados multiprocessador.

Em alguns exemplos ilustrativos, o sistema de processamento de dados 200 pode ser um assistente pessoal digital (PDA), que é configurado com memória flash para fornecer memória não volátil para armazenar arquivos do sistema operacional e / ou dados gerados pelo usuário. 10

Um sistema de barramento pode ser composto de um ou mais barramentos, como o barramento 238 ou o barramento 240 como mostrado na Figura 2. Claro que o sistema de barramento pode ser implementado usando qualquer tipo de tecido ou arquitetura de comunicações que prevê uma transferência de dados entre diferentes componentes ou dispositivos conectados ao tecido ou arquitetura. Uma unidade de comunicações pode incluir um ou mais dispositivos utilizados para transmitir e receber dados, 15 tais como modem 222 ou placa de rede 212 da Figura 2. Uma memória pode ser, por exemplo, a memória principal 208, memória só de leitura 224, ou um cache tal como encontrado no hub controlador de memória e ponte norte 202 na Figura 20

2. Os exemplos descritos nas Figuras 1-2 e os exemplos acima descritos não se destinam a implica limitações arquitetônicas. Por exemplo, os sistema de processamento de dados 200 também podem ser um computador tablet, o 5 computador portátil, ou dispositivo de telefone além de tomar a forma de uma PDA.

A presente invenção proporciona um método implementado por computador, aparelho, e código de programa utilizável por computador para adaptativamente controlar a largura de 10 banda utilizada na transferência de dados. Os aspectos da presente invenção reconhecem que os processos de controle adaptativo de largura de banda atualmente utilizados são incapazes de levar em conta as condições de rede que não fazem parte do cliente. Os aspectos da presente invenção 15 reconhecem que as redes diferentes têm diferentes características, que podem afetar a transferência de dados. Os aspectos da presente invenção proporcionam diferentes mecanismos para tendo estes tipos de fatores em consideração. Os aspectos da presente invenção proporcionam 20 uma capacidade de definir os parâmetros para controlar como pacotes de dados individuais são enviados em uma rede para utilização no controle de largura de banda adaptativo.

Os aspectos da presente invenção reconhecem que os sistemas de transferência de dados disponíveis atualmente

tendem a correr muito devagar nos modos de plano de fundo e pode demorar muito tempo para ser concluído. Os aspectos da presente invenção também reconhecem que existem muitos casos em que os clientes desejam funcionalidade de adaptação e não querem que distribuições levem muito tempo, em certos casos. Por exemplo, uma distribuição de patches ou atualizações de vírus é considerada muito importante com relação a uma atualização de um aplicativo. Assim, os aspectos da presente invenção proporcionam uma capacidade de definir uma prioridade numa base por distribuição.

Nestes exemplos ilustrativos, três diferentes prioridades são definidas: alta, média e baixa. A prioridade nestes exemplos é uma prioridade adaptativa na qual um eixo prioritário selecionado, como alta, muda a forma como um processo de controle adaptativo de banda se comporta. Por exemplo, uma atualização de software pode ser enviada como uma prioridade baixa, seguido por um patch de vírus em uma prioridade alta. Como resultado, as diferentes distribuições podem ser dadas prioridades diferentes com base na importância destas distribuições. As distribuições com as prioridades diferentes proporcionam um conjunto diferente de parâmetros para o processo adaptativo de controle de largura de banda para alterar o comportamento

deste processo dependendo da prioridade particular selecionada.

Além disso, a configuração das prioridades diferentes também pode ser fixada numa base por passagem. Nestes 5 exemplos, um gateway é um sistema de processamento de dados ou dispositivo que serve como um portal para um conjunto de dispositivos. Por exemplo, um gateway pode servir como um portal ou entrada de uma rede de área local ou uma rede de longa distância. Além disso, um gateway também pode servir 10 como uma ligação a uma rede sem fio. O gateway também é referido como gerenciando um conjunto de clientes.

Embora estes exemplos ilustrativos são dirigidos para o controle de largura de banda adaptativa, os aspectos da presente invenção podem ser aplicados a qualquer tipo de 15 transferência de dados em massa a um ou mais sistemas de processamento de dados alvo.

Voltando agora à Figura 3, um diagrama que ilustra um exemplo de sistema de processamento de dados em rede em que as distribuições podem ser feitas utilizando diferentes 20 prioridades é descrito de acordo com uma concretização ilustrativa da presente invenção. Neste exemplo, o sistema de processamento de dados em rede 300 contém fonte 302, que está ligado a gateways 304, 306, e 308. O Gateway 304 proporciona uma ligação para os clientes 310, 312, 314, e

316 por meio do roteador 318. O gateway 306 proporciona uma ligação para os clientes 320, 322, 324, 326 e através do roteador de satélite 328. O gateway 308 proporciona uma ligação para os clientes 330, 332, 334, e 336. Nestes 5 exemplos, esses clientes estão viajando com computadores portáteis que se conectam ao gateway 308 através de vários meios, como uma conexão sem fio, uma conexão dial-up, um modem a cabo ou algum sistema de conexão semelhante.

A rede 300 é um exemplo de uma rede contida dentro do 10 sistema de processamento de dados em rede 100 na Figura 1. Em particular, os diferentes clientes e passagens pode ser implementados utilizando um sistema de processamento de dados semelhante ao sistema de processamento de dados 200 na Figura 2.

15 Neste exemplo, o gateway 304 gerencia servidores altamente seguros da rede de área local. Em outras palavras, os clientes 310, 312, 314, e 316 são sistemas de servidores de processamento de dados. Gateway 306 gerencia ponto-de-venda de sistemas localizados em filiais 20 diferentes. O gateway 308 é usado para gerenciar os clientes na forma de laptops que viajam.

O administrador 338 pode enviar uma distribuição usando configurações diferentes. Nestes exemplos, uma distribuição é o envio de qualquer tipo de dados a um ou

mais pontos finais ou receptores. Por exemplo, a distribuição pode ser uma atualização para um aplicativo, uma biblioteca de vínculo dinâmico de atualização, um patch ou atualização de definição de vírus, ou um arquivo de dados. Neste exemplo ilustrativo, a distribuição pode ser enviada através de um dos três níveis de prioridade, alta, média ou baixa. Claro que, outros números de níveis de prioridade podem ser implementados, dependendo da aplicação particular. Três diferentes níveis de prioridade são empregues para fins de ilustração de uma concretização da presente invenção.

Como resultado, o administrador 338 pode enviar uma distribuição para gateways 304, 306 e 308 para distribuição aos clientes com uma prioridade, como alta. O administrador 338 pode então enviar uma subsequente distribuição de gateways 304, 306 e 308 com uma prioridade diferente, tal como o meio. Desta forma, diferentes distribuições podem ser dadas prioridades diferentes para transferência para os receptores com base na importância associada com uma distribuição particular. Como resultado, as distribuições mais importantes podem atingir clientes mais rapidamente, embora este tipo de distribuição usa mais largura de banda. Com uma distribuição de baixa prioridade, menor largura de

banda é utilizada com esse tipo de distribuição tendo mais tempo para distribuir.

Além disso, os aspectos da presente invenção proporcionam uma capacidade de uma prioridade selecionada 5 para utilizar quantidades diferentes de largura de banda em gateways diferentes para o nível de prioridade selecionada. Em outras palavras, para um nível de prioridade selecionada, duas passagens podem utilizar quantidades diferentes de largura de banda para transferir dados dadas 10 as mesmas condições de rede. Os diferentes efeitos de um nível de prioridade selecionada são ajustados em uma base por gateway nestes exemplos ilustrativos, ajustando os parâmetros do processo de controle adaptativo da largura de banda em execução sobre os gateways para enviar a 15 distribuição para os diferentes clientes.

Por exemplo, o gateway 304 pode utilizar setenta por cento da largura de banda ao enviar uma distribuição de alta prioridade para os clientes 310, 312, 314, e 316. O gateway 306 podem utilizar 50 por cento da largura de banda na transferência da mesma distribuição com uma alta prioridade para os clientes 320, 322, 324, e 326. No gateway 308, a mesma distribuição que está sendo enviada em uma prioridade alta só pode utilizar até 30 por cento da

largura de banda ao enviar a mesma distribuição para os clientes 330, 332, 334 e 336.

O impacto real sobre o uso da banda dentro de uma rede particular, acessada por um gateway é ajustado pelo ajuste 5 dos parâmetros no processo adaptativo de controle de largura de banda em execução sobre os gateways. Cada uma destes gateways pode ser pré-configurado quando os gateways são inicialmente definidos. Além disso, estes parâmetros podem ser alterados com base em mudanças na rede ou outras 10 alterações identificadas pelo administrador 338. Estas alterações podem ser administradas através de configurações definidas pelo usuário, como discutido abaixo.

Voltando agora à Figura 4, um diagrama que ilustra os componentes utilizados no controle de largura de banda adaptativo está representado de acordo com uma concretização ilustrativa da presente invenção. Neste exemplo, o remetente adaptativo 400 envia os dados para os receptores 402, 404, e 406. Neste exemplo ilustrativo, remetente adaptativo 400 pode ser um gateway, tal como 20 gateway 304 na Figura 3. Em particular, o remetente adaptativo 400 pode ser implementado como servidor 106 na Figura 1, utilizando hardware tal como o encontrado no sistema de processamento de dados 200 da Figura 2. Os receptores 402, 404, e 406 podem ser clientes, tais como os

clientes 310, 312, e 314 na Figura 3. Estes receptores também podem ser implementados utilizando um sistema de processamento de dados, como sistema de processamento de dados 200 da Figura 2. Os dados são enviados através do envio de pacotes 408 para um mecanismo de encaminhamento, tal como roteador 410. O roteador 410 é um dispositivo que serve para rotear ou enviar pacotes 408 para os receptores apropriados com base no encaminhamento de dados encontrados em pacotes 408. Quando o roteador 410 é forçado a processar pacotes demais, é referido como um roteador *backlogged*. Em outras palavras, o roteador de *backlog* é o roteador que tem mais carga ou maior número de pacotes na fila. Pode haver zero ou mais roteadores entre um emissor e um receptor. O remetente é a fonte dos pacotes e os receptores são a fonte de reconhecimentos. Como esses receptores receber pacotes 408, eles retornam reconhecimentos 412 para o remetente adaptativo 400. Neste exemplo, as confirmações são parte de comunicação regular TCP / IP.

Nestes exemplos ilustrativos, o remetente adaptativo 400 pode ser implementado como um processo em um sistema de processamento de dados, tal como servidor 104 na Figura 1. Os receptores 402, 404, e 406 são processos que podem ser executados em um dispositivo de recepção, tal como clientes 108, 110 e 112 na Figura 1. Em particular, estes processos

diferentes podem ser executados a partir de um sistema de processamento de dados, como sistema de processamento de dados 200 na Figura 2.

O remetente adaptativo 400 rastreia pacotes 408 enviados para cada receptor. O recebimento de reconhecimentos 412 também é controlado e é usado para identificar os parâmetros, como tempo de ida e volta. Tempo de ida e volta é o tempo desde que um pacote é enviado até que a confirmação é recebida. Neste exemplo, o tempo de viagem é baseado na perspectiva do remetente adaptativo 400.

Além disso, esta informação é utilizada para identificar blocos. Um bloco começa quando um pacote é enviado arbitrariamente; dados estatísticos são mantidos para todos os pacotes no bloco até que o pacote inicial que começou o bloco seja reconhecido pelo receptor. Quando o reconhecimento do pacote arbitrário é retornado, o remetente adaptativo 400 calcula as estatísticas para o bloqueio de pacotes. Em outras palavras, um ou mais pacotes podem estar presentes num bloco dependendo de quantos pacotes são enviados antes da confirmação para o pacote arbitrário no início do bloco ter sido devolvido. Além disso, o remetente adaptativo 400 também identifica uma

janela. A janela é o número de pacotes na rede que tinham sido enviados sem receber uma confirmação.

Além disso, o remetente adaptativo 400 também estima que o número de pacotes que estão localizados em roteador 5 410. Esta informação é determinada pelo tempo de viagem atual rodado para calcular o número esperado de pacotes não confirmados versus o número real de pacotes não confirmados. Por exemplo, se cinco pacotes devem existir em uma rede, devido a tempos de ciclo de viagem atual e oito 10 pacotes não confirmados são identificados pelo remetente adaptativo 400, o remetente adaptativo 400 pode-se estimar que três pacotes estão localizados no roteador 410.

O remetente adaptativo 400 controla a velocidade, aumentando ou diminuindo o tamanho da janela esperada 15 durante a tentativa de manter um número selecionado de pacotes no roteador 410. Uma grande janela é mais agressiva porque o roteador 410 passa mais tempo no processamento de pacotes de adaptação versus o tráfego da rede. Desta forma, o remetente adaptativo 400 pode sintonizar um parâmetro 20 alfa e beta para o roteador 410. Alfa é um inteiro que indica o número mínimo de pacotes no roteador 410 por ligação antes da janela ser aumentada por um pacote. Beta é um número inteiro que indica o número máximo de pacotes no roteador atrasado por ligação antes da janela ser diminuída

por um pacote. Estes parâmetros permitem pequenos aumentos ou diminuições na velocidade através do ajuste da janela com base em alfa e beta.

Outro parâmetro, o limiar, permite uma diminuição 5 rápida na velocidade quando 50% dos pacotes num bloco satisfazem as condições do presente limiar. Neste exemplo, o limiar é um percentual configurável do tempo de base de tempo de ida e volta ao máximo tempo de ida e volta.

Voltando à Figura 5, um diagrama de um Protocolo de 10 Internet / protocolo de controle de transmissão (TCP / IP) e protocolos similares são descritos de acordo com uma concretização ilustrativa da presente invenção. Protocolos TCP / IP e similares são utilizados por arquitetura de comunicações 500. Neste exemplo, a arquitetura de 15 comunicações 500 é um sistema de 4 camadas. Esta arquitetura inclui camada de aplicação 502, camada de transporte 504, camada de rede 506, e camada de enlace 508. Cada camada é responsável por lidar com várias tarefas de comunicação. A camada de enlace 508 também é referida como 20 a camada de enlace de dados ou a camada de interface de rede e normalmente inclui o controlador de dispositivo no sistema operacional e a placa de interface de rede correspondente no computador. Esta camada lida com todos os detalhes de hardware da interface física com os meios de

comunicação de rede sendo utilizados, tais como cabos ópticos ou cabos de Ethernet.

A camada de rede 506 também é referida como a camada de Internet e manipula o movimento de pacotes de dados em 5 torno da rede. Por exemplo, a camada de rede 506 manipula o roteamento de pacotes de vários dados que são transferidos através da rede. A camada de rede 506 na suíte TCP / IP é composta de vários protocolos, incluindo protocolo Internet (IP), controle de mensagem de protocolo Internet (ICMP) e o 10 protocolo *Internet Group Management* (IGMP).

Em seguida, a camada de transporte 504 fornece uma interface entre a camada de rede 506 e a camada de aplicação 502 que facilita a transferência de dados entre dois computadores host. A camada de transporte 504 está 15 relacionada com coisas tais como, por exemplo, dividir os dados passados para ela a partir da aplicação em pedaços de tamanho apropriado para a camada de rede abaixo, reconhecendo os pacotes recebidos, e ajustando o tempo limite para fazer o fim de certos outros pacotes de 20 reconhecimento que são enviados. No conjunto de protocolos TCP / IP, dois protocolos de transporte distintos estão presentes, TCP e protocolo *User Datagram* (UDP). TCP fornece serviços de confiabilidade para assegurar que os dados

sejam devidamente transmitidos entre dois hosts, incluindo a detecção de evasão e serviços de retransmissão.

Por outro lado, UDP fornece um serviço muito mais simples para a camada de aplicação simplesmente enviando 5 pacotes de dados chamados datagramas de um host para o outro, sem fornecer qualquer mecanismo para garantir que os dados são transferidos corretamente. Ao usar UDP, a camada de aplicação deve executar a funcionalidade, confiabilidade.

10 A camada de aplicação 502 lida com os detalhes da aplicação particular. Muitos aplicativos comuns de TCP / IP estão presentes para quase toda implementação, incluindo um Telnet para login remoto, um *File Transfer Protocol* (FTP), um protocolo *Simple Mail Transfer Protocol* (SMTP) para o 15 correio eletrônico e um *Simple Network Management Protocol* (SNMP).

Os aspectos da presente invenção são implementados na camada de aplicação 502 para adaptativamente controlar a transferência de dados de uma maneira que permite que as 20 definições do usuário sejam introduzidas por um usuário. Desta maneira, o usuário pode alterar as configurações diferentes para controlar como pacotes são enviados na rede para utilização no controle de largura de banda adaptativo. Ao permitir que as configurações do usuário sejam

introduzidas a partir do nível de aplicação, um usuário pode fazer alterações para diferentes tipos de redes e diferentes condições de rede que não são normalmente levadas em conta por um processo de controle adaptativo de largura de banda. Desta maneira, as outras condições do que aquelas em que os clientes podem ter levado em conta. Por exemplo, os aspectos da presente invenção permitem a um usuário alterar as configurações com base em uma identificação de fatores, tais como o número de saltos num caminho para o receptor e ligações no caminho para o receptor que tem uma grande quantidade de tráfego ou congestionamento.

Embora os exemplos ilustrativos da presente invenção são aplicados em um nível de aplicação, os diferentes processos também podem ser aplicados sobre as outras camadas. Por exemplo, os aspectos da presente invenção podem ser aplicados no interior da camada de transporte 504 ou camada de rede 506, dependendo da aplicação particular.

Voltando agora à Figura 6, um diagrama que ilustra processos de software e componentes utilizados no fornecimento de controle de banda adaptativo está representado de acordo com uma concretização ilustrativa da presente invenção. Neste exemplo, o remetente adaptativo 600 é uma ilustração mais pormenorizada dos processos

dentro remetente adaptativo 400 na Figura 4. O remetente adaptativo 600 contém três threads neste exemplo ilustrativo. A Thread de envio 602, thread de controle de congestionamento 604, e thread de sniffing de pacotes 606 5 são os componentes usados para adaptativamente enviar os dados para um ou mais receptores. A thread de envio 602 é usada para enviar dados através de fazer chamadas a um soquete. Um soquete é um objeto de software que conecta um aplicativo a um protocolo de rede, como um protocolo TCP / 10 IP em uma pilha TCP / IP.

A thread de envio 602 envia pedido 608 para a thread de controle de congestionamento 604 perguntando quantos dados podem ser enviados pela thread de envio 602. A thread de controle de congestionamento 604 retorna resposta 610, 15 dizendo quantos dados podem ser enviados. A thread de envio 602 usa a resposta 610 para enviar pacotes 612 para transmissão. Esses pacotes são armazenados em fila do roteador de backlog 614 até um roteador de backlog encaminha os pacotes para o seu destino. A fila do roteador 20 de Backlog 614 está localizada em um roteador atraso, tal como roteador 410 na Figura 4. Os reconhecimentos 616 são devolvidos ao remetente adaptativo 600, quando os pacotes são recebidos.

A thread de Congestionamento controle 604 identifica a quantidade de dados a serem enviados através da realização de um processo de controle adaptativo de banda. A thread de controle de congestionamento 604 nestes exemplos utiliza o processo de controle adaptativo de largura de banda para adaptativamente identificar as taxas de transmissão para a transmissão de pacotes de dados através de uma rede em resposta a alterações nas condições de rede. Estas condições de rede incluem, por exemplo, a quantidade de congestionamento na rede devido a várias outras transmissões de dados em adição àquelas que estão sendo processadas pelo remetente adaptativo. Em outras palavras, a taxa de velocidade com a qual os dados podem ser enviados altera de acordo com as condições da rede. A thread de controle de congestionamento 604 identifica diferentes taxas que aceleram ou retardam manter um mínimo impacto na rede com base nos diferentes parâmetros que estão definidos pelo usuário.

Um exemplo de condições de rede é ilustrado usando a Figura 7 abaixo.

Antes do servidor de FTP 752 enviar dados para o cliente de FTP 754, a condição de roteador 746 está inativa. Esta situação permite remetente adaptável 700 para enviar para os receptores ou pontos de extremidade a uma

taxa muito alta de velocidade, mesmo quando a prioridade é baixa. No entanto, assim que o servidor de FTP 752 começa a enviar dados para cliente de FTP 754, o roteador 746 fica sobrecarregado. Em seguida, o processo de controle adaptativo de largura de banda vai reagir de acordo com a sua prioridade. Em uma baixa prioridade, o processo de controle adaptativo de banda vai abrandar para onde ele dificilmente envia qualquer coisa, e terá um impacto mínimo sobre a distribuição de FTP. Na alta prioridade, este processo irá enviar muito rapidamente e causar a distribuição FTP para abrandar.

O seguinte é um exemplo de como distribuições adaptativas reagirão a condições de rede. Um banco contém um único roteador que gerencia a conexão de rede 10 para sistemas no interior do banco. Através deste roteador o banco está ligado a um site central (que gerencia 300 bancos em todo o país). Às 6 da manhã (antes da abertura do banco) não há ninguém no banco, e a rede (especificamente único roteador do banco) está ocioso. Neste momento, uma distribuição muito grande adaptativa é enviada em baixa prioridade a partir do local central. Como a distribuição é o único tráfego no roteador do banco, a distribuição acelera para usar 100 por cento da largura de banda do roteador. Isso continua até oito horas, quando os clientes

começam a utilizar ATM do banco e os sistemas de processamento de empréstimos, que também devem compartilhar a rede com o único roteador. A distribuição adaptativa irá imediatamente reconhecer que há um tráfego adicional pelo 5 vínculo lento (roteador do banco neste cenário). Como a distribuição de adaptação é de baixa prioridade, ele vai abrandar para usar uma porcentagem muito pequena do roteador causando um impacto mínimo sobre a ATM e tráfego de processamento de empréstimos. Às 5 da tarde, a 10 distribuição de adaptação ainda está em andamento. O tráfego de processamento de empréstimos diminui à medida que o banco fecha. A distribuição adaptativa irá sentir a carga reduzida da rede e continuar a acelerar como mais largura de banda torna-se disponível.

15 Reportando-nos novamente à Figura 6, este processo utiliza a informação localizada na fila 618. Em particular, thread de *sniffing* de pacotes 606 agarram pacotes e reconhecimentos a partir da rede e posiciona informação de pacote 620e informações de confirmação 622 na fila 618. A 20 thread de *sniffing* de pacote 606 filtram os pacotes e reconhecimentos para colocar dados apropriados necessários para a thread de controle de congestionamento 604 em fila 618. Alternativamente, todos os pacotes e reconhecimentos podem ser colocado em fila 618 para processamento pela

thread de controle de congestionamento 604. A informação na fila 618 é utilizada pelo thread de controle de congestionamento 604 para identificar os parâmetros, tais como um tempo de ida e volta a partir de quando um pacote 5 foi enviado até que a confirmação para o pacote foi recebida. Outros parâmetros que podem ser identificados e mantidos pela thread de controle de congestionamento 604 a partir de informações na fila 618 incluem um bloco, uma janela, e os pacotes no roteador de backlog.

10 Além disso, os aspectos da presente invenção incluem interface de usuário 624 que pode ser usada para definir os parâmetros definidos pelo usuário 626. A interface de usuário 624 é fornecida pela thread de parâmetro 628. Geralmente, os parâmetros são criados quando o processo 15 adaptativo é primeiro configurado em um sistema. Estes parâmetros podem ser alterados através de interface de usuário 624 e thread de parâmetro 628. Parâmetros definidos pelo usuário 626 são armazenados numa memória não-volátil, tal como um disco. Estes parâmetros podem ser lidos mais tarde, quando o processo adaptativo de banda inicia. Neste exemplo ilustrativo, estes parâmetros incluem um conjunto 20 de parâmetros em que os valores diferentes estão presentes para diferentes prioridades. Nos exemplos ilustrativos, o conjunto de parâmetros que se alteram com base no nível de

prioridade de um limiar de distribuição são, de dados, e tempo de espera máximo. Embora um usuário possa inserir os valores diferentes para os diferentes níveis de prioridade, esses parâmetros podem ser definidos diretamente para uso 5 pelo remetente adaptativo 600. A interface de usuário 624 permite ao usuário selecionar vários parâmetros de entrada ou para uso da thread de controle de congestionamento 604 na realização de processos de controle adaptativo de largura de banda.

10 Nestes exemplos ilustrativos, os aspectos da presente invenção permitem que um usuário para insira ou selecione valores de cinco parâmetros. Estes cinco parâmetros compreendem um limiar, uma mudança do tempo máximo de viagem, beta, suavização do tempo de ida e volta, e um 15 tempo máximo de espera. A variação máxima tempo de ida e volta é um parâmetro utilizado para restringir a quantidade de mudança no tempo de ida e volta atualmente medido em relação ao tempo de viagem medido da rodada anterior. O processo acompanha o tempo de viagem da atual rodada, o 20 tempo máximo de viagem e base de tempo de ida e volta.

 A thread de controle de congestionamento monitora os tempos de ida e volta em uma base por soquete e seqüência. Se um pacote tem um tempo de ida e volta de dez e um segundo pacote tem um tempo de ida e volta de 20, então o

tempo de ida e volta do segundo pacote é limitado por um valor de tempo viagem de variação máxima de um vírgula cinco (1,5). Mesmo que o tempo de ida e volta seja 20, o valor do tempo de ida e volta é registrado como dez vezes 5 um ponto cinco é igual a 15 ($10 \times 1,5 = 15$). Em seguida, o pacote subsequente pode ter um tempo de ida e volta, no máximo, de um ponto cinco vezes 15 ($1,5 \times 15$), o tempo de ida e volta registrado para o segundo pacote.

Redes normalmente têm uma certa quantidade de 10 aleatoriedade. Este parâmetro permite um processo de controle adaptativo de banda para ignorar um tempo de viagem demasiado grande ou demasiado pequeno, mas ainda permitir a grandes tempos de ida e volta, se ocorrer com freqüência suficiente. Este parâmetro permite ao usuário 15 configurar o quanto a mudança é tolerada.

O parâmetro de suavização de tempo de ida e volta nestes exemplos é um número inteiro utilizado para indicar o quanto as medições do tempo de ida e volta devem ser suavizadas por decaimento exponencial. O decaimento 20 exponencial é realizado pela média da medição anterior. Este parâmetro pode ajudar nos casos em que o processo de controle adaptativo de banda se comporta de forma irregular por compensação de flutuações no tempo de ida e volta. A

suavização tipicamente ocorre antes do parâmetro variação máxima levado em conta.

O tempo máximo de espera é um parâmetro que indica o múltiplo de tempo máximo de ida e volta de uma conexão irá esperar para enviar antes de desistir e reajustar. Às vezes reconhecimentos pode ser perdido em uma rede fazendo com que o remetente espere por um longo período de tempo antes de enviar os dados adicionais. Este é um valor para repor o processo de controle de banda adaptativo se muito tempo etapau antes que os dados serem enviados. No que diz respeito ao parâmetro de limiar, quando os tempos de ida e volta de pacotes são enviados em uma tomada, os tempos tendem a cair dentro de uma gama selecionada. Esta situação é especialmente verdadeira quando a rede está ociosa.

O limite é uma porcentagem do caminho entre o tempo de ida e volta para o mínimo tempo máximo de viagem. Por exemplo, um limiar de 20 por cento significa que o limiar é 20 por cento do caminho a partir do tempo de disparo base para o tempo máximo. Dependendo do tipo de rede, os tempos de ida e volta terão variação mais ou menos. As diferentes variações normalmente requerem diferentes valores. Por exemplo, uma rede de área local, bem comportada terá muito pouca variação no tempo de ida e volta, permitindo um valor baixo de limite. Uma rede de área ampla terá variações mais

elevadas do tempo de viagem. Com este tipo de rede, um limiar superior é necessário. Este parâmetro é configurável pelo usuário em particular para ter em conta tipos diferentes de redes que podem ser encontrados no envio de 5 dados. No que diz respeito ao parâmetro de limiar, quando os tempos de ida e volta de pacotes são enviados em uma tomada, os tempos tendem a cair dentro de uma gama selecionada. Esta situação é especialmente verdadeiro quando a rede está ocioso.

10 Se o tempo de ida e volta de um pacote é mais lento do que o valor de limiar, este pacote é considerado lento. Se 50 por cento dos pacotes num bloco são considerados lento, em seguida, a janela pode ser cortada ao meio, reduzindo grandemente a velocidade de distribuição. Como resultado, 15 permitindo a configuração do usuário deste valor permite ter em conta os tipos de rede e condições diferentes.

Os aspectos das mudanças de prioridades da presente invenção por modificação três parâmetros nos exemplos ilustrativos. Os parâmetros modificados para as diferentes 20 prioridades nestes exemplos são o limiar, beta, e o tempo de espera máximo.

Voltando à Figura 7, um diagrama que ilustra uma rede para o envio de uma distribuição de um remetente adaptável para receptores que utilizam prioridades é descrito de

acordo com uma concretização ilustrativa da presente invenção. Neste exemplo, remetente adaptativo 700 envia uma distribuição através da rede 702 a clientes 704, 706, 708, 710, 712, 714, 716, 718, 720, 722, 724, 726, 728, 730, 732, 5 734, 736, 738, 740, e 742. 750, 748, 746, e 744 são roteadores 746 é o roteador de backlog. Estes clientes estão ligados ao roteador 744, que é por sua vez ligado ao roteador 746. O roteador 746 conecta ao roteador 748. O remetente adaptativo 700 conecta à rede através do servidor 10 702, 750. Neste exemplo, a rede 702 contém vinte hops. Além de uma distribuição, *File Transfer Protocol* (FTP) 752 também pode enviar dados para cliente de FTP 754. Neste exemplo, servidor de FTP 752 envia dados para cliente de FTP 754, enquanto a distribuição do remetente adaptativo 15 700 é enviada para os clientes. Os resultados ilustrativos do exemplo no servidor de FTP gerando uma alta demanda de tráfego causando uma inundação no link em roteador 746. Ao definir prioridades, o remetente adaptativo 700 pode usar diferentes quantidades de largura de banda ao enviar a 20 distribuição para esses clientes.

Voltando agora à Figura 8, um diagrama que ilustra o uso da banda numa rede é descrito de acordo com uma concretização ilustrativa da presente invenção. O gráfico da Figura 8 ilustra a percentagem de largura de banda

utilizada na transferência de dados para uma rede, tal como o ilustrado na Figura 7. Neste exemplo, a linha 802 mostra a quantidade de largura de banda utilizada para diferentes tipos de distribuição. Na seção 804, apenas uma 5 transferência de FTP do servidor de FTP 752 para cliente de FTP 754 da Figura 7 é mostrado. Cem por cento da largura de banda é utilizada na presente distribuição. Com uma configuração de baixa prioridade para a distribuição a partir do remetente adaptativo 700 na Figura 7 para os 10 clientes, roteador 744 na Figura 7 aloca cerca de 80 por cento da largura de banda para a transferência de FTP, como ilustrado na secção 806. Com uma parte média, cerca de 50 por cento da largura de banda é usada como mostrado na seção 808 para a transferência de FTP. Como pode ser visto, 15 à medida que aumenta a prioridade, menor largura de banda é alocada para a transferência de FTP com maior largura de banda a ser alocada para a distribuição pelo remetente adaptativo. Na seção 810, uma alta prioridade é dada para a distribuição para os clientes. Como pode ser visto, a 20 transferência de FTP cai para um nível variando em torno de 30 por cento da largura de banda. Na seção 812, o processo de controle adaptativo de banda não é usado. Como resultado, nenhum do controle de congestionamento descrito é utilizado na presente secção particular.

Voltando agora à Figura 9, um fluxograma de um processo para a configuração de definições de usuário para os parâmetros é descrito de acordo com uma concretização ilustrativa da presente invenção. O processo ilustrado na 5 Figura 9 pode ser implementado numa thread de parâmetro 628 na Figura 6. Este processo é utilizado para permitir a um usuário para definir ou alterar os parâmetros utilizados no controle de largura de banda adaptativo. Nestes exemplos, os parâmetros são limite, ida e volta mudança do tempo 10 máximo, suavização do tempo de ida e volta, máximo tempo de espera, e beta. Em particular, o processo ilustrado na Figura 9 pode ser utilizado para definir os parâmetros para os gateways diferentes para uso em diferentes níveis de prioridade, tais como alta, média e baixa. Nestes exemplos, 15 uma alta prioridade pode fixar beta igual a sete, limiar igual a 99 e o tempo de espera máximo igual a 20. A prioridade média pode definir beta igual a cinco, valor igual a 40, e um tempo de espera máximo igual a 30. Com uma baixa prioridade, beta é definido igual a três, limiar 20 igual a 25, e um tempo de espera máximo igual a 40. Estas configurações específicas são aquelas utilizadas para os diferentes níveis de prioridade. Essas configurações podem ser definidas através da interface do usuário, tal como previsto.

Além disso, os aspectos da presente invenção também levam em conta os efeitos de qualquer roteador ou servidor da gateway para o ponto final nestes exemplos ilustrativos. Esta capacidade é fornecida através do uso de tempos de ida 5 e volta nos processos de controle adaptativo de largura de banda.

O processo começa por configurações de usuário que apresentam (etapa 900). Essas configurações podem ser apresentadas em uma interface de usuário, tais como 10 interface de usuário 624 na Figura 6. O processo, então, recebe entrada do usuário (etapa 902). A determinação é feita para saber se a entrada do usuário altera as configurações dos parâmetros (etapa 904). Se a entrada do usuário altera as configurações, as configurações antigas 15 são substituídas com as novas configurações (etapa 906). Depois disso, o processo retorna para a etapa 900 para apresentar estas definições para o usuário.

Com referência novamente para a etapa 904, se a entrada do usuário não alterar as configurações, é feita 20 uma determinação quanto a se a entrada do usuário é para finalizar o processo de alteração das configurações do usuário (etapa 908). Se a entrada de usuário não está para terminar o processo, o processo retorna para a etapa 900. Caso contrário, as configurações do usuário são salvas

(etapa 910) com o processo terminando depois. Essas configurações são salvas como parâmetros definidos pelo usuário 626 na Figura 6 nestes exemplos.

Voltando, a seguir, para a Figura 10, um fluxograma de 5 um processo para um thread de envio é descrito de acordo com uma concretização ilustrativa da presente invenção. O processo ilustrado na Figura 10 pode ser implementado em uma thread de envio, tal como a thread de envio 602 na Figura 6.

10 O processo começa por enviar um pedido para a quantidade de dados podem ser enviados para a thread de controle de congestionamento (etapa 1000). O processo, então, recebe uma resposta (etapa 1002). Esta resposta contém a quantidade de dados que podem ser enviados como um 15 resultado do processo de controle adaptativo de largura de banda executado pela thread de controle de congestionamento. Em resposta à recepção da resposta, o processo envia uma chamada para o soquete para enviar apenas a quantidade aceitável de dados (etapa 1004).
20 Posteriormente, é feita uma determinação quanto a se os dados estão presente para ser enviado (etapa 1006). Se mais dados estão presente, o processo retorna para a etapa de 1000. Caso contrário, o processo termina.

Voltando, a seguir, para a Figura 11, um fluxograma de um processo para uma thread de sniffing de pacotes é descrita de acordo com uma concretização ilustrativa da presente invenção. O processo ilustrado na Figura 11 pode 5 ser implementado em threads de sniffing pacotes 606 na Figura 6.

O processo começa por determinar se um pacote tenha sido detectado (etapa 1100). Se um pacote tiver sido detectado, o processo extrai o identificador do pacote e o 10 carimbo a partir do pacote (etapa 1102). O processo, em seguida, armazena os dados em uma fila (etapa 1104). Esta fila é acessível por uma thread de controle de congestionamento para que os dados possam ser utilizados na determinação de ida e volta e executar processos de 15 controle adaptativo de largura de banda.

Com referência de novo para a etapa 1100, se um pacote não é detectado, é feita uma determinação sobre se um reconhecimento foi detectado (etapa 1106). Se uma confirmação não tiver sido detectada, o processo retorna 20 para a etapa de 1100. Se um reconhecimento foi detectado na etapa 1106, o identificador do pacote e a hora para o reconhecimento são extraídos (etapa 1108). O processo prossegue então para a etapa 1104, como descrito acima.

Voltando agora à Figura 12, um fluxograma de um processo para um segmento de controle congestão é descrito de acordo com uma concretização ilustrativa da presente invenção. Neste exemplo, o processo ilustrado na Figura 12 pode ser 5 implementado numa thread de controle de congestionamento 604 na Figura 6.

O processo começa pela recepção de um pedido de um thread de envio (etapa 1200). Nestes exemplos, o processo de controle adaptativo utilizado pela thread de controle de 10 congestionamento está localizado num gateway. Cada gateway tem seus próprios valores para os diferentes parâmetros utilizados para as prioridades. Quando um thread de envio começa a enviar uma distribuição, o thread de envio passa esses valores, juntamente com a prioridade atual 15 distribuição para o thread de controle de congestionamento.

O thread de controle de congestionamento irá usar os valores que correspondem à prioridade de distribuição atual. Esta informação pode ser recebida na etapa 1200 a primeira vez que o pedido é feito para o envio de uma 20 distribuição. Este pedido pede uma identificação da quantidade de dados que podem ser enviados. Seguidamente, os dados são puxados a partir da fila (etapa 1202). Estes dados incluem os horários de chegada e identificadores de pacote. Posteriormente, os tempos de ida

e volta são identificados a partir dos dados na fila (etapa 1204). Os parâmetros são então obtidos com base em uma prioridade para a distribuição (etapa 1206). Estes parâmetros provenientes dos parâmetros definido pelo 5 usuário 626 na Figura 6. Em particular, os parâmetros definidos pelo usuário são para um determinado conjunto de parâmetros associados com a prioridade selecionada. Nestes exemplos ilustrativos, os parâmetros que têm configurações diferentes para diferentes prioridades são limiar, beta e 10 tempo de espera máximo. Depois disso, o processo é executado um processo de controle adaptativo de largura de banda (etapa 1208). Este processo pode ser, por exemplo, as etapas contido dentro do thread de controle de congestionamento. Alternativamente, o segmento pode chamar 15 uma função ou processo externo na etapa 1208. O processo então obtém um resultado (etapa 1210), e uma resposta é devolvida com a quantidade de dados que podem ser enviados (etapa 1212) com o processo de terminação em seguida.

Voltando a seguir para a Figura 13, um fluxograma de 20 um processo para a realização de controle de banda adaptativo está representado de acordo com uma concretização ilustrativa da presente invenção. O processo representado na Figura 13 é uma descrição mais detalhada da etapa 1208 na Figura 12.

O processo começa pela identificação o menor tempo de ida e volta e os maiores tempos de ida e volta (etapa 1300). Depois disso, a janela é identificado (etapa 1302). O processo então identifica o número de pacotes acima e 5 abaixo do limiar por reconhecimento (etapa 1304). Em seguida, o processo determina se 50 por cento dos pacotes num bloco estão acima do limiar (etapa 1306). Se 50 por cento dos pacotes em um bloco não estão acima do limiar, a taxa de transferência esperado é calculada etapa (etapa 10 1308). Na etapa 1308, o rendimento esperado é calculado como se segue:

$$E \leftarrow \underline{W}$$

$$\min RTT$$

E é o rendimento esperado, W é a janela, e minRTT é o menor tempo de ida e volta visto a partir dos tempos de ida e volta obtidos a partir da fila. Depois disso, 15 o processo identifica a taxa de transferência real (etapa 1310).

Esta transferência real é identificada usando o 20 seguinte:

$$A \leftarrow \underline{W}$$

$$\text{observedRTT}$$

A é a taxa de transferência real, W é a janela, e observedRTT é o valor que é medido utilizando a

diferença de o tempo de quando o pacote é enviado e quando o reconhecimento é recebido. Como resultado, o rendimento esperado baseia-se o tempo mínimo de ida e volta, porque uma expectativa de que está presente em
5 uma rede ociosa o tempo de viagem mínimo rodada é sempre o resultado de envio de um pacote. O rendimento real baseia-se nas condições atuais da rede em que o tempo de viagem atual rodada é maior do que o tempo de viagem mínimo rodada. Nestes exemplos, o tempo de
10 viagem mínimo é o mesmo que o tempo de disparo base. O processo, então, calcula o número de pacotes no roteador backlog (etapa 1312). O número real de pacotes é estimado usando o seguinte:

$$\text{Dif} \leftarrow (\text{E} - \text{A}) \cdot \text{minRTT}$$

15 Diff é o número de pacotes no roteador atraso, E é a taxa de transferência esperado, A é a taxa de transferência real, e minRTT é a menor rodada tempo de viagem visto.

Em seguida, é feita uma determinação sobre se o número
20 de pacotes no roteador atraso é inferior a alfa (etapa 1314). Como descrito acima, o alfa é um número inteiro que indica o número de pacotes que numeral devem estar presentes em um roteador atraso por ligação antes da janela é aumentada por um pacote. Se o número de pacotes no

roteador atraso é inferior a alfa, os incrementos de processo a janela por um (etapa 1322).

Depois disso, o processo indica que não há problema para enviar dados até o tamanho da janela (etapa 1318) com 5 o processo encerra daí em diante.

Com referência de novo para a etapa 1314, se o número de pacotes no roteador atraso não é inferior a alfa, é feita uma determinação sobre se o número de pacotes é maior do que o beta (etapa 1316). Se o número de pacotes é maior 10 do que o beta, o tamanho da janela é diminuído por um (etapa 1324). O processo prossegue então para a etapa 1318, como descrito acima.

Caso contrário, o processo prossegue para a etapa 1318, sem alterar o tamanho da janela.

15 Com referência de volta para a etapa 1306, se 50 por cento dos pacotes num bloco estão acima do limiar, a janela é reduzida para metade (etapa 1320). O processo prossegue então para a etapa 1318, como descrito acima.

O processo ilustrado na Figura 13 acima é baseado em 20 modificações de algoritmos de controle de largura de banda disponíveis atualmente, como o algoritmo de Nice descrito no Venkataramani et al. "TCP Nice: um mecanismo para transferências de plano fundo", ACM SIGOPS Sistemas Operacionais Review, vol. 36, Issue SI Inverno 2002, pp 1-

15., Que é aqui incorporada por referência.

Os aspectos da presente invenção têm identificado um número de parâmetros que afetam a implementação do processo 5 de controle adaptativo de largura de banda descrito na Figura 13. Estes parâmetros e as descrições são listadas a seguir:

NICE_ALPHA - Um inteiro que indica o número mínimo de pacotes no roteador de backlog por conexão antes que a 10 janela é aumentada. O padrão é 1.

NICE_BASE_DISCARD - Um inteiro que indica o número de medições RTT iniciais de base para descartar por conexão. O raciocínio é que quando a distribuição é iniciado e a rede ainda não está saturada, medições de RTT excessivamente 15 baixas de base podem ser feitas inicialmente. O padrão é 3.

NICE_BASE_SCALE - Um número de ponto flutuante que especifica um fator de escala que é aplicado ao RTT mínimo global (v_baseRTT) mantido pela NICE. Definindo-a para um pequeno número positivo, como 1.1, pode ajudar se a rede 20 permite, ocasionalmente RTTs atípicamente curtos. O padrão é 0,0.

NICE_BELOW_ONE - Um número inteiro que fornece um limite inferior para v_cwnd_below_one, que é o número

máximo de RTTs que uma ligação pode ser inativo fazer para uma janela de baixa. O padrão é 48.

BETA NICE - Um inteiro que indica o número máximo de pacotes no roteador de backlog por conexão antes que a 5 janela ser diminuída. Este é um valor padrão de beta. O padrão é baseado em prioridade.

NICE_CLAMP - Quando definido de aperto o tamanho da janela (snd_cwnd) está limitado a não ser mais do que quatro pacotes maior do que o número de pacotes que estão 10 atualmente em rede. O padrão é definido.

COND NICE - Quando definir a thread aguarda o envio de uma condição que é sinalizado pela thread ack em vez de esperar uma quantidade arbitrária de tempo. Padrão não está definido.

15 NICE_CONG_RTX_Q - Quando definido o fio de extensão cong irá adicionar ou atualizar uma estimativa de cada pacote de saída para o rtx_q; que é usado para calcular RTTs. O segmento de cong tem a vantagem de que a sua estimativa do tempo de saída é precisa, mas pode descartar 20 pacotes. O padrão é definido.

NICE_DYNAMIC_MSS - Quando definido, o processo de controle adaptativo banda vai começar com um baixo valor para MSS e aumentá-la cada vez que um pacote de saída que tem um MSS maior. Desta forma, o MSS deve rapidamente

aproximar-se do MSS utilizado para a conexão. O padrão é definido.

NICE_FAST - Um inteiro que especifica o número de milissegundos cada adaptativa banda Tomada processo de controle é estar em fase de arranque rápido. Quanto mais tempo a fase de arranque rápido mais precisa a estimativa de maxRTT. No entanto, definindo-o para um valor grande causa processo de controle adaptativo de banda para ser efetivamente desativado para essa quantidade de tempo.

10 O padrão é 5000 = 5 seg.

NICE_FAST_RETURN - Quando definida a fase de início rápido é encerrado após o primeiro enviar () com falha com um errno de EWOULDBLOCK. Isso deve minimizar a quantidade de tempo gasto no modo de início rápido, que não produz para o gateway, uma vez enchendo a fila de saída do soquete deve ser muito rápido. O padrão é definido.

NICE_INTERFACE - A interface (rede identificador do cartão) que é para ser usado por agradável. Neste momento, o processo de controle adaptativo de banda não é capaz de determinar dinamicamente a interface correta. Assim, a interface precisa ser configurada manualmente, se não é a primeira interface ativa. Normalmente, isso seria definido como coisas como "eth1". O padrão é removida.

NICE_MAX_MULT - Um número de ponto flutuante que especifica um valor mínimo para maxRTT como um múltiplo de baseRTT. Quando definido, deve ajudar a prevenir valores excessivamente baixos para maxRTT, bem como a baixa taxa de transferência que tende a ir junto com isso. O padrão é 5 0,0.

NICE_MIN_MSS - Um inteiro que indica o MSS mínimo a ser utilizado. No interesse da eficiência valores mais baixos fornecidos pelo sistema operacional são ignorados em 10 favor de NICE_DEFAULT_MSS. O padrão é 1000.

NICE_MIN_PACKET - Um inteiro que especifica a quantidade mínima de dados que devem ser enviados em uma única vez. Isso é diferente de MIN_NICE_SEND em que se ok para enviar é menor que o valor especificado é definido 15 como 0. Isso é para evitar uma situação em que o thread de controle de congestionamento continua dizendo que a thread de envio envia valores muito pequenos como 1 byte. Ao definir este valor para 10, o thread de controle de congestionamento iria esperar até pelo menos 10 bytes 20 possam ser enviados (retornando 0 até cálculos específicarem 10). O padrão é 0.

MIN_NICE_SEND - Um inteiro que especifica o valor mínimo de ok para enviar, o que significa que o thread de controle de congestionamento sempre dirá a thread de envio

para enviar pelo menos esta quantidade de dados. Ao definir esta a 10, em seguida, mesmo se os cálculos adaptativas especificarem 3, a thread de controle de congestionamento retornará 10. O padrão é 0.

5 NICE_NANO_FIXED - Um inteiro que especifica o número de micro segundos que a função de retorno interna, select_delay (), é para dormir, quando utilizado.

NICE_NANO_FIXED não tem efeito quando o retorno é definido como algo diferente do que select_delay (). Quando 10 não definido, o atraso é calculado dinamicamente com base em quanto tempo deve demorar para que haja espaço suficiente para enviar um pacote, como indicado pela taxa de transferência do último bloco (determinado pelo RTT do pacote o mais rápido no último bloco e o tamanho do bloco).
15 Quando definido para 1234 nanosleep () não é chamado: não definido - Atraso é calculado dinamicamente 0 - nanosleep () é chamado com um valor de 0, que pode ser um atraso, desde que 10 ms em alguns sistemas 1234 - Não chame nanosleep () outro - Chamada de nanosleep () com o número de 20 microsssegundos especificados padrão é 0.

NICE_PCAP_TIMEOUT - Um inteiro que indica quanto tempo o sistema operacional deve esperar antes de retornar com uma lista de pacotes capturados. Esta variável não tem efeito sobre os sistemas Linux uma vez que os sistemas

Linux só esperaram até que um pacote esteja disponível independentemente de quanto tempo leva. Em sistemas onde esta variável tem um efeito, como o Solaris, há um trade-off entre a obtenção RTTs precisos quando o tempo limite é 5 baixo e desperdiçar tempo de CPU quando a rede está ociosa. O padrão é 10 ms (o menor valor suportado no Solaris).

NICE_QUEUE_LIMIT - O número inteiro que especifica o comprimento máximo da fila de pacotes escritos pela thread que são lidos a partir de pelo segmento Cong. Quanto mais 10 tempo na fila de menos sensível é bom como a thread cong está agindo sobre a informação que está atrasada por mais tempo que leva para pacotes de trabalhar seu caminho através da fila. O padrão é 10.

NICE_RTT_MAX_CHANGE - Um inteiro que indica a 15 quantidade máxima que os RTTs base e Max estão autorizados a mudar em relação ao valor anterior. Quando definido, o processo de controle adaptativo de banda deve ser mais tolerante com falsos valores RTT extremas. O padrão é 1,5.

NICE_RTT_MIN_STDS - Um inteiro que indica o número 20 mínimo de desvios-padrão acima dos RTTs base RTT terá que ser considerado acima do RTT base. Padrão em 0.

NICE_RTT_SMOOTHING - Um inteiro que indica o quanto o RTT (Round Trip Time) as medições devem ser suavizadas por decaimento exponencial, que é feito pela média com a

medição anterior. Definir o que pode ajudar nos casos em que se comporta de forma irregular agradável, compensando ao longo de cada flutuação RTT. O padrão é 50.

NICE_RTT_STD_SMOOTH - Um número inteiro que é 5 semelhante ao NICE_RTT_SMOOTHING, mas para os desvios padrão. O desvio padrão é baseado em uma média ponderada das medições RTT recentes. O padrão é 0.

NICE_SEND_RTX_Q - Quando definir o thread enviando irá adicionar ou atualizar uma estimativa de cada pacote de 10 saída para o rtx_q que é usado para calcular RTTs. O thread de envio tem a vantagem de que não cair pacotes, mas a sua estimativa do tempo de saída pode ser impreciso. O padrão é definido.

NICE_THROUGHPUT_AVG - Um inteiro que indica como 15 muitos dos pacotes mais recentes são para serem incluídos no cálculo de transferência. Valores maiores devem resultar em mais cálculos de rendimento precisos, mas à custa de ser menos sensível. O padrão é 20.

NICE_THROUGHPUT_START - Um inteiro que indica o número 20 mínimo de pacotes que devem ser enviadas antes de os cálculos de transferência são começado. Esta é a conta para a fila de link lento inicialmente estar vazia, então o rendimento durante a parte inicial do soquete não pode ser típico da tomada e deve ser ignorado. O padrão é 100.

NICE_WAIT_BASE - Um inteiro que indica o múltiplo de v_baseRTT (RTT mínimo global) uma conexão irá esperar para enviar antes de desistir e redefinir snd_nxt e snd_una. O tempo limite calculado é adicionado a que, para 5 NICE_WAIT_MAX. O padrão é 0.

NICE_WAIT_MAX - Um inteiro que indica o múltiplo de v_maxRTT (global máximo RTT) uma conexão irá esperar para enviar antes de desistir e reajuste. O padrão é baseado em prioridade.

10 Voltando agora à Figura 14, um fluxograma de um processo para personalizar o envio de distribuições de receptores para um cliente está representado de acordo com uma concretização ilustrativa da presente invenção. O processo ilustrado na Figura 14 é um processo usado para 15 fornecer soluções para os clientes que desejam sistemas de distribuição personalizados dentro das suas redes.

O processo começa pela recepção de um pedido do cliente (etapa 1400). Este pedido inclui informação necessária para gerar a solução. Por exemplo, uma 20 identificação de gateways na rede do cliente está contida no pedido. Além disso, as características dos diferentes clientes acedidos através dos portais também podem ser incluídos. Além disso, os requisitos do cliente para o envio de diferentes tipos de distribuição também estão

incluídos no exemplo ilustrativo. Em resposta à recepção deste pedido, os parâmetros de cliente são analisados para o envio de distribuições para os pontos de extremidade (etapa 1402).

5 Nesta análise, as definições são identificadas para os parâmetros utilizados para as diferentes prioridades (etapa 1404). Nestes exemplos ilustrativos, as prioridades são limiar, beta e tempo de espera máximo. Os parâmetros podem ser identificados para vários níveis de prioridade. Por 10 exemplo, os exemplos ilustrativos usar três níveis de prioridade, alta, média e baixa. Dependendo do pedido do cliente, um número diferente de níveis de prioridade pode ser utilizado, tais como dois ou quatro. Os parâmetros gerados para um nível de prioridade particular pode diferir 15 para gateways diferentes, dependendo a solicitação do cliente e as características da rede na qual os portais estão localizados.

Depois disso, o código é gerado para controle de banda adaptável para cada gateway (etapa 1406). Este código é 20 embalado de tal forma que o código pode ser instalado de um gateway particular. Este código inclui o processo de controle adaptativo de banda e os parâmetros para o gateway particular. Este código também pode incluir os arquivos executáveis necessários para instalar e configurar o

processo de controle adaptativo de banda em um gateway. Em seguida, o código é enviado ao cliente (etapa 1408) para a distribuição. Desta maneira, um cliente pode receber os sistemas de distribuição personalizados a partir de uma 5 empresa, ou outro fornecedor em resposta a um pedido.

Assim, os aspectos da presente invenção proporcionam um método implementado por computador, aparelho, e código de programa de computador utilizável para ajustar a taxa de transferência de dados. Os aspectos da presente invenção 10 recebem a entrada do usuário para selecionar valores para os parâmetros em um nível de aplicativo. Estes parâmetros definidos pelo usuário são utilizados em um processo adaptativo de largura de banda de controle para identificar o número de pacotes que podem ser enviadas para um destino 15 através de uma rede. Os aspectos da presente invenção permitem que um usuário para defina os parâmetros com base nas condições de rede e características diferentes. Desta maneira, os aspectos da presente invenção permitem o envio adaptativo de dados com base em diversos tipos de redes e 20 condições.

A invenção pode assumir a forma de uma concretização totalmente em hardware, uma concretização inteiramente em software ou uma concretização contendo tanto o hardware e elementos de software. Numa concretização preferida, a

invenção é implementada em software, o que inclui, mas não está limitado a firmware, residente microcódigo de software, etc.

Além disso, a invenção pode assumir a forma de um 5 produto de programa de computador acessível a partir de um meio utilizável por computador ou legível pelo computador fornecendo código de programa para utilização ou em conexão com um computador ou qualquer sistema de execução de instruções. Para os fins desta descrição, um meio 10 utilizável computador ou legível por computador pode ser qualquer aparelho tangível que pode conter, armazenar, comunicar, propagar ou transportar o programa para utilização ou em conexão com o sistema de execução da instrução, o aparelho ou o dispositivo.

15 O meio pode ser um sistema eletrónico, magnético, óptico, eletromagnéticos, infravermelho ou de semicondutores (ou um aparelho ou dispositivo) ou um meio de propagação. Exemplos de um meio legível por computador incluem um semicondutor ou a memória de estado 20 sólido, fita magnética, um disquete de computador removível, uma memória de acesso aleatório (RAM), uma memória só de leitura (ROM), um disco rígido magnético e um disco óptico. Exemplos atuais de discos ópticos incluem

disco compacto - memória apenas para leitura (CD-ROM), CD de leitura e gravação(CD-R / W) e DVD.

Um sistema de processamento de dados adequada para armazenar e / ou executar de código de programa irá incluir, pelo menos, um processador acoplado direta ou indiretamente para elementos de memória através de um barramento de sistema. Os elementos de memória podem incluir memória local empregue durante a execução efetiva do código de programa, carga de armazenamento e memórias cache que fornecem armazenamento temporário de pelo menos alguns código de programa, a fim de reduzir o número de código vezes devem ser recuperados a partir de grandes quantidades de armazenamento durante a execução.

Dispositivos de entrada / saída (incluindo, mas não se limitam a teclados, monitores, dispositivos apontadores, etc) podem ser acoplado ao sistema, quer diretamente ou através de intervenção de controladores de E / S.

Os adaptadores de rede podem também ser acoplados ao sistema para permitir que o sistema de processamento de dados tornar-se acoplado a outros sistemas de processamento de dados ou impressoras remotas ou dispositivos de armazenamento através da intervenção de redes privadas ou pública. Modems, cable modems e placas de Ethernet são

apenas alguns dos tipos disponíveis atualmente de adaptadores de rede.

A descrição da presente invenção foi apresentada para fins de ilustração e descrição e não se destina a ser exaustiva ou limitada com a invenção sob a forma divulgada. Muitas modificações e variações serão evidentes para aqueles com conhecimento corrente na técnica. A concretização foi escolhida e descrita a fim de melhor explicar os princípios da invenção, a aplicação prática, e para permitir que outros especialistas no assunto compreendam a invenção, para várias concretizações com várias modificações que são adequadas para o uso particular contemplado.

REIVINDICAÇÕES

1. Método implementado por computador para controlar de forma adaptativa a transmissão de dados, o método implementado por computador **caracterizado pelo** fato de que compreende:

configurar um conjunto de parâmetros para um determinado gateway na pluralidade de gateways, em que o conjunto de parâmetros para o gateway particular é diferente do conjunto de parâmetros para outro gateway na pluralidade de gateways;

receber dados de uma fonte (302) em uma pluralidade de gateways (304, 306, 308) para distribuição usando uma prioridade selecionada; e

transmitir os dados da pluralidade de gateways para uma pluralidade de receptores (310, 320, 330) utilizando a prioridade selecionada, em que cada gateway na pluralidade de gateways tem um processo de controle de largura de banda adaptativo e um respectivo conjunto de parâmetros para controlar o processo de controle de largura de banda adaptativo para enviar os dados na prioridade selecionada, em que o processo de controle de largura de banda adaptativo leva em consideração o congestionamento ao longo de um percurso de um gateway na pluralidade de gateways para um receptor da distribuição e em que a transmissão dos dados de cada gateway para a prioridade selecionada utiliza uma largura de banda diferente da utilizada pelos outros gateways diferentes na pluralidade de gateways para a prioridade selecionada quando são definidos valores

diferentes para cada um dos respectivos conjuntos de parâmetros para os diferentes gateways.

2. Método implementado por computador, de acordo com a reivindicação 1, **caracterizado pelo** fato de que os dados compreendem um de um arquivo de dados, uma atualização de aplicativo e um patch de vírus.

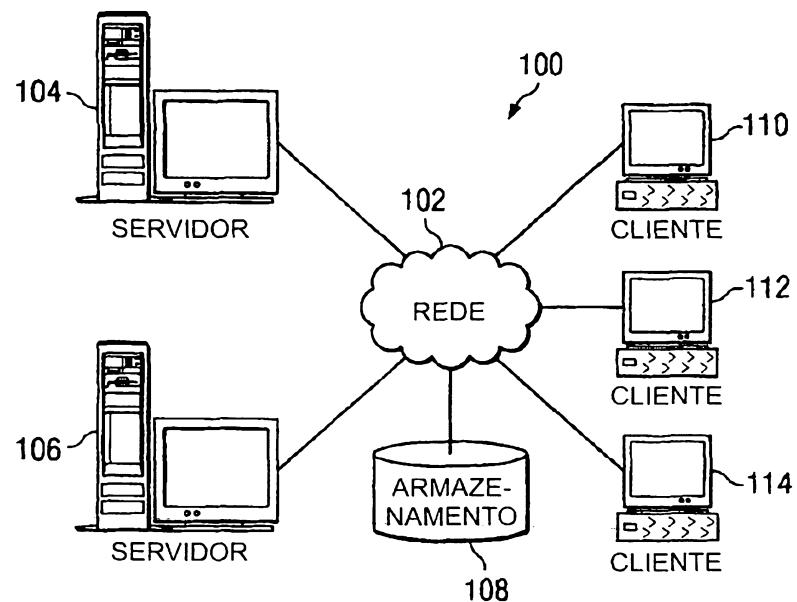


FIGURA 1

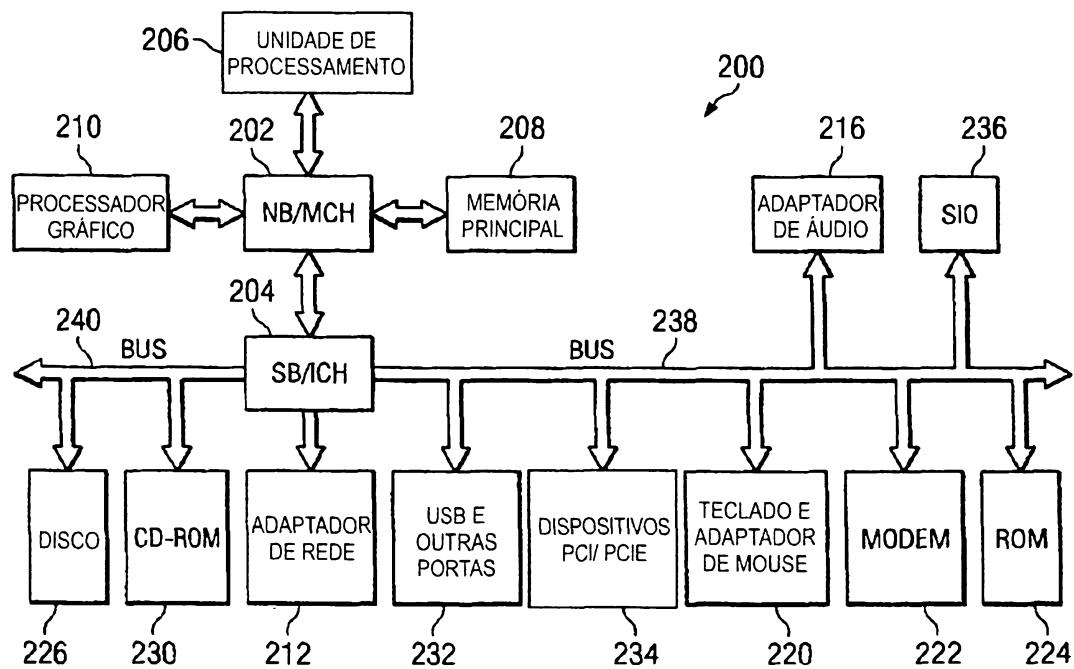


FIGURA 2

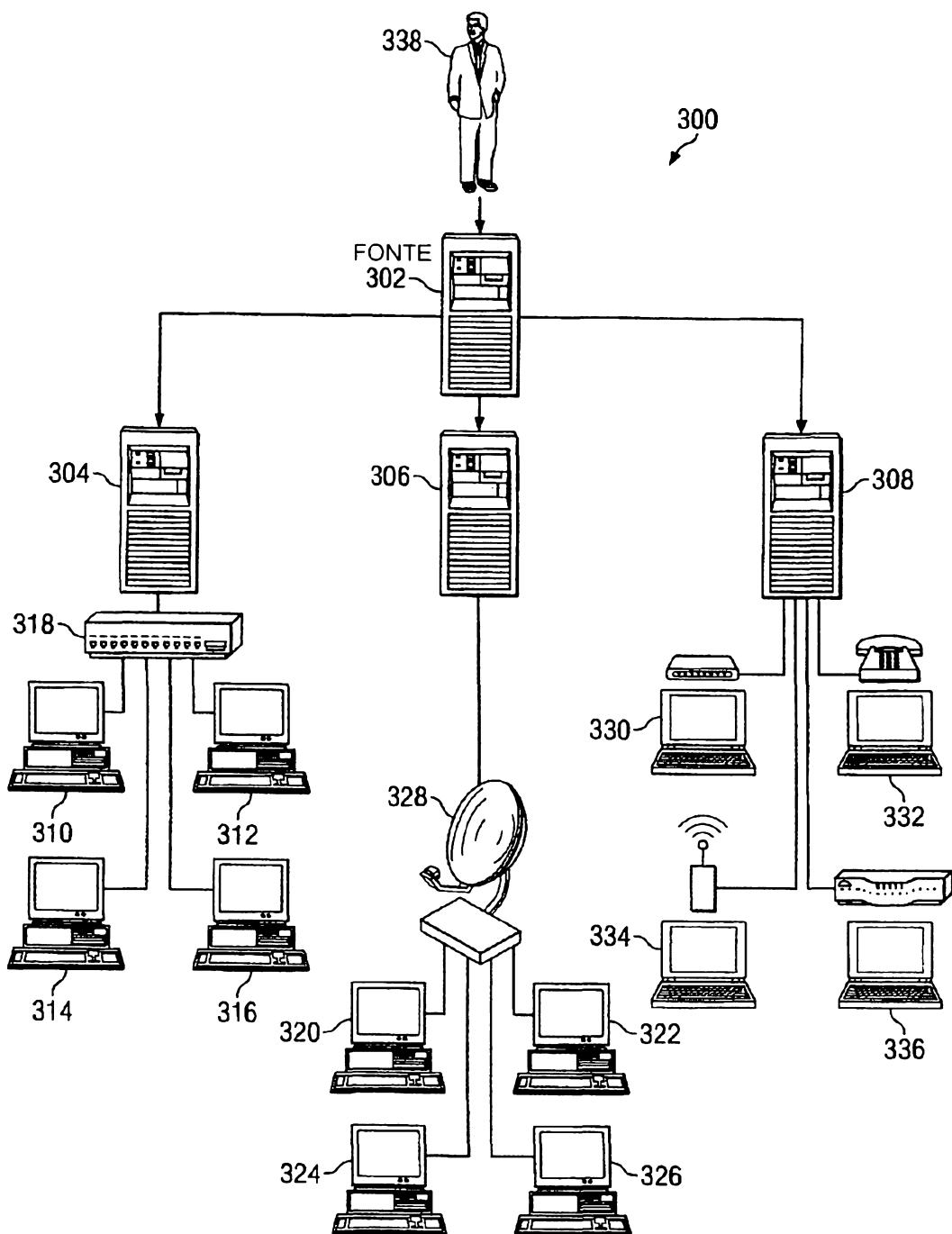


FIGURA 3

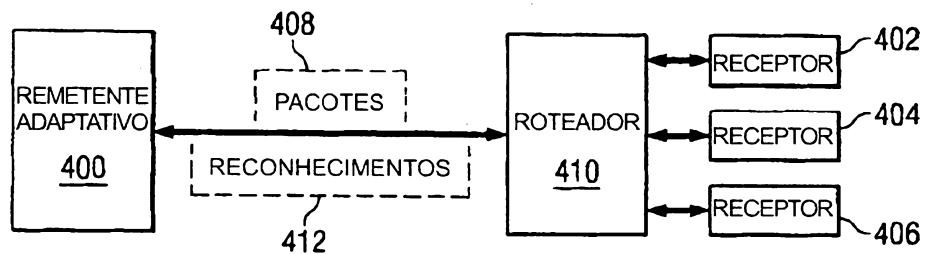


FIGURA 4

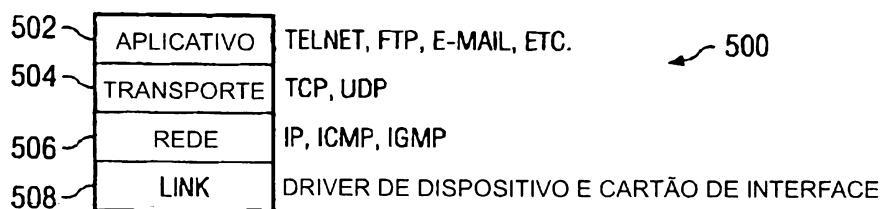


FIGURA 5

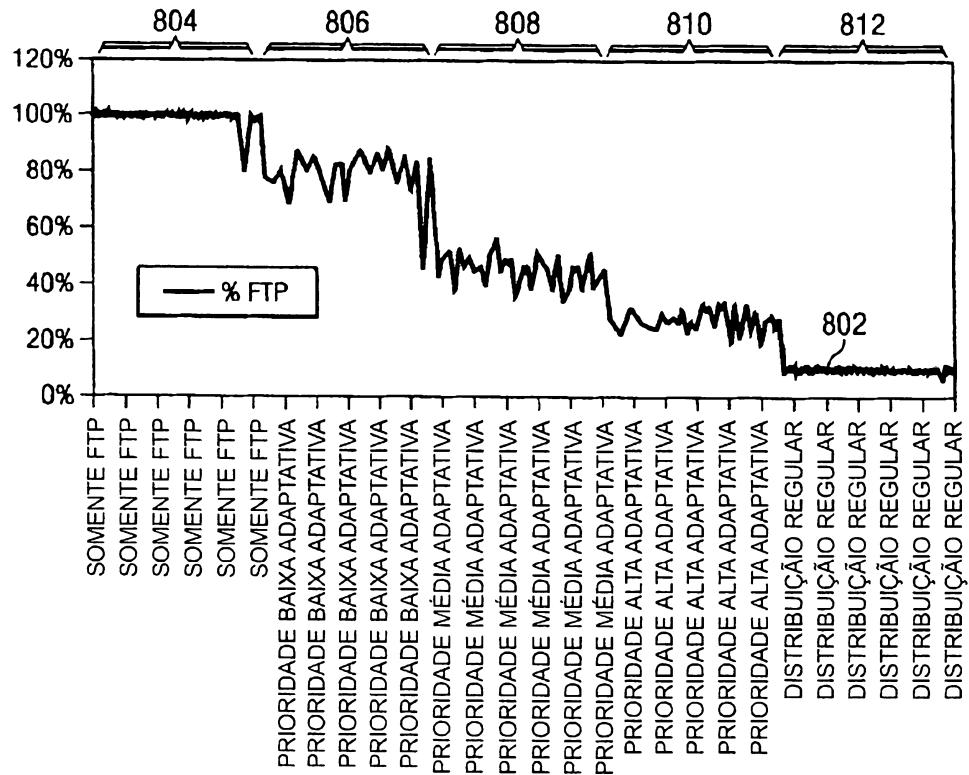


FIGURA 8

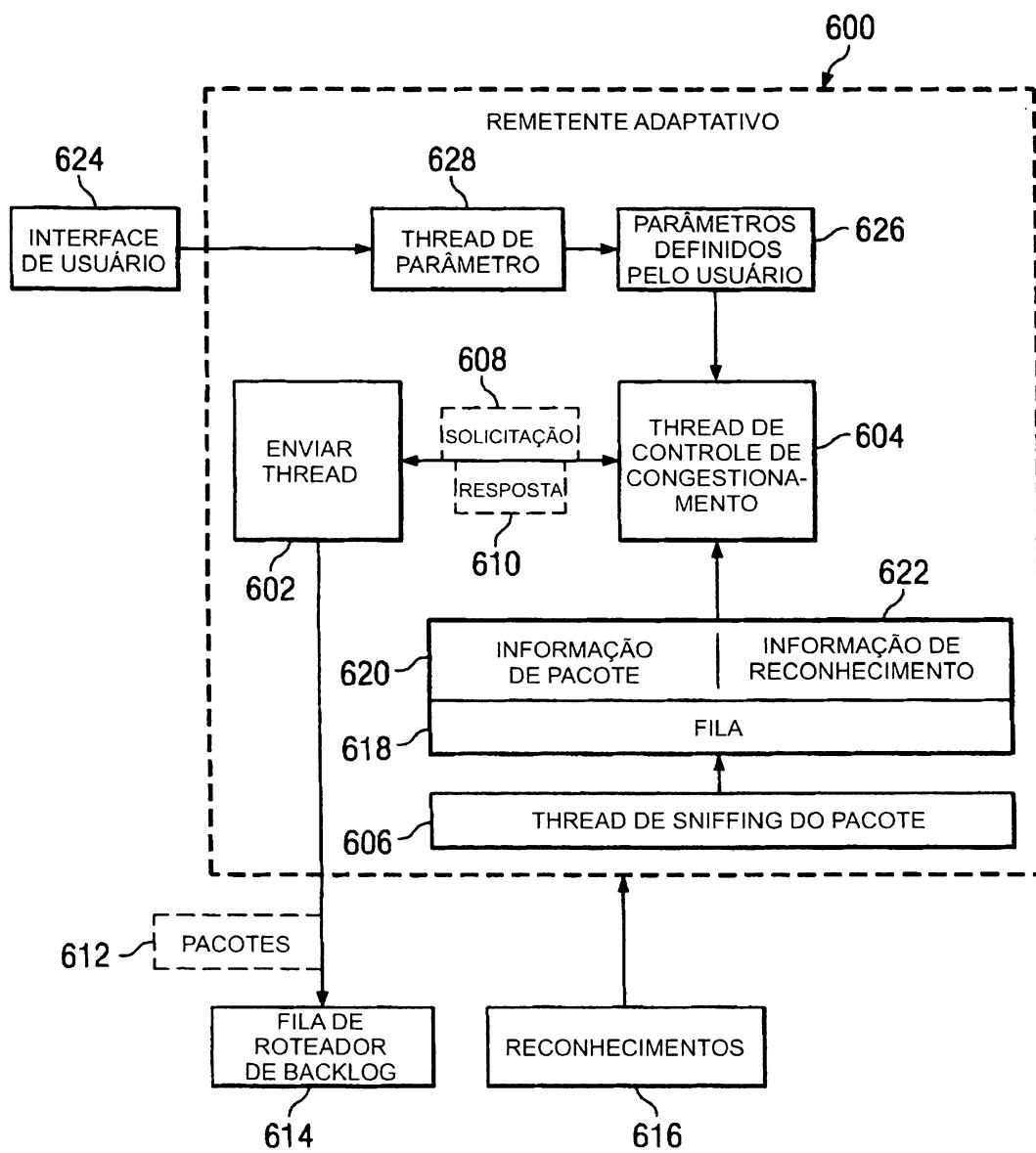


FIGURA 6

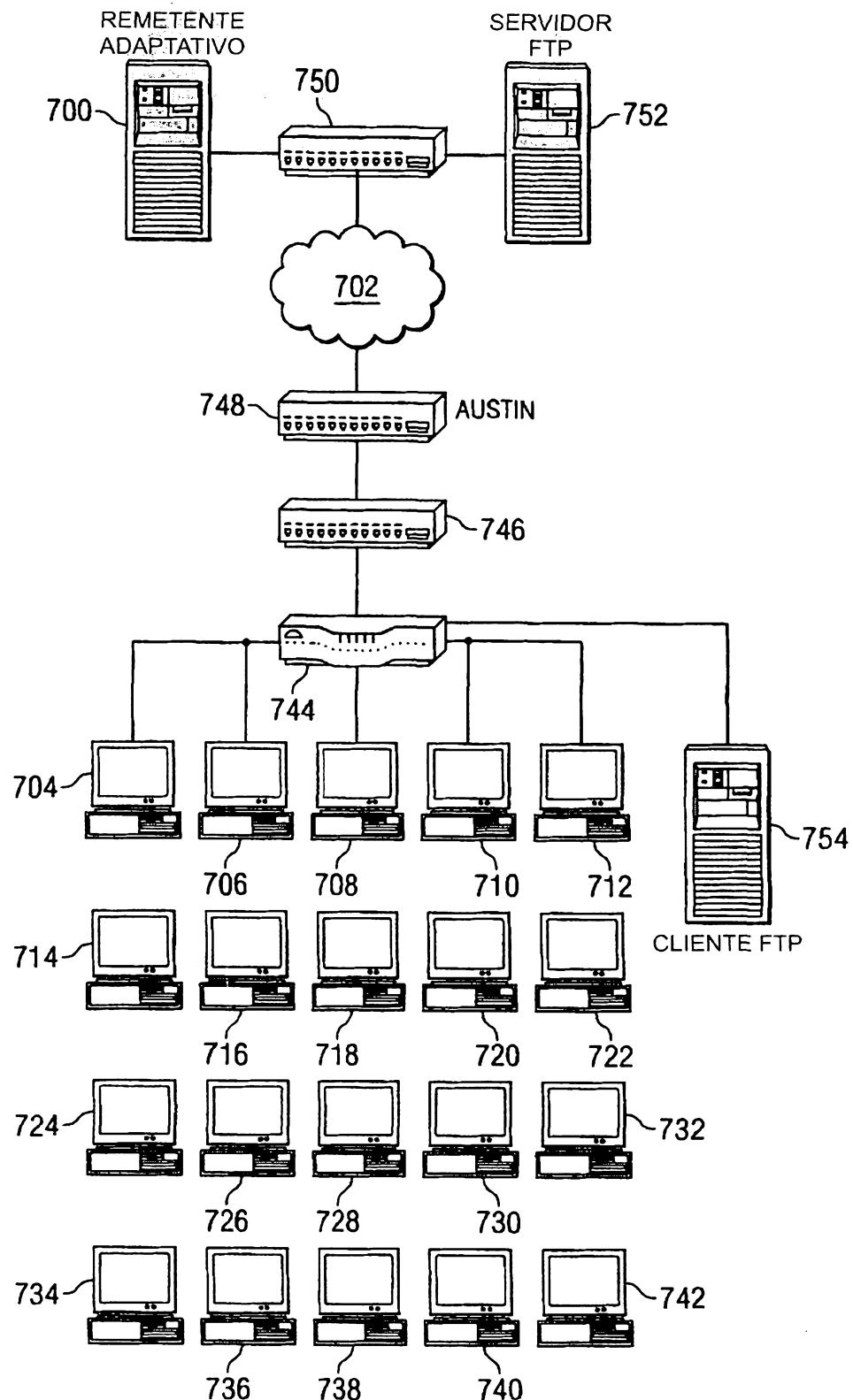


FIGURA 7

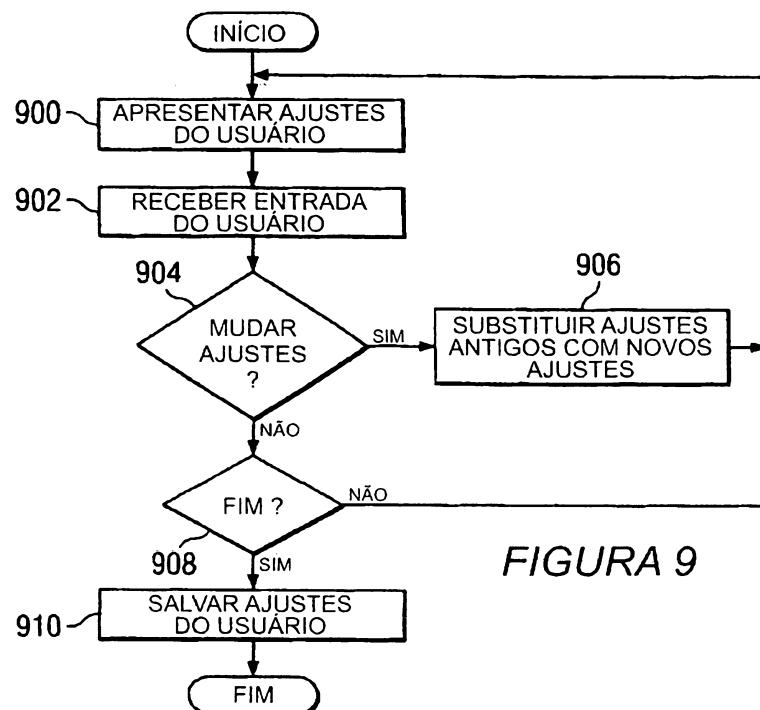
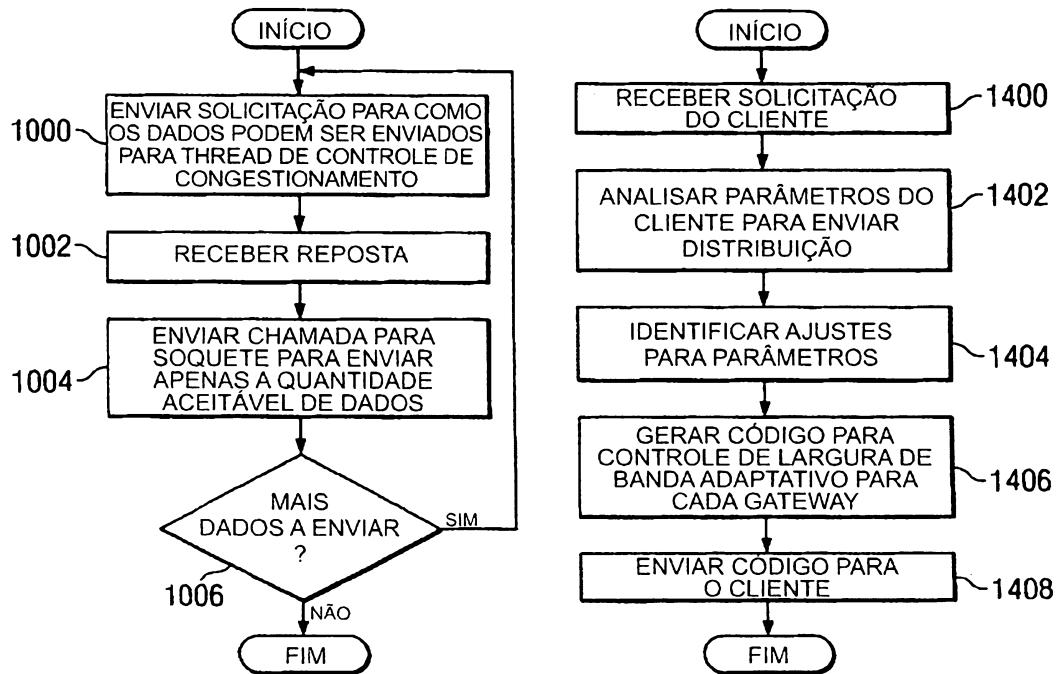
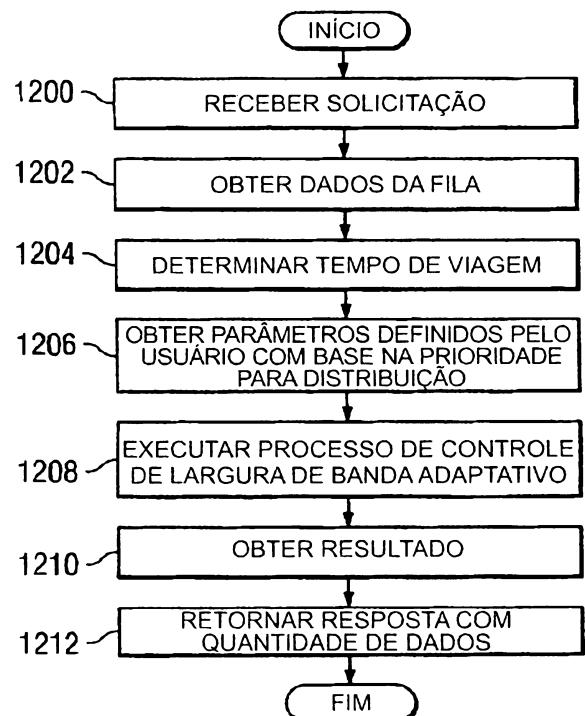
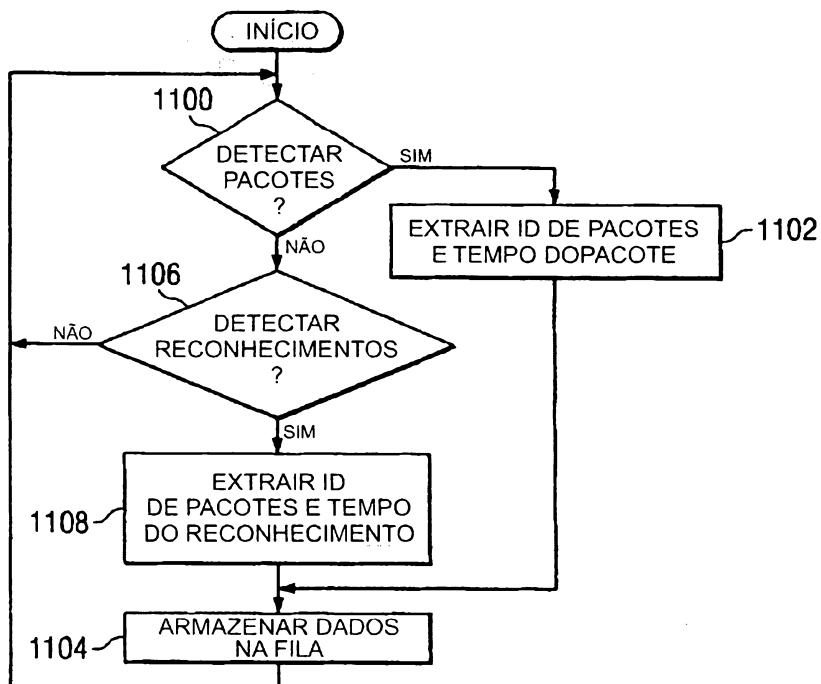


FIGURA 9





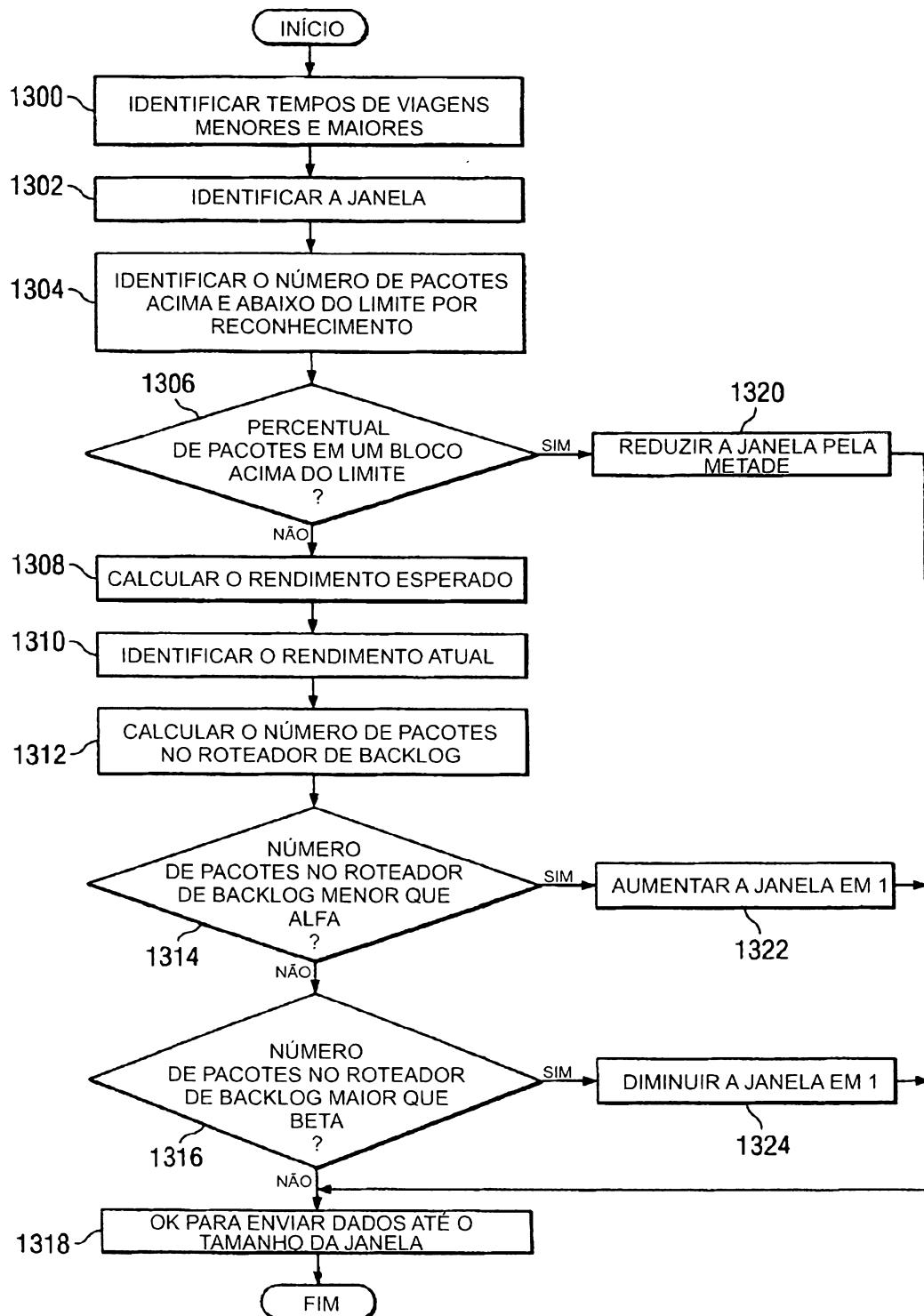


FIGURA 13