

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2014-127790

(P2014-127790A)

(43) 公開日 平成26年7月7日(2014.7.7)

(51) Int.Cl.	F I	テーマコード (参考)
HO4L 12/707 (2013.01)	HO4L 12/56 100A	5B089
HO4L 29/04 (2006.01)	HO4L 13/00 303B	5K030
HO4W 80/06 (2009.01)	HO4W 80/06	5K034
HO4W 28/10 (2009.01)	HO4W 28/10	5K067
GO6F 13/00 (2006.01)	GO6F 13/00 353A	

審査請求 未請求 請求項の数 6 O L (全 25 頁)

(21) 出願番号 特願2012-282146 (P2012-282146)
 (22) 出願日 平成24年12月26日 (2012.12.26)

特許法第30条第2項適用申請有り (1) 遅延差の大きなネットワークに適したマルチパスTCPパケット分配方式の提案と評価 平成24年度 電気通信大学 情報理工学研究所 情報・通信工学専攻 コンピュータサイエンスコース 修士論文中間発表会において発表 平成24年9月24日発表 (2) 遅延差の大きなネットワークにおけるマルチパスTCPパケット分配方式の提案 一般社団法人電子情報通信学会、電子情報通信学会技術研究報告、Vol. 112、No. 287、平成24年11月8日発行 (3) 遅延差の大きなネットワークにおけるマルチパスTCPパケット分配方式の提案 電子情報通信学会ネットワークシステム研究会において発表 平成24年11月15日発表

(71) 出願人 504133110
 国立大学法人電気通信大学
 東京都調布市調布ヶ丘一丁目5番地1
 (74) 代理人 100082131
 弁理士 稲本 義雄
 (74) 代理人 100121131
 弁理士 西川 孝
 (72) 発明者 及川 永寿
 東京都調布市調布ヶ丘一丁目5番地1 国立大学法人電気通信大学内
 (72) 発明者 中山 泰一
 東京都調布市調布ヶ丘一丁目5番地1 国立大学法人電気通信大学内
 Fターム(参考) 5B089 GA11 GA25 GB01 HA10 HA13
 HB02 KA07 KC02 KD01 KE09
 最終頁に続く

(54) 【発明の名称】 情報処理装置および方法、並びにプログラム

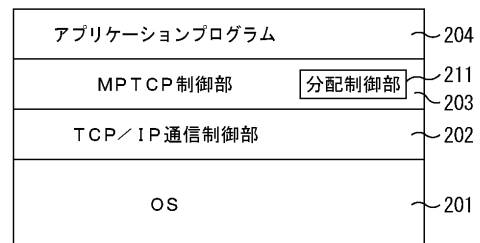
(57) 【要約】

【課題】 伝送遅延の差が大きい複数の経路を利用したマルチパス通信において、より効率的に帯域幅を活用することができるようにする。

【解決手段】 OS (オペレーティングシステム 201) 上で実行されるTCP/IP通信制御部 202、および、MPTCP制御部 203 が設けられている。TCP/IP通信制御部 202、および、MPTCP制御部 203 の上位層としてアプリケーションプログラム 204 が実行される。MPTCP制御部 203 の一部として分配制御部 211 が設けられ、各ネットワークインタフェースから送信されるデータに係る分配を制御するようになされている。分配制御部 211 は、サブフロー毎に伝送遅延を特定し、輻輳ウィンドウサイズを予測しながら、到達予想時間を算出/更新し、到達予想時間が最小となるサブフローにセグメントを送出する。

【選択図】 図6

図6



【特許請求の範囲】**【請求項 1】**

複数の TCP コネクションをまとめた単一のコネクションを確立し、API (Application Program Interface) により、アプリケーションプログラムに提供する情報処理装置であって、

前記複数の TCP コネクションのそれぞれに対応するサブフローを介して送信されるセグメントを蓄積するキューに対応づけられた到達予想時間に基づいて、送信バッファから取り出したセグメントを挿入するキューを選択するキュー選択部と、

前記キューから前記セグメントが送出されるときに輻輳ウィンドウサイズを予測するウィンドウサイズ予測部と、

前記サブフローの遅延時間、および、前記輻輳ウィンドウサイズに基づいて、前記キューに挿入されたセグメントが前記キューから送出されるまでに要すると推測される送出予想時間を算出する送出予想時間算出部と、

前記送出予想時間に基づいて、前記キューに対応づけられた到達予想時間を更新する更新部と、

前記サブフローのいずれかにおいて輻輳またはパケットロスが発生した場合、少なくとも前記輻輳またはパケットロスの発生したサブフローに対応するキューに対応付けられた到達予想時間を再計算するとともに、前記輻輳またはパケットロスの発生したサブフローに対応するキューに蓄積されたセグメントの少なくとも一部を、他のサブフローに対応するキューに分配する輻輳時キュー制御部と

を備える情報処理装置。

【請求項 2】

前記輻輳時キュー制御部は、

前記輻輳またはパケットロスの発生に伴って縮小された輻輳ウィンドウサイズを初期値として取得し、

前記輻輳またはパケットロスの発生したサブフローのキューに、前記キューに蓄積されていたセグメントを 1 つずつ仮想的に挿入し、前記キューに対応づけられた到達予想時間を、前記セグメント毎に再計算し、

前記再計算により、前記輻輳またはパケットロスの発生したサブフロー以外のサブフローに対応するキューのそれぞれに対応づけられた到達予想時間のうち、最大の到達予想時間を超える到達予想時間が得られた場合、そのセグメントより送信順が後のセグメントを、前記輻輳またはパケットロスの発生したサブフロー以外のサブフローに対応するキューに分配する

請求項 1 に記載の情報処理装置。

【請求項 3】

前記複数のサブフローは、それぞれ伝送遅延が異なるネットワーク上でのデータの送受信により生成された TCP コネクションに対応する

請求項 1 に記載の情報処理装置。

【請求項 4】

前記複数の TCP コネクションのうち少なくともいずれか 1 つは、広域無線通信によるネットワークを利用する

請求項 1 に記載の情報処理装置。

【請求項 5】

複数の TCP コネクションをまとめた単一のコネクションを確立し、API (Application Program Interface) により、アプリケーションプログラムに提供する情報処理装置の情報処理方法であって、

キュー選択部が、前記複数の TCP コネクションのそれぞれに対応するサブフローを介して送信されるセグメントを蓄積するキューに対応づけられた到達予想時間に基づいて、送信バッファから取り出したセグメントを挿入するキューを選択し、

ウィンドウサイズ予測部が、前記キューから前記セグメントが送出されるときに輻輳ウ

10

20

30

40

50

インドウサイズを予測し、

送出予想時間算出部が、前記サブフローの遅延時間、および、前記輻輳ウィンドウサイズに基づいて、前記キューに挿入されたセグメントが前記キューから送出されるまでに要すると推測される送出予想時間を算出し、

更新部が、前記送出予想時間に基づいて、前記キューに対応づけられた到達予想時間を更新し、

輻輳時キュー制御部が、前記サブフローのいずれかにおいて輻輳またはパケットロスが発生した場合、少なくとも前記輻輳またはパケットロスの発生したサブフローに対応するキューに対応づけられた到達予想時間を再計算するとともに、前記輻輳またはパケットロスの発生したサブフローに対応するキューに蓄積されたセグメントの少なくとも一部を、
他のサブフローに対応するキューに分配するステップ

を含む情報処理方法。

【請求項 6】

コンピュータを、

複数の TCP コネクションをまとめた単一のコネクションを確立し、API (Application Program Interface) により、アプリケーションプログラムに提供する情報処理装置であって、

前記複数の TCP コネクションのそれぞれに対応するサブフローを介して送信されるセグメントを蓄積するキューに対応づけられた到達予想時間に基づいて、送信バッファから取り出したセグメントを挿入するキューを選択するキュー選択部と、

前記キューから前記セグメントが送出されるときに輻輳ウィンドウサイズを予測するウィンドウサイズ予測部と、

前記サブフローの遅延時間、および、前記輻輳ウィンドウサイズに基づいて、前記キューに挿入されたセグメントが前記キューから送出されるまでに要すると推測される送出予想時間を算出する送出予想時間算出部と、

前記送出予想時間に基づいて、前記キューに対応づけられた到達予想時間を更新する更新部と、

前記サブフローのいずれかにおいて輻輳またはパケットロスが発生した場合、少なくとも前記輻輳またはパケットロスの発生したサブフローに対応するキューに対応づけられた到達予想時間を再計算するとともに、前記輻輳またはパケットロスの発生したサブフローに対応するキューに蓄積されたセグメントの少なくとも一部を、他のサブフローに対応するキューに分配する輻輳時キュー制御部とを備える情報処理装置として機能させる

プログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本技術は、情報処理装置および方法、並びにプログラムに関し、特に、伝送遅延の差が大きい複数の経路を利用したマルチパス通信において、より効率的に帯域幅を活用することができるようにする情報処理装置および方法、並びにプログラムに関する。

【背景技術】

【0002】

近年、無線通信インフラの多様化により、スマートフォン等のモバイル機器には複数のネットワークインタフェースが搭載されることが多い。例えば、モバイル機器において、Bluetooth (登録商標)、Wi-Fi などの近距離無線通信を行いつつ、3G (IMT-2000)、WiMAX (Worldwide Interoperability for Microwave Access) などの広域無線通信を行うことが提案されている。

【0003】

すなわち、物理層、データリンク層、または、ネットワーク層において、それぞれ異なる経路で同一の宛先と通信する、マルチパス通信が提案されている。

【0004】

10

20

30

40

50

例えば、1つのアプリケーションプログラムが、複数のネットワークインタフェースを効率的に利用したマルチパス通信を行うことが可能であれば、アプリケーションプログラムの処理に係る時間を短縮することも可能になる。しかし、既存のトランスポート層プロトコルであるTCPでは、複数のインタフェースを同時に利用する負荷分散はできず、利用可能な帯域を十分に活用できない。

【0005】

また、TCPセッションは、IPアドレスと1対1で対応していることから、ネットワーク層での接続が失われるとTCPセッションが切断されることになる。

【0006】

例えば、Wi-Fiで通信中にエリア外に移動した場合、ネットワーク層での接続が一旦失われることになるが、この際に都度、TCPセッションが切断されると、アプリケーションの処理などに支障をきたしかねない。このような問題は、例えば、ユビキタスネットワークの発展を阻害する要因となり得る。

10

【0007】

このような問題を解決するために提案されたプロトコルの1つに、マルチパスTCP(MPTCP)がある。MPTCPは、複数のネットワークインタフェースを同時に利用することで、マルチホーミング(通信経路の冗長性確保や負荷分散)を実現するトランスポート層プロトコルである。MPTCPは、既存のTCPを拡張する(ヘッダ内のオプションを利用する)ものであるため、世界中に広く普及している既存のファイヤウォールやNATの変更なしに用いることが可能である。

20

【0008】

MPTCPでは、リソース利用効率と冗長性を高めるために、同一宛先に対してTCPコネクション(リンクとも称する)を複数経路に確立することができ、複数のTCPコネクションを1つにまとめて上位層とのインタフェースを提供する。

【0009】

一方、ネットワーク層でのマルチホーミングの方式として、SHAKE(ShARing multi path procedure for a cluster network Environment)が提案されている(例えば、非特許文献1参照)。

【0010】

SHAKEでは、Mobile IPをベースとしてホームエージェント(HA)にトラヒック分配機構を実装し、移動端末(MN)とHAとの間の帯域幅を集約する。

30

【0011】

MNがセグメントを送信する際には、パケットが順序通りに到達するよう、最も早くHAへ到着すると推測される経路を用いてセグメントを送信する。リンクには仮想的な送信キューを設け、キューから出力されるまでの待ち時間と伝送遅延からパケット到達時刻を予想し各経路へ振り分けるようになっている。

【先行技術文献】

【非特許文献】

【0012】

【非特許文献1】K. Koyama, Y. Ito, H. Mineno and S. Ishihara, Performance Evaluation of TCP on Mobile IP SHAKE," IPSJ Journal, Vol. 45, No.10, pp. 2270{2278, Oct. 2004.

40

【発明の開示】

【発明が解決しようとする課題】

【0013】

ところで、MPTCPでは、リンク性能差(伝送遅延や帯域幅)が大きいネットワークを同時に用いた通信では、送信側の経路選択に応じて受信側においてパケット到着の順序が大きく入れ替わる。しかし、受信側のアプリケーションプログラムでは、所定の順番でパケットを処理する必要があるため、到着したパケットを並べ替えるために、受信側でパケットを一時的に保管するバッファが必要となる。

50

【 0 0 1 4 】

つまり、伝送遅延が大きいネットワークを介して送信されるパケットは、受信側に到着する時間に遅れが生じるため、伝送遅延が小さいネットワークを介して送信されるパケットより後に到着する。そうすると、送信側から送信した順番とは異なる順番で受信側にパケットが到着することがある。

【 0 0 1 5 】

また、M P T C Pでは、送信側でのパケットの分配方式が受信側での並べ替え時間に影響を与えるため、分配方式の如何により受信側のアプリケーションのスループットが不安定になりやすい。すなわち、送信側でのパケットの分配が適切に行われないと、全リンクの帯域幅を有効に活用できない場合もある。

10

【 0 0 1 6 】

この点、非特許文献1の技術では、キューから出力されるまでの待ち時間と伝送遅延に基づいてパケット到達時刻を予測したパケットの分配が行われる。しかしながら、輻輳やパケットロスなどが発生した場合、到達時刻を精度高く予測することは極めて困難であり、リンク毎に伝送遅延が異なる場合、やはり全リンクの帯域幅を有効に活用することは難しかった。

【 0 0 1 7 】

本技術はこのような状況に鑑みて開示するものであり、伝送遅延の差が大きい複数の経路を利用したマルチパス通信において、より効率的に帯域幅を活用することができるようにするものである。

20

【 課題を解決するための手段 】

【 0 0 1 8 】

本技術の一側面は、複数のT C Pコネクションをまとめた単一のコネクションを確立し、A P I (Application Program Interface)により、アプリケーションプログラムに提供する情報処理装置であって、前記複数のT C Pコネクションのそれぞれに対応するサブフローを介して送信されるセグメントを蓄積するキューに対応づけられた到達予想時間に基づいて、送信バッファから取り出したセグメントを挿入するキューを選択するキュー選択部と、前記キューから前記セグメントが送出される際の輻輳ウィンドウサイズを予測するウィンドウサイズ予測部と、前記サブフローの遅延時間、および、前記輻輳ウィンドウサイズに基づいて、前記キューに挿入されたセグメントが前記キューから送出されるまでに要すると推測される送出予想時間を算出する送出予想時間算出部と、前記送出予想時間に基づいて、前記キューに対応づけられた到達予想時間を更新する更新部と、前記サブフローのいずれかにおいて輻輳またはパケットロスが発生した場合、少なくとも前記輻輳またはパケットロスの発生したサブフローに対応するキューに対応付けられた到達予想時間を再計算するとともに、前記輻輳またはパケットロスの発生したサブフローに対応するキューに蓄積されたセグメントの少なくとも一部を、他のサブフローに対応するキューに分配する輻輳時キュー制御部とを備える情報処理装置である。

30

【 0 0 1 9 】

前記輻輳時キュー制御部は、前記輻輳またはパケットロスの発生に伴って縮小された輻輳ウィンドウサイズを初期値として取得し、前記輻輳またはパケットロスの発生したサブフローのキューに、前記キューに蓄積されていたセグメントを1つずつ仮想的に挿入し、前記キューに対応づけられた到達予想時間を、前記セグメント毎に再計算し、前記再計算により、前記輻輳またはパケットロスの発生したサブフロー以外のサブフローに対応するキューのそれぞれに対応づけられた到達予想時間のうち、最大の到達予想時間を超える到達予想時間が得られた場合、そのセグメントより送信順が後のセグメントを、前記輻輳またはパケットロスの発生したサブフロー以外のサブフローに対応するキューに分配することができる。

40

【 0 0 2 0 】

前記複数のサブフローは、それぞれ伝送遅延が異なるネットワーク上でのデータの送受信により生成されたT C Pコネクションに対応するようにすることができる。

50

【 0 0 2 1 】

前記複数のTCPコネクションのうち少なくともいずれか1つは、広域無線通信によるネットワークを利用するようにすることができる。

【 0 0 2 2 】

本技術の一側面は、複数のTCPコネクションをまとめた単一のコネクションを確立し、API (Application Program Interface) により、アプリケーションプログラムに提供する情報処理装置の情報処理方法であって、キュー選択部が、前記複数のTCPコネクションのそれぞれに対応するサブフローを介して送信されるセグメントを蓄積するキューに対応づけられた到達予想時間に基づいて、送信バッファから取り出したセグメントを挿入するキューを選択し、ウィンドウサイズ予測部が、前記キューから前記セグメントが送出されるときの輻輳ウィンドウサイズを予測し、送出予想時間算出部が、前記サブフローの遅延時間、および、前記輻輳ウィンドウサイズに基づいて、前記キューに挿入されたセグメントが前記キューから送出されるまでに要すると推測される送出予想時間を算出し、更新部が、前記送出予想時間に基づいて、前記キューに対応づけられた到達予想時間を更新し、輻輳時キュー制御部が、前記サブフローのいずれかにおいて輻輳またはパケットロスが発生した場合、少なくとも前記輻輳またはパケットロスの発生したサブフローに対応するキューに対応付けられた到達予想時間を再計算するとともに、前記輻輳またはパケットロスの発生したサブフローに対応するキューに蓄積されたセグメントの少なくとも一部を、他のサブフローに対応するキューに分配するステップを含む情報処理方法である。

10

【 0 0 2 3 】

本技術の一側面は、コンピュータを、複数のTCPコネクションをまとめた単一のコネクションを確立し、API (Application Program Interface) により、アプリケーションプログラムに提供する情報処理装置であって、前記複数のTCPコネクションのそれぞれに対応するサブフローを介して送信されるセグメントを蓄積するキューに対応づけられた到達予想時間に基づいて、送信バッファから取り出したセグメントを挿入するキューを選択するキュー選択部と、前記キューから前記セグメントが送出されるときの輻輳ウィンドウサイズを予測するウィンドウサイズ予測部と、前記サブフローの遅延時間、および、前記輻輳ウィンドウサイズに基づいて、前記キューに挿入されたセグメントが前記キューから送出されるまでに要すると推測される送出予想時間を算出する送出予想時間算出部と

20

30

、前記送出予想時間に基づいて、前記キューに対応づけられた到達予想時間を更新する更新部と、前記サブフローのいずれかにおいて輻輳またはパケットロスが発生した場合、少なくとも前記輻輳またはパケットロスの発生したサブフローに対応するキューに対応付けられた到達予想時間を再計算するとともに、前記輻輳またはパケットロスの発生したサブフローに対応するキューに蓄積されたセグメントの少なくとも一部を、他のサブフローに対応するキューに分配する輻輳時キュー制御部とを備える情報処理装置として機能させるプログラムである。

【 0 0 2 4 】

本技術の一側面においては、前記複数のTCPコネクションのそれぞれに対応するサブフローを介して送信されるセグメントを蓄積するキューに対応づけられた到達予想時間に基づいて、送信バッファから取り出したセグメントを挿入するキューが選択され、前記キューから前記セグメントが送出されるときの輻輳ウィンドウサイズが予測され、前記サブフローの遅延時間、および、前記輻輳ウィンドウサイズに基づいて、前記キューに挿入されたセグメントが前記キューから送出されるまでに要すると推測される送出予想時間が算出され、前記送出予想時間に基づいて、前記キューに対応づけられた到達予想時間が更新され、前記サブフローのいずれかにおいて輻輳またはパケットロスが発生した場合、少なくとも前記輻輳またはパケットロスの発生したサブフローに対応するキューに対応付けられた到達予想時間を再計算するとともに、前記輻輳またはパケットロスの発生したサブフローに対応するキューに蓄積されたセグメントの少なくとも一部が、他のサブフローに対応するキューに分配される。

40

50

【発明の効果】

【0025】

本技術によれば、伝送遅延の差が大きい複数の経路を利用したマルチパス通信において、より効率的に帯域幅を活用することができる。

【図面の簡単な説明】

【0026】

【図1】従来のマルチパス通信を説明する図である。

【図2】図1のマルチパス通信におけるパケットの分配方式を説明する図である。

【図3】図1のマルチパス通信におけるパケットの分配方式を説明する図である。

【図4】本技術を適用したネットワークシステムの構成を示す図である。

10

【図5】図4のスマートフォンの構成例を示すブロック図である。

【図6】図5のCPUで実行されるソフトウェアの機能的構成を説明する図である。

【図7】MPTCP制御部によるマルチパス通信の方式を説明する図である。

【図8】分配制御部によるデータの分配の方式を説明する図である。

【図9】輻輳、または、パケットロスが発生した場合の分配制御部によるデータの分配の方式を説明する図である。

【図10】セグメント分配処理の例を説明するフローチャートである。

【図11】セグメント退避処理の例を説明するフローチャートである。

【図12】遅延差の小さい複数のネットワークを用いた従来のマルチパス通信における受信側のリオーダー待ちデータの長さ、アプリケーションで処理可能なデータの送達レートとの関係を示す図である。

20

【図13】遅延差の大きい複数のネットワークを用いた従来のマルチパス通信における受信側のリオーダー待ちデータの長さ、アプリケーションで処理可能なデータの送達レートとの関係を示す図である。

【図14】図12および図13の測定結果を得るために用いたネットワークシステムの構成例を示す図である。

【図15】パーソナルコンピュータの構成例を示すブロック図である。

【発明を実施するための形態】

【0027】

以下、図面を参照して、ここで開示する技術の実施の形態について説明する。

30

【0028】

最初に従来の技術の問題点を説明する。

【0029】

図1は、従来のマルチパス通信を説明する図である。同図に示されるネットワークシステム10においては、スマートフォン11とサーバ12がマルチパス通信を行うようになされている。

【0030】

この例では、スマートフォン11は、Bluetooth(登録商標)、Wi-Fiなどの近距離無線通信を行うためのネットワークインタフェースと、3G、WiMAX(Worldwide Interoperability for Microwave Access)などの広域無線通信を行うネットワークインタフェースとを有している。

40

【0031】

近距離無線通信では、スマートフォン11から送信されたデータは、アクセスポイント21を経由してインターネット31に送出され、サーバ12に受信される。一方、広域無線通信では、スマートフォン11から送信されたデータは、基地局23を経由してインターネット31に送出され、サーバ12に受信される。

【0032】

ここで、近距離無線通信を行うネットワークインタフェースは、広帯域で遅延の小さい通信環境を提供するものとし、広域無線通信を行うネットワークインタフェースは、狭帯域で遅延の大きい通信環境を提供するものとする。

50

【 0 0 3 3 】

スマートフォン 1 1 は、より効率的な通信を行うために、サーバ 1 2 との通信において、近距離無線通信および広域無線通信のそれぞれのネットワークインタフェースを同時に利用したマルチパス通信を行う。このため、スマートフォン 1 1 は、自分の送信バッファに蓄積されたパケットを、それぞれのネットワークインタフェースに分配する。

【 0 0 3 4 】

図 1 の例では、送信バッファには、「 1 」、「 2 」、および、「 3 」のパケットが蓄積されており、パケットを表す番号は、サーバ 1 2 で実行されるアプリケーションプログラムにおける当該パケットの処理順序に対応している。

【 0 0 3 5 】

例えば、図 2 に示されるように、スマートフォン 1 1 は、パケット「 1 」を、広域無線通信を行うネットワークインタフェースに送出し、パケット「 2 」およびパケット「 3 」を、近距離無線通信を行うネットワークインタフェースに送出する。なお、広帯域の近距離無線通信には、2 つのパケットが割り振られたのに対し、狭帯域の広域無線通信には、1 つのパケットが割り振られている。

【 0 0 3 6 】

例えば、マルチパス TCP (M P T C P) を実装したスマートフォンおよびサーバ 1 2 によれば、上述したようなパケットの分配が行われる。M P T C P では、リソース利用率と冗長性を高めるために、同一宛先に対して TCP コネクション (リンクとも称する) を複数経路に確立することができ、複数の TCP コネクションを 1 つにまとめて上位層とのインタフェースを提供する。

【 0 0 3 7 】

しかしながら、スマートフォン 1 1 により、このようなパケットの分配が行われた場合、図 3 に示されるように、パケット「 1 」より先に、パケット「 2 」およびパケット「 3 」が先にサーバ 1 2 に到着する可能性が高い。広域無線通信は、近距離無線通信と比較して遅延が大きいいため、サーバ 1 2 で受信されるまでに長い時間を要するからである。このような場合、サーバ 1 2 では、パケット「 1 」が到着するまで、パケット「 2 」およびパケット「 3 」を処理することができない。このため、サーバ 1 2 では、受信したパケット「 2 」、パケット「 3 」をバッファに蓄積し、パケット「 1 」を受信後に、これらのパケットを並べ替えに係る処理待ち (リオーダ待ち) が発生する。

【 0 0 3 8 】

このようなリオーダ待ちが発生すると、サーバ 1 2 のアプリケーションプログラムが使用できる情報の伝送効率であるグッドプットが低下し、アプリケーションプログラムの処理速度の低下を招くことになる。

【 0 0 3 9 】

M P T C P とは異なる方式でパケットを分配する技術も提案されている。例えば、一方、ネットワーク層でのマルチホーミングの方式として、S H A K E (S H A R i n g m u l t i p a t h p r o c e d u r e f o r a c l u s t e r n e t w o r k E n v i r o n m e n t) が提案されている。S H A K E では、キューから出力されるまでの待ち時間と伝送遅延に基づいてパケット到達時刻を予測したパケットの分配が行われる。

【 0 0 4 0 】

しかしながら、輻輳やパケットロスなどが発生した場合、到達時刻を精度高く予測することは極めて困難であり、リンク毎に伝送遅延が異なる場合、やはり全リンクの帯域幅を有効に活用することは難しかった。

【 0 0 4 1 】

そこで、本技術では、輻輳やパケットロスなどが発生した場合であっても、極力リオーダ待ちを回避できるようにする。

【 0 0 4 2 】

図 4 は、本技術を適用したネットワークシステムの構成を示す図である。このネットワークシステム 1 0 0 においては、スマートフォン 1 1 1 とサーバ 1 1 2 がマルチパス通信

10

20

30

40

50

を行うようになされている。

【0043】

この例では、スマートフォン111は、Bluetooth(登録商標)、Wi-Fiなどの近距離無線通信を行うためのネットワークインタフェースと、3G、WiMAX(Worldwide Interoperability for Microwave Access)などの広域無線通信を行うネットワークインタフェースとを有している。さらに、スマートフォン111は、有線通信であるLANのネットワークインタフェースを有している。

【0044】

近距離無線通信では、スマートフォン111から送信されたデータは、アクセスポイント121を経由してインターネット131に送出され、サーバ112に受信される。また、有線通信では、スマートフォン111から送信されたデータは、ルータ122を経由してインターネット131に送出され、サーバ112に受信される。広域無線通信では、スマートフォン111から送信されたデータは、基地局123を経由してインターネット131に送出され、サーバ112に受信される。

10

【0045】

図5は、図4のスマートフォン111の構成例を示すブロック図である。

【0046】

図5において、スマートフォン111のCPU(Central Processing Unit)151は、ROM(Read Only Memory)152に記憶されているプログラム、または記憶部163からRAM(Random Access Memory)153にロードされたプログラムに従って各種の処理を実行する。RAM153にはまた、CPU151が各種の処理を実行する上において必要なデータなども適宜記憶される。

20

【0047】

CPU151、ROM152、およびRAM153は、バス154を介して相互に接続されている。このバス154にはまた、入出力インタフェース160も接続されている。

【0048】

入出力インタフェース160には、キーボード、マウスなどよりなる入力部161、LCD(Liquid Crystal Display)などよりなるディスプレイ、並びにスピーカなどよりなる出力部162、および、ハードディスクやフラッシュメモリなどより構成される記憶部163が接続されている。

30

【0049】

入出力インタフェース160にはまた、必要に応じてドライブ165が接続され、磁気ディスク、光ディスク、光磁気ディスク、或いは半導体メモリなどのリムーバブルメディア166が適宜装着され、それらから読み出されたコンピュータプログラムが、必要に応じて記憶部163にインストールされる。

【0050】

入出力インタフェース160にはさらに、広域無線通信部171、有線通信部172、および近距離無線通信部173が接続されている。

【0051】

広域無線通信部171は、基地局123と無線通信を行い、広域通信網を介した通信を行う無線通信デバイスである。広域無線通信部171は、例えば2GHzの周波数帯を使い、通話アプリケーションだけでなく、最大2Mbpsのデータ通信を用いてインターネット接続など各種通信アプリケーションに利用される。広域無線通信部171は、例えば、いわゆる第3世代携帯電話の通信方式による通信が可能なデバイスなどとして構成される。

40

【0052】

有線通信部172は、例えば、イーサネット(登録商標)などのLANに対応するインタフェースとされ、最大100Mbpsのデータ通信を行う。

【0053】

近距離無線通信部173は、例えば、Bluetooth(登録商標、BTとも称する)やIEEE(Institute of Electrical and Electronic Engineers)802.11x等の近距離無線通信デ

50

バイスである。ここで、近距離無線通信とは、通信可能最大距離が数メートル乃至数十メートル程度のローカルな（狭域の）無線通信を意味する。通信規格は任意である。例えば、近距離無線通信部 173 が BT の通信を行うものである場合は、アンテナを経由して 2.4GHz 帯にて最大通信速度 3 Mビット / 秒（バージョン 2.0 + EDR 以降）の通信を行う。

【0054】

スマートフォン 111 の各部は、CPU 151 により制御される。制御プログラムの実行バイナリコードは ROM 152 や記憶部 163 に保存されており、各種演算処理のためスタック、ヒープ領域は RAM 153 上に展開される。

【0055】

図 6 は、図 5 の CPU 151 で実行されるソフトウェアの機能的構成を説明する図である。

10

【0056】

図 6 の例では、OS（オペレーティングシステム 201）上で実行される TCP/IP 通信制御部 202、および、MPTCP 制御部 203 が設けられている。そして、TCP/IP 通信制御部 202、および、MPTCP 制御部 203 の上位層としてアプリケーションプログラム 204 が実行されるようになされている。

【0057】

また、図 6 に示されるように、MPTCP 制御部 203 の一部として分配制御部 211 が設けられている。分配制御部 211 は、後述するように、各ネットワークインタフェースから送信されるデータに係る分配を制御するようになされている。

20

【0058】

図 7 は、MPTCP 制御部 203 によるマルチパス通信の方式を説明する図である。

【0059】

図 7 に示されるように、MPTCP では、広域無線通信部 171 による広域無線通信における TCP コネクション、有線通信部 172 による有線通信における TCP コネクション、および、近距離無線通信部 173 による近距離無線通信における TCP コネクションのそれぞれが TCP サブフローとして MPTCP コアにより管理される。そして、MPTCP コアが 3 つの TCP サブフローをまとめて単一の MPTCP コネクションを確立し、それをソケット API（Application Program Interface）により、アプリケーションプログラムに提供する。なお、ここでは、TCP サブフローの数を 3 としているが、2 以上のいくつかであってもよい。

30

【0060】

図 8 は、分配制御部 211 によるデータの分配の方式を説明する図である。

【0061】

図 8 の例では、送信バッファ 231 に送信すべきデータが蓄積されており、分配制御部 211 が、送信バッファ 231 からデータを取り出して、サブフロー SF1 乃至サブフロー SFN に割り当てられたキュー 241 - 1 乃至キュー 241 - N に分配している。

【0062】

なお、分配制御部 211 は、トランスポート層における処理を実行する機能ブロックであるから、キュー 241 - 1 乃至キュー 241 - N に分配されるデータは、セグメントと称される。

40

【0063】

また、図中において送信バッファ 231 内の「8」、「9」・・・「12」の番号が付された矩形のそれぞれ、および、キュー 241 - 1 乃至キュー 241 - N 内の「1」、「2」、・・・「7」の番号が付された矩形のそれぞれがセグメントを表している。これらの 12 個のセグメントのそれぞれは、同じデータサイズであるものとし、例えば、TCP の最大セグメントサイズ（Maximum Segment Size）のデータとされる。さらに、各セグメントに付された番号は、それぞれのセグメントを送信すべき順番を表しているものとする。

【0064】

50

なお、図 8 の例では、サブフロー S F 1 は、遅延が小さいサブフローであり、サブフロー S F 2 は、サブフロー S F 1 より遅延が大きいサブフローであり、・・・サブフロー S F N はサブフロー S F N - 1 より遅延が大きいサブフローであるものとする。例えば、図 4 のスマートフォン 1 1 1 の場合、サブフロー S F 1 乃至サブフロー S F 3 が存在することになり、サブフロー S F 1 は有線通信に対応するサブフローとされ、サブフロー S F 2 は近距離無線通信に対応するサブフローとされ、サブフロー S F 3 は広域無線通信に対応するサブフローとされる。

【 0 0 6 5 】

分配制御部 2 1 1 は、キュー最後尾のセグメントが受信側に到達するまでの到達予想時間 D を算出し、到達予想時間 D が最も小さいキューを特定する。分配制御部 2 1 1 は、キューに格納されていたセグメントが出力され、ネットワーク層のプロトコルへ渡るときの輻輳ウィンドウを予測し、精度の高い到達予想時間 D を算出する。

10

【 0 0 6 6 】

T C P の輻輳制御アルゴリズムによれば、受信側から送出された A C K を送信側で受信した場合（輻輳が発生しなかった場合）、予め設定された割合で輻輳ウィンドウサイズが拡大される。すなわち、輻輳が発生しなかった場合、ネットワークの帯域に余裕があるとみなして伝送速度を上げるようになされている。

【 0 0 6 7 】

一方、T C P の輻輳制御アルゴリズムによれば、輻輳が発生した場合、または、パケットロスが発生した場合、輻輳ウィンドウサイズが縮小される。すなわち、輻輳またはパケットロスが発生した場合、輻輳またはパケットロスを回避するために伝送速度を下げるようになされている。

20

【 0 0 6 8 】

分配制御部 2 1 1 は、到達予想時間 D を次のようにして算出する。

【 0 0 6 9 】

まず、キューが空の場合、分配制御部 2 1 1 は、当該キューの到達予想時間 D を 0 に設定する。

【 0 0 7 0 】

分配制御部 2 1 1 は、サブフロー毎に、往復遅延時間 (R T T : round-trip time) を特定する。そして片道遅延時間 $d = R T T / 2$ を算出する。なお、セグメントが送信されてから A C K が受信されるまでの時間の平均値が往復遅延時間とされ、サブフロー毎に逐次更新されるようになされている。

30

【 0 0 7 1 】

そして、分配制御部 2 1 1 は、 $D + d$ が最小となるキューを選択し、そのキューにセグメントを挿入する。

【 0 0 7 2 】

また、分配制御部 2 1 1 は、これからキューに挿入されるセグメントがキューから出力される時の輻輳ウィンドウサイズ $expect_cwnd$ を予測する。なお、輻輳ウィンドウサイズは、最初は十分に小さい初期値が設定され、その後、拡大されていく。

【 0 0 7 3 】

上述したように、T C P の輻輳制御アルゴリズムによれば、輻輳が発生しなかった場合、予め設定された割合で輻輳ウィンドウサイズが拡大される。すなわち、サーバ 1 1 2 とのセグメントの送受信が行われる都度、輻輳ウィンドウサイズが徐々に拡大されていくようになされている。分配制御部 2 1 1 は、輻輳が発生しないことを前提として、現在の輻輳ウィンドウサイズと、キューに蓄積されている全てのセグメントを合計したデータサイズに基づいて、これからキューに挿入されるセグメントがキューから出力される時のウィンドウサイズを予測する。

40

【 0 0 7 4 】

さらに、分配制御部 2 1 1 は、式 (1) により送出予想時間 W を算出する。すなわち、1 つのセグメントがキューから送出されるまでに要すると推測される時間が、送出予想時

50

間 W として算出される。

【 0 0 7 5 】

【 数 1 】

$$W = \frac{RTT}{\frac{\text{expect-cwnd}}{MSS}} \dots (1)$$

【 0 0 7 6 】

なお、式 (1) における MSS は、TCP の最大セグメントサイズ (Maximum Segment Size) を意味している。また、送出予想時間 W は、サーバ 1 1 2 がセグメントを受信した際に送信する ACK を受信したときの到達予想時間 D の更新に用いられるため、メモリなどに記憶される。

10

【 0 0 7 7 】

そして、分配制御部 2 1 1 は、式 (2) により、セグメントが挿入されたキューの新たな到達予想時間 D_{new} を求める。

【 0 0 7 8 】

【 数 2 】

$$D_{new} = W + D \dots (2)$$

【 0 0 7 9 】

20

その後、分配制御部 2 1 1 は、式 (2) により得られた到達予想時間 D_{new} を、当該キューの到達予想時間とし、到達予想時間 D を更新する。また、サーバ 1 1 2 からの ACK を受信した場合、分配制御部 2 1 1 は、式 (3) により得られた到達予想時間 D_{new} を、当該キューの到達予想時間とし、到達予想時間 D を更新する。

【 0 0 8 0 】

【 数 3 】

$$D_{new} = D - W \dots (3)$$

【 0 0 8 1 】

30

なお、式 (3) における W は、当該 ACK に対応するセグメントをキューに挿入する際に算出した送出予想時間 W とされる。

【 0 0 8 2 】

そして、分配制御部 2 1 1 は、 $D + d$ が最小となるキューを選択し、そのキューに次のセグメントを挿入する。

【 0 0 8 3 】

このようにして、キュー 2 4 1 - 1 乃至キュー 2 4 1 - N のそれぞれに対応する到達予想時間 D が算出 / 更新されていくとともに、到達予想時間 D に基づいてセグメントが分配されていく。

【 0 0 8 4 】

40

本技術では、サブフロー毎に伝送遅延を特定し、輻輳ウィンドウサイズを予測しながら、到達予想時間を算出 / 更新するようにしたので、スマートフォン 1 1 1 から送信されたセグメントが、意図した順番にサーバ 1 1 2 に到着するように制御することが可能となる。例えば、図 8 に示される「 1 」乃至「 1 2 」の各セグメントが、サブフロー SF 1 乃至サブフロー SF N を介して送信され、「 1 」乃至「 1 2 」の番号順にサーバ 1 1 2 に受信されるようになる。

【 0 0 8 5 】

このように、本技術では、伝送遅延の異なるネットワークを利用したマルチパス通信において、受信側でのリオーダ待ちを回避して、アプリケーションプログラムの処理に要する時間を短縮させることができる。その結果、より効率的にネットワークの帯域幅を活用することができる。

50

【0086】

しかしながら、サブフローSF1乃至サブフローSF N において、輻輳、または、パケットロスが発生する場合がある。このような場合、スマートフォン111から送信された各セグメントが、「1」乃至「12」の番号順にサーバ112に受信されなくなる可能性が高い。

【0087】

そこで、分配制御部211は、サブフローSF1乃至サブフローSF N において、輻輳、または、パケットロスが発生した場合、次のようにして、セグメントの再分配を行う。

【0088】

最初に、分配制御部211は、輻輳、または、パケットロスが発生したサブフローを特定する。

10

【0089】

次に、分配制御部211は、特定されたサブフローの輻輳ウィンドウサイズを取得する。上述したように、輻輳制御アルゴリズムによれば、輻輳が発生した場合、または、パケットロスが発生した場合、輻輳ウィンドウサイズが縮小される。分配制御部211は、輻輳、または、パケットロスが発生したサブフローの縮小後の輻輳ウィンドウサイズ $cwnd$ を取得する。

【0090】

そして、分配制御部211は、輻輳ウィンドウサイズ $expect_cwnd = 輻輳ウィンドウサイズcwnd$ とし、輻輳、または、パケットロスが発生したサブフローに対応するキューに蓄積されているセグメントについて、式(1)および式(2)の計算を再度行う。つまり、当該キューのセグメントのそれぞれに対応する到達予想時間 D を再計算する。

20

【0091】

さらに、分配制御部211は、輻輳、または、パケットロスが発生したサブフロー以外の全てのサブフローに対応するキューの到達予想時間 D をそれぞれ取得し、それらの中の最大値を特定するとともに、上述のように再計算された D と比較する。そして、再計算された D が最大値を超えたセグメント以降のセグメントを、退避バッファに退避させる。

【0092】

図9を参照して、さらに詳細に説明する。例えば、サブフローSF1で、輻輳、または、パケットロスの発生が検出されたものとする。

30

【0093】

分配制御部211は、サブフローSF1以外のサブフローであるサブフローSF2乃至サブフローSF N のそれぞれに対応するキュー241-2乃至キュー241- N のそれぞれの到達予想時間 D を取得する。この場合、キュー241-2乃至キュー241- N のそれぞれ最後のセグメントについて計算された到達予想時間 D_2 乃至到達予想時間 D_N が取得される。

【0094】

そして、分配制御部211は、到達予想時間 D_2 乃至到達予想時間 D_N の中の最大値を特定する。いまの場合、キュー241- K の到達予想時間である到達予想時間 D_K が最大値として特定されたものとする。

40

【0095】

分配制御部211は、サブフローSF1の縮小後の輻輳ウィンドウサイズ $cwnd$ を取得する。輻輳ウィンドウサイズ $cwnd$ が輻輳ウィンドウサイズ $expect_cwnd$ の初期値となる。分配制御部211は、キュー241-1内に蓄積されているセグメントに対応する到達予想時間 D_1 を再計算する。

【0096】

図9のキュー241-1内には、4つのセグメントが蓄積されている。分配制御部211は、最も先に送出されるセグメント(図中最も下の矩形に対応するセグメント)の到達予想時間を再計算する。すなわち、キュー241-1が空であるものと仮定して、1つのセグメントがキュー241-1に挿入された場合の到達予想時間 D_1 が算出される。

50

【0097】

そして、分配制御部211は、到達予想時間 D_1 と、到達予想時間 D_K を比較する。いまの場合、到達予想時間 D_K の方が到達予想時間 D_1 より大きかった($D_K > D_1$)ものとする。

【0098】

この場合、分配制御部211は、キュー241-1の中で2番目に送られるセグメント(図中下から2番目の矩形に対応するセグメント)の到達予想時間を再計算する。すなわち、キュー241-1にセグメントが1つ蓄積されているものと仮定して、1つのセグメントがキュー241-1に挿入された場合の到達予想時間 D_1 があらためて算出される。

10

【0099】

このように、分配制御部211は、キュー241-1が1度空になったものと仮定し、輻輳ウィンドウサイズの初期値がcwndであるものとし、キュー241-1に蓄積されていたセグメントを、送信順に1つずつ仮想的にキュー241-1に挿入しながら、到達予想時間 D_1 を算出して更新していく。

【0100】

そして、分配制御部211は、到達予想時間 D_1 と、到達予想時間 D_K を比較する。いまの場合、到達予想時間 D_1 の方が到達予想時間 D_K より大きかった($D_1 > D_K$)ものとする。

【0101】

この場合、分配制御部211は、キュー241-1の中で2番目に送られるセグメントより後のセグメント(図中下から3番目と4番目の矩形に対応するセグメント)を退避バッファ232に退避させる。これにより、キュー241-1には、2つのセグメントのみが蓄積された状態となる。

20

【0102】

分配制御部211は、退避バッファ232に蓄積されているセグメントを、キュー241-1乃至キュー241-Nに再度分配する。このとき、退避バッファ232内の最も下のセグメントから順に、 $D+d$ が最小となるキューが選択されて挿入されていく。そして、セグメントが挿入されたキューの到達予想時間が更新されていく。

【0103】

このようにすることで、サブフローSF1乃至サブフローSFNにおいて、輻輳、または、パケットロスが発生した場合であっても、スマートフォン111から送信されたセグメントが、サーバ112にできるだけ順番に到着するようになる。

30

【0104】

従って、本技術によれば、伝送遅延の異なるネットワークを利用したマルチパス通信において、受信側でのリオーダー待ちを回避して、アプリケーションプログラムの処理に要する時間を短縮させることができる。さらに、輻輳、または、パケットロスが発生した場合であっても、受信側でのリオーダー待ちをできるだけ回避することが可能となる。その結果、常に効率的にネットワークの帯域幅を活用することができる。

【0105】

次に、図10のフローチャートを参照して、分配制御部211によるセグメント分配処理の例について説明する。この処理は、例えば、スマートフォン111がマルチパス通信を行っているとき実行される。

40

【0106】

ステップS21において、分配制御部211は、退避バッファ232が空であるか否かを判定する。

【0107】

ステップS21において、退避バッファ232が空であると判定された場合、処理は、ステップS23に進み、分配制御部211は、送信バッファ231からセグメントを取り出す。

50

【 0 1 0 8 】

一方、ステップ S 2 1 において、退避バッファ 2 3 2 が空ではないと判定された場合、処理は、ステップ S 2 2 に進み、分配制御部 2 1 1 は、退避バッファ 2 3 2 からセグメントを取り出す。

【 0 1 0 9 】

ステップ S 2 4 において、分配制御部 2 1 1 は、キュー毎の到達予想時間を取得する。このとき、例えば、キュー 2 4 1 - 1 乃至キュー 2 4 1 - N のそれぞれの到達予想時間 D が取得される。

【 0 1 1 0 】

ステップ S 2 5 において、分配制御部 2 1 1 は、ステップ S 2 4 の処理で取得した到達予想時間 D が最小のキューを選択する。 10

【 0 1 1 1 】

ステップ S 2 6 において、分配制御部 2 1 1 は、ステップ S 2 2 またはステップ S 2 3 の処理で取り出したセグメントを、ステップ S 2 5 の処理で選択したキューに挿入する。

【 0 1 1 2 】

ステップ S 2 7 において、分配制御部 2 1 1 は、ステップ S 2 6 の処理でキューに挿入されたセグメントがキューから出力される時の輻輳ウィンドウサイズ $expect_cwnd$ を予測する。

【 0 1 1 3 】

ステップ S 2 8 において、分配制御部 2 1 1 は、ステップ S 2 7 の処理で予測した輻輳ウィンドウサイズ $expect_cwnd$ を用いて送出予想時間 W を計算する。このとき、上述した式 (1) により送出予想時間 W が算出される。 20

【 0 1 1 4 】

ステップ S 2 9 において、分配制御部 2 1 1 は、当該キューに対応する到達予想時間 D を更新する。すなわち、ステップ S 2 8 の処理で計算された W を用いて式 (2) の演算が行われ、到達予想時間 D が更新される。

【 0 1 1 5 】

ステップ S 3 0 において、分配制御部 2 1 1 は、いずれかのサブフローにおいて輻輳、または、パケットロスが発生したか否かを判定する。

【 0 1 1 6 】

ステップ S 3 0 において、輻輳、または、パケットロスが発生していないと判定された場合、処理は、ステップ S 2 1 に戻り、それ以降の処理が繰り返し実行される。 30

【 0 1 1 7 】

一方、ステップ S 3 0 において、輻輳、または、パケットロスが発生していないと判定された場合、処理は、ステップ S 3 0 に進み、図 1 1 を参照して後述するセグメント退避処理が実行される。

【 0 1 1 8 】

次に、図 1 1 のフローチャートを参照して、図 1 0 のステップ S 3 0 のセグメント退避処理の詳細な例について説明する。

【 0 1 1 9 】

ステップ S 5 1 において、分配制御部 2 1 1 は、輻輳、または、パケットロスが発生したサブフロー以外のサブフローに対応するキューの到達予想時間 D を取得する。 40

【 0 1 2 0 】

例えば、サブフロー S F 1 で輻輳、または、パケットロスが発生した場合、サブフロー S F 1 以外のサブフローであるサブフロー S F 2 乃至サブフロー S F N のそれぞれに対応するキュー 2 4 1 - 2 乃至キュー 2 4 1 - N のそれぞれの到達予想時間 D_2 乃至到達予想時間 D_N が取得される。そして、分配制御部 2 1 1 は、到達予想時間 D_2 乃至到達予想時間 D_N の中の最大値を特定する。いまの場合、キュー 2 4 1 - K の到達予想時間である到達予想時間 D_K が最大値として特定されたものとする。

【 0 1 2 1 】

ステップ S 5 2 において、分配制御部 2 1 1 は、サブフロー S F 1 の縮小後の輻輳ウィンドウサイズ $cwnd$ を取得する。

【 0 1 2 2 】

ステップ S 5 3 において、分配制御部 2 1 1 は、サブフロー S F 1 の最初のセグメントの到達予想時間を再計算する。すなわち、キュー 2 4 1 - 1 が空であるものと仮定して、1 つのセグメントがキュー 2 4 1 - 1 に挿入された場合の到達予想時間 D_1 が算出される。

【 0 1 2 3 】

ステップ S 5 4 において、分配制御部 2 1 1 は、ステップ S 5 3 の処理により得られた到達予想時間 D_1 と、ステップ S 5 1 の処理により得られた到達予想時間 D_K を比較し、到達予想時間 D_1 が、到達予想時間 D_K を超えたか否かを判定する。

10

【 0 1 2 4 】

ステップ S 5 4 において、到達予想時間 D_1 が、到達予想時間 D_K を超えていないと判定された場合、処理は、ステップ S 5 5 に進む。

【 0 1 2 5 】

ステップ S 5 5 において、分配制御部 2 1 1 は、キュー 2 4 1 - 1 に挿入されたセグメントがキュー 2 4 1 - 1 から出力される時の輻輳ウィンドウサイズ $expect_cwnd$ を予測する。

【 0 1 2 6 】

ステップ S 5 6 において、分配制御部 2 1 1 は、ステップ S 5 5 の処理で予測した輻輳ウィンドウサイズ $expect_cwnd$ を用いて送出予想時間 W を計算する。このとき、上述した式 (1) により送出予想時間 W が算出される。

20

【 0 1 2 7 】

ステップ S 5 7 において、分配制御部 2 1 1 は、ステップ S 5 6 の処理で計算された W を用いて式 (2) の演算を行い、次のセグメントが挿入されたときの到達予想時間を再度算出する。すなわち、キュー 2 4 1 - 1 にセグメントが 1 つ蓄積されているものと仮定して、1 つのセグメントがキュー 2 4 1 - 1 に挿入された場合の到達予想時間 D_1 があらためて算出される。

【 0 1 2 8 】

ステップ S 5 4 に戻り、分配制御部 2 1 1 は、ステップ S 5 7 の処理により得られた到達予想時間 D_1 と、ステップ S 5 1 の処理により得られた到達予想時間 D_K を比較し、到達予想時間 D_1 が、到達予想時間 D_K を超えたか否かを判定する。

30

【 0 1 2 9 】

ここで、到達予想時間 D_1 が、到達予想時間 D_K を超えていないと判定された場合、処理は、ステップ S 5 5 に進み、その後、ステップ S 5 4 乃至ステップ S 5 7 の処理が繰り返し実行される。

【 0 1 3 0 】

一方、ステップ S 5 4 において、到達予想時間 D_1 が、到達予想時間 D_K を超えていると判定された場合、ステップ S 5 5 乃至ステップ S 5 7 の処理はスキップされる。

【 0 1 3 1 】

ステップ S 5 8 において、分配制御部 2 1 1 は、以降のセグメントを退避バッファ 2 3 2 に退避させる。

40

【 0 1 3 2 】

このようにして、セグメント退避処理が実行される。

【 0 1 3 3 】

図 1 2 と図 1 3 は、従来のマルチパス通信における受信側のリオーダ待ちデータの長さ、アプリケーションで処理可能なデータの送達レートとの関係を示す図である。

【 0 1 3 4 】

図 1 2 と図 1 3 は、横軸が時間、縦軸がリオーダ待ちデータの長さ (Queue Length)、および、アプリケーションで処理可能なデータの送達レート (Goodput) とされ、線 3 0

50

1により、時間の経過に伴うQueue Lengthの変化が示されており、線302により、時間の経過に伴うGoodputの変化が示されている。

【0135】

図12と図13は、図14に示されるようなネットワークシステムにおける測定結果を表すものである。図14の例では、送信側（Sender）のパーソナルコンピュータ321から受信側（Receiver）のパーソナルコンピュータ322にデータが送信される。

【0136】

パーソナルコンピュータ321は、例えば、イーサネットLANのような有線通信のネットワークインタフェースを2つ有している。すなわち、パーソナルコンピュータ321は、経路325-1および経路325-2の2つの経路でルータ324と接続されている。経路325-1には、遅延発生装置323-1が接続されており、経路325-2は、遅延発生装置323-2が接続されている。

【0137】

なお、経路325-1の帯域幅は5Mbpsとされ、経路325-2の帯域幅も5Mbpsとされている。遅延発生装置323-1および遅延発生装置323-2は、それぞれ経路325-1および経路325-2上で任意の遅延時間を発生させることができる。

【0138】

パーソナルコンピュータ322は、高速LANなどの有線通信のネットワークインタフェースを有しており、ルータ324と接続されている。

【0139】

図12は、経路325-1上の遅延時間を10msとし、経路325-2上の遅延時間も10msとした場合のマルチパス通信における受信側のリオーダー待ちデータの長さ、アプリケーションで処理可能なデータの送達レートの関係を示している。なお、マルチパス通信における各径路へのデータの分配の方式は従来方式を用いている。すなわち、ここでのマルチパス通信には、本技術のように、キューから出力されるまでの時間と伝送遅延を考慮して各経路にデータを分配するような方式が採用されていない。

【0140】

つまり、図12は、従来方式のマルチパス通信であって、遅延差の小さい（無い）複数のネットワークを用いたマルチパス通信における受信側のリオーダー待ちデータの長さ、アプリケーションで処理可能なデータの送達レートの関係を示している。

【0141】

図12に示されるように、約7ms経過後は、Queue Lengthがほぼ一定となり、Goodputは、10Mbpsよりやや低い値でほぼ一定となっている。なお、経路325-1と経路325-2の帯域幅の合計が10Mbpsなので、図12に示されるGoodputは、高い値で安定していると考えられる。

【0142】

図13は、経路325-1上の遅延時間を100msとし、経路325-2上の遅延時間も10msとした場合のマルチパス通信における受信側のリオーダー待ちデータの長さ、アプリケーションで処理可能なデータの送達レートの関係を示している。なお、マルチパス通信における各径路へのデータの分配の方式は従来方式を用いている。すなわち、ここでのマルチパス通信には、本技術のように、キューから出力されるまでの時間と伝送遅延を考慮して各経路にデータを分配するような方式が採用されていない。

【0143】

つまり、図13は、従来方式のマルチパス通信であって、遅延差の大きい複数のネットワークを用いたマルチパス通信における受信側のリオーダー待ちデータの長さ、アプリケーションで処理可能なデータの送達レートの関係を示している。

【0144】

図13に示されるように、Queue Lengthは大きく変化しており、goodputは、通信開始直後が著しく低く、また、その後もQueue Lengthの増加に伴って、goodputの谷が形成されている。

10

20

30

40

50

【 0 1 4 5 】

例えば、図 1 4 のパーソナルコンピュータ 3 2 1 に本技術を適用すれば、経路 3 2 5 - 1 上の遅延時間を 1 0 0 m s とし、経路 3 2 5 - 2 上の遅延時間も 1 0 m s とした場合であっても、図 1 2 とほぼ同様の結果を得ることができる。つまり、本技術によれば、遅延差の大きい複数のネットワークを用いたマルチパス通信において、Queue Length がほぼ一定となり、Goodput も高い値で安定させることが可能となる。

【 0 1 4 6 】

なお、キューから出力されるまでの時間と伝送遅延を考慮して各経路にデータを分配するアルゴリズムも存在する。例えば、S H A K E (SHAring multipath procedure for a cluster network Environment) では、M N がセグメントを送信する際には、パケットが順序通りに到達するように、最も早く H A へ到着すると推測される経路を用いてセグメントを送信する。リンクには仮想的な送信キューを設け、キューから出力されるまでの待ち時間と伝送遅延からパケット到達時刻を予想し各経路へ振り分けるようになされている。

10

【 0 1 4 7 】

しかしながら、S H A K E は、ネットワーク層でのマルチホーミングを行うためのプロトコルとされる。従って、ネットワーク層の制御情報などを用いてパケット到達時刻が予想される。しかしながら、ネットワーク層のプロトコル (例えば、I P) には、輻輳ウィンドウに関する制御情報が規定されておらず、本技術のように、これからキューに挿入されるセグメントがキューから出力される時の輻輳ウィンドウサイズ expect_cwnd を予測することができない。

20

【 0 1 4 8 】

また、T C P のアルゴリズムでは、輻輳が発生しない限り、輻輳ウィンドウサイズが徐々に拡大されるようになされている。従って、送信側からのデータの送出レートが経路 3 2 5 - 1 または経路 3 2 5 - 2 の帯域幅を超えるまで輻輳ウィンドウサイズが拡大され続けることもある。送信側からのデータの送出レートが経路 3 2 5 - 1 または経路 3 2 5 - 2 の帯域幅を超えた場合、その時点で必ず輻輳が発生することになり、輻輳ウィンドウサイズが縮小されることになる。パケットロスが発生した場合も、やはり輻輳ウィンドウサイズが縮小されることになる。

【 0 1 4 9 】

このように、実際の通信では輻輳の発生に伴ってウィンドウサイズが適宜変更されるが、ネットワーク層での制御情報を用いたマルチホーミングの方式では、実際の輻輳ウィンドウサイズの変化を認識することができない。このため、輻輳やパケットロスが発生した場合、送信側で意図した順番で受信側にデータが到着しなくなる。すなわち、ネットワーク層での制御情報を用いたマルチホーミングの方式では、遅延差の大きい複数のネットワークを用いたマルチパス通信において、Goodput を高い値で安定させることができないと考えられる。

30

【 0 1 5 0 】

これに対して、本技術によれば、遅延差の大きい複数のネットワークを用いたマルチパス通信において、輻輳やパケットロスが発生しても、Goodput を高い値で安定させることができる。

40

【 0 1 5 1 】

なお、上述した実施の形態では、輻輳、または、パケットロスが発生したサブフローのみについて、セグメントの再分配を行っているが、輻輳、または、パケットロスが発生していない 1 以上のサブフローについて、未処理のセグメントの再分配を行うようにしてもよい。

【 0 1 5 2 】

以上においては、本技術をスマートフォン、パーソナルコンピュータなどに適用する例について説明したが、その他の電子機器にも本技術を適用することが可能である。

【 0 1 5 3 】

なお、上述した一連の処理は、ハードウェアにより実行させることもできるし、ソフト

50

ウェアにより実行させることもできる。上述した一連の処理をソフトウェアにより実行させる場合には、そのソフトウェアを構成するプログラムが、専用のハードウェアに組み込まれているコンピュータ、または、各種のプログラムをインストールすることで、各種の機能を実行することが可能な、例えば図 15 に示されるような汎用のパーソナルコンピュータ 700 などに、ネットワークや記録媒体からインストールされる。

【0154】

図 15 において、CPU (Central Processing Unit) 701 は、ROM (Read Only Memory) 702 に記憶されているプログラム、または記憶部 708 から RAM (Random Access Memory) 703 にロードされたプログラムに従って各種の処理を実行する。RAM 703 にはまた、CPU 701 が各種の処理を実行する上において必要なデータなども適宜記憶される。

10

【0155】

CPU 701、ROM 702、および RAM 703 は、バス 704 を介して相互に接続されている。このバス 704 にはまた、入出力インタフェース 705 も接続されている。

【0156】

入出力インタフェース 705 には、キーボード、マウスなどよりなる入力部 706、LCD (Liquid Crystal display) などよりなるディスプレイ、並びにスピーカなどよりなる出力部 707、ハードディスクなどより構成される記憶部 708、モデム、LAN カードなどのネットワークインタフェースカードなどより構成される通信部 709 が接続されている。通信部 709 は、インターネットを含むネットワークを介しての通信処理を行う。

20

【0157】

入出力インタフェース 705 にはまた、必要に応じてドライブ 710 が接続され、磁気ディスク、光ディスク、光磁気ディスク、或いは半導体メモリなどのリムーバブルメディア 711 が適宜装着され、それらから読み出されたコンピュータプログラムが、必要に応じて記憶部 708 にインストールされる。

【0158】

上述した一連の処理をソフトウェアにより実行させる場合には、そのソフトウェアを構成するプログラムが、インターネットなどのネットワークや、リムーバブルメディア 711 などからなる記録媒体からインストールされる。

【0159】

なお、この記録媒体は、図 15 に示される、装置本体とは別に、ユーザにプログラムを配信するために配布される、プログラムが記録されている磁気ディスク（フロッピディスク（登録商標）を含む）、光ディスク（CD-ROM (Compact Disk-Read Only Memory)、DVD (Digital Versatile Disk) を含む）、光磁気ディスク（MD (Mini-Disk)（登録商標）を含む）、もしくは半導体メモリなどよりなるリムーバブルメディア 711 により構成されるものだけでなく、装置本体に予め組み込まれた状態でユーザに配信される、プログラムが記録されている ROM 702 や、記憶部 708 に含まれるハードディスクなどで構成されるものも含む。

30

【0160】

なお、本明細書において上述した一連の処理は、記載された順序に沿って時系列的に行われる処理はもちろん、必ずしも時系列的に処理されなくとも、並列的あるいは個別に実行される処理をも含むものである。

40

【0161】

また、本技術の実施の形態は、上述した実施の形態に限定されるものではなく、本技術の要旨を逸脱しない範囲において種々の変更が可能である。

【符号の説明】

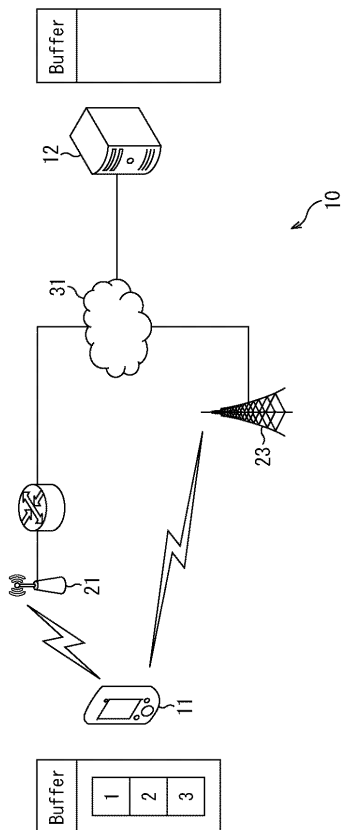
【0162】

100 ネットワークシステム, 111 スマートフォン, 112 サーバ, 121 アクセスポイント, 122 ルータ, 123 基地局, 131 インターネット, 151 CPU, 171 広域無線通信部, 172 有線通信部, 173

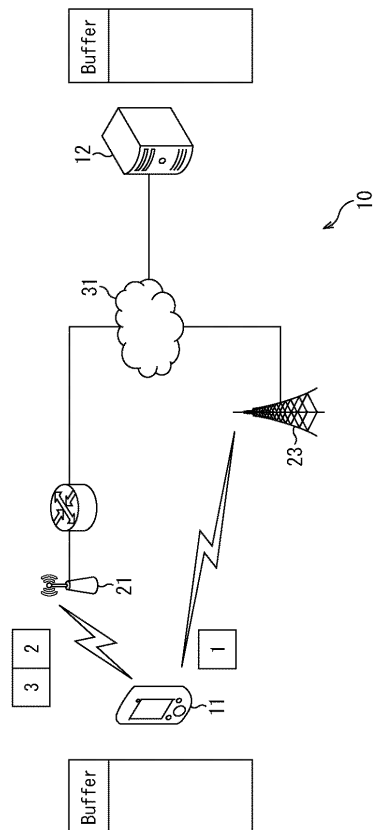
50

近距離無線通信部, 201 OS, 202 TCP/IP通信制御部, 203 MPTCP制御部, 204 アプリケーションプログラム, 211 分配制御部, 231 送信バッファ, 232 退避バッファ, 241-1乃至241-N キュー

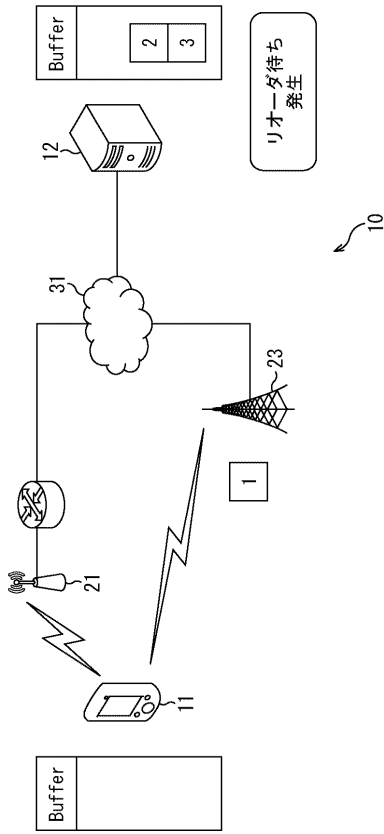
【図1】



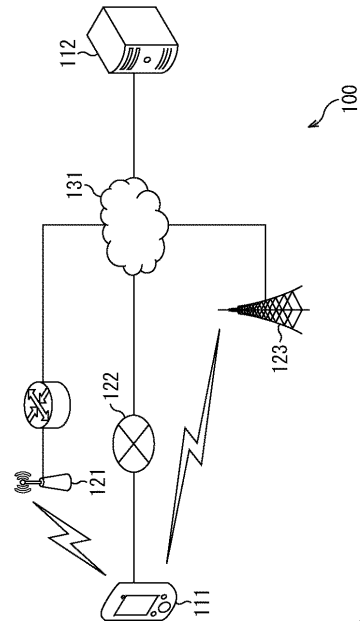
【図2】



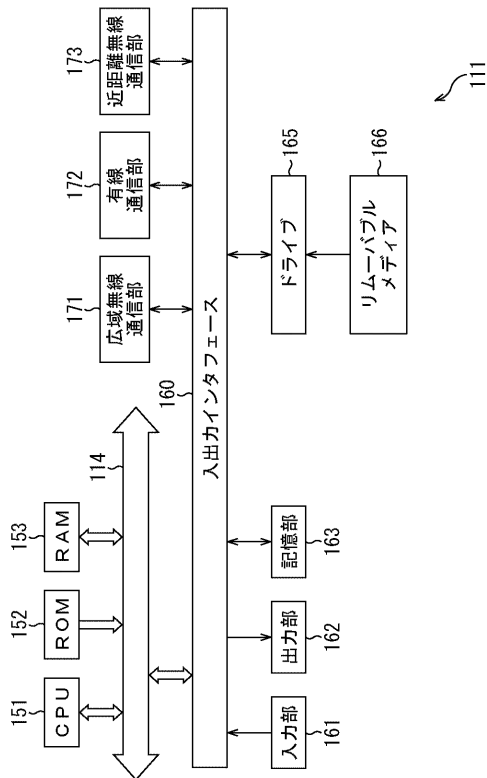
【 図 3 】
図3



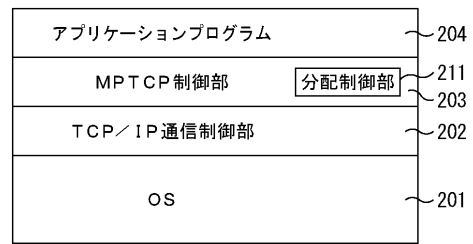
【 図 4 】
図4



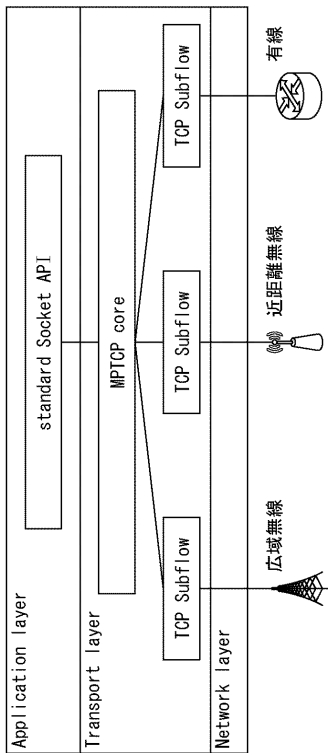
【 図 5 】
図5



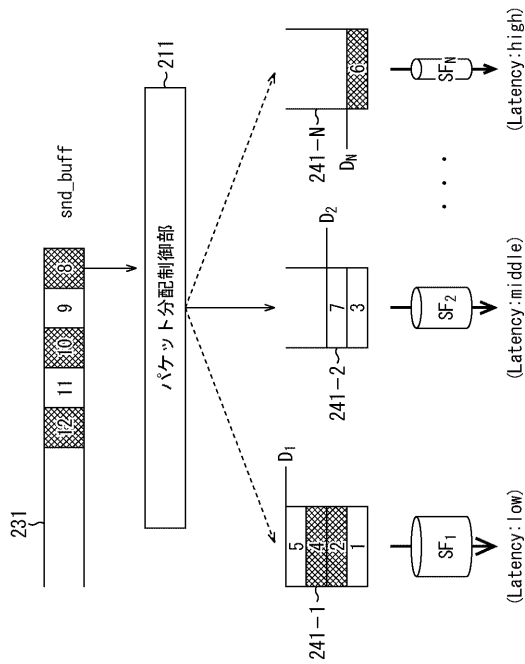
【 図 6 】
図6



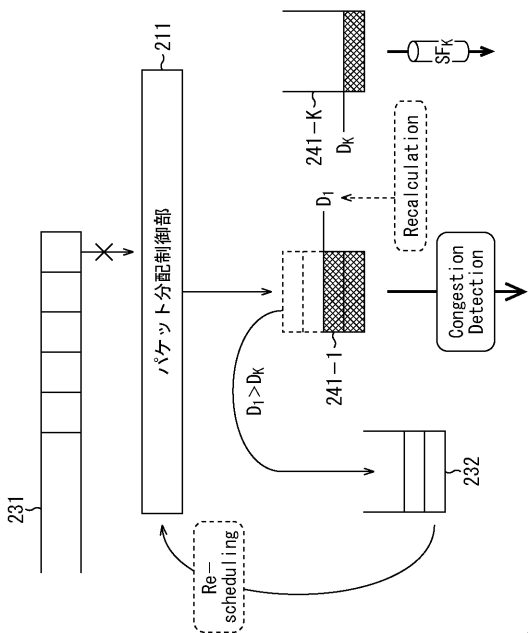
【 図 7 】



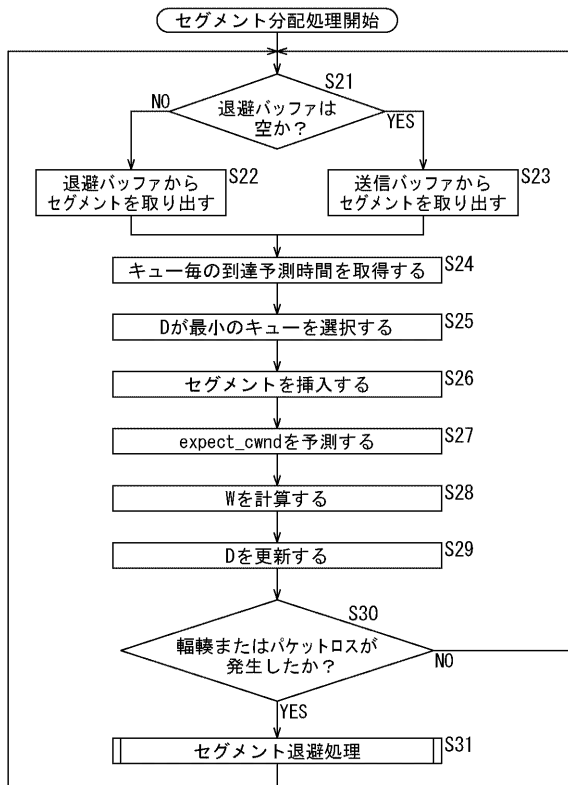
【 図 8 】



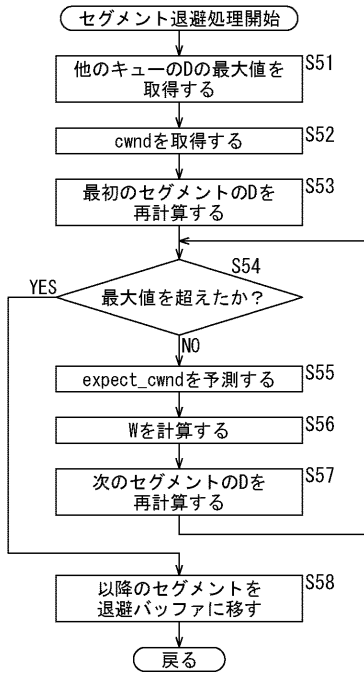
【 図 9 】



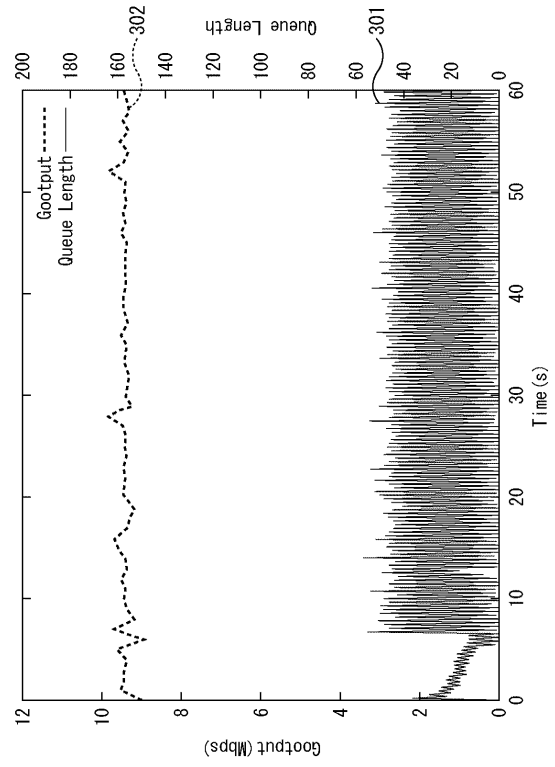
【 図 10 】



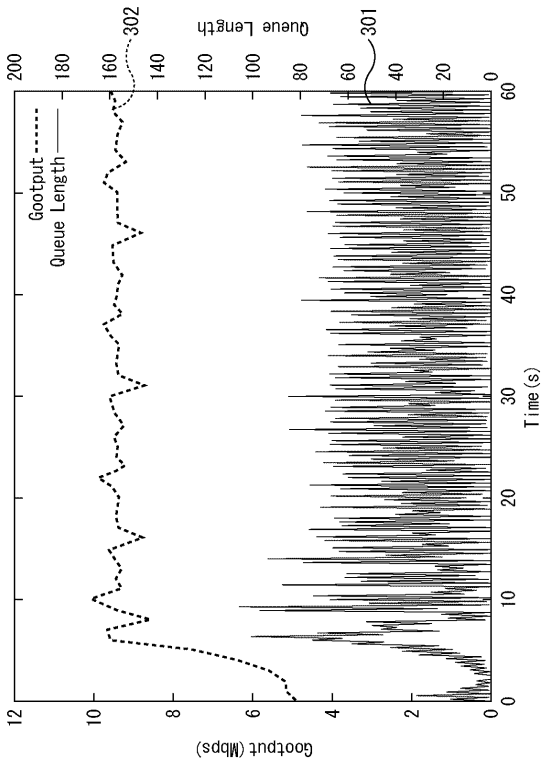
【図11】
図11



【図12】
図12



【図13】
図13



【図14】
図14

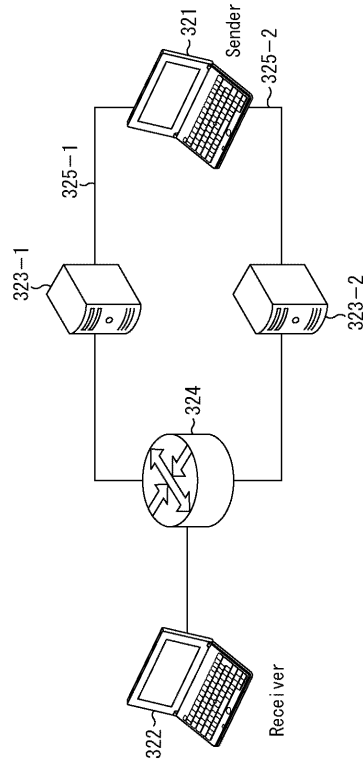


図15 【 図 1 5 】

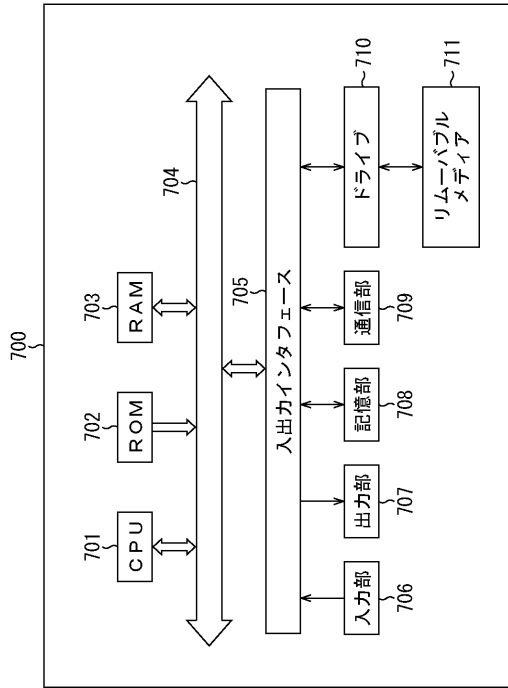


図15

フロントページの続き

Fターム(参考) 5K030 GA03 GA11 KA03 LB06 LE03
5K034 AA07 DD01 EE11 FF02 FF13 GG03 HH01 HH02 HH06 HH42
HH64 KK21 MM16 MM18 QQ05
5K067 AA13 BB04 BB21 CC08 DD11 DD57 EE02 EE10 EE16 FF05
HH22 HH23