

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第5444647号
(P5444647)

(45) 発行日 平成26年3月19日(2014.3.19)

(24) 登録日 平成26年1月10日(2014.1.10)

(51) Int.Cl.		F I			
HO4L 1/00	(2006.01)	HO4L 1/00			B
HO4L 29/08	(2006.01)	HO4L 13/00		3 O 7 Z	
HO4L 12/70	(2013.01)	HO4L 12/70			E
		HO4L 1/00			E

請求項の数 5 (全 23 頁)

(21) 出願番号	特願2008-172626 (P2008-172626)	(73) 特許権者	000005223 富士通株式会社 神奈川県川崎市中原区上小田中4丁目1番1号
(22) 出願日	平成20年7月1日(2008.7.1)	(74) 代理人	100070150 弁理士 伊東 忠彦
(65) 公開番号	特開2010-16484 (P2010-16484A)	(72) 発明者	亀山 裕亮 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
(43) 公開日	平成22年1月21日(2010.1.21)	(72) 発明者	松園 明久 神奈川県川崎市中原区上小田中4丁目1番1号 富士通アドバンステクノロジー株式会社内
審査請求日	平成23年4月18日(2011.4.18)	審査官	谷岡 佳彦

最終頁に続く

(54) 【発明の名称】 データ転送装置、データ転送方法及びデータ転送プログラム

(57) 【特許請求の範囲】

【請求項1】

データを転送するデータ転送装置であって、

転送するデータの符号化に必要な複数種類のデータサイズのブロックが予め用意されており、前記複数種類のデータサイズのブロック毎に符号化行列を格納する符号化行列格納手段と、

前記データを蓄積するデータ蓄積手段に蓄積されたデータのデータサイズに基づき、前記ブロックをデータサイズの大きい方から順に決定する符号化サイズ決定手段と、

決定した前記ブロックに応じた前記符号化行列を取得し、取得した前記符号化行列を用いて前記データ蓄積手段に蓄積されたデータを、決定した前記ブロックのデータサイズ毎に符号化する符号化手段と、

符号化された符号化データに、符号化に用いた前記符号化行列に関する符号化情報を付加したパケットを生成するパケット生成手段と、

生成した前記パケットを送信する送信手段とを有し、

前記符号化サイズ決定手段は、データ蓄積手段に蓄積されたデータよりもデータサイズが小さくなる前記ブロックと、該ブロックのデータサイズを所定の割合だけ小さくした縮小データサイズと前記データ蓄積手段に蓄積されたデータのデータサイズとを比較して前記データ蓄積手段に蓄積されたデータのデータサイズよりも縮小データサイズが小さくなる前記ブロックとのうち、データサイズが最も大きい前記ブロックを決定し、

10

20

前記縮小データサイズの前記ブロックを決定した場合、前記符号化手段は、決定した前記ブロックの前記符号化行列を取得し、決定した前記ブロックの縮小データサイズ分、前記データ蓄積手段から前記データを取得し、前記データ蓄積手段から取得した前記データにダミーデータを追加し、取得した前記符号化行列を用いて前記データ蓄積手段から取得してダミーデータを追加した前記データを符号化するデータ転送装置。

【請求項 2】

前記符号化手段は、決定した前記ブロックの前記符号化行列を取得し、決定した前記ブロックのデータサイズ分、前記データ蓄積手段から前記データを取得し、取得した前記符号化行列を用いて前記データ蓄積手段から取得した前記データを符号化する請求項 1 記載のデータ転送装置。

10

【請求項 3】

前記パケットが連続して送信されない無送信時間を計時するタイマー手段と、
前記データ蓄積手段に蓄積されたデータのデータサイズが前記ブロックのデータサイズのうち最も小さいデータサイズより大きくなったとき又は前記パケットが連続して送信されない無送信時間が所定時間を経過したとき、前記符号化サイズ決定手段に前記ブロックのデータサイズを決定させる符号化サイズ決定指示手段と
を有する請求項 1 又は 2 記載のデータ転送装置。

【請求項 4】

データを転送するデータ転送装置のデータ転送方法であって、
転送するデータの符号化に必要な複数種類のデータサイズのブロックが予め用意されており、符号化行列格納手段が、前記複数種類のデータサイズのブロック毎に符号化行列を格納される符号化行列格納ステップと、
符号化サイズ決定手段が、前記データを蓄積するデータ蓄積手段に蓄積されたデータのデータサイズに基づき前記ブロックをデータサイズの大きい方から順に決定する符号化サイズ決定ステップと、
符号化手段が、決定した前記ブロックに応じた前記符号化行列を取得し、取得した前記符号化行列を用いて前記データ蓄積手段に蓄積されたデータを、決定した前記ブロックのデータサイズ毎に符号化する符号化ステップと、
パケット生成手段が、符号化された符号化データに、符号化に用いた前記符号化行列に関する符号化情報を付加したパケットを生成するパケット生成ステップと、
送信手段が、生成した前記パケットを送信する送信ステップと
を有し、

20

30

前記符号化サイズ決定ステップは、データ蓄積手段に蓄積されたデータよりもデータサイズが小さくなる前記ブロックと、該ブロックのデータサイズを所定の割合だけ小さくした縮小データサイズと前記データ蓄積手段に蓄積されたデータのデータサイズとを比較して前記データ蓄積手段に蓄積されたデータのデータサイズよりも縮小データサイズが小さくなる前記ブロックとのうち、データサイズが最も大きい前記ブロックを決定し、
前記縮小データサイズの前記ブロックを決定した場合、前記符号化ステップは、決定した前記ブロックの前記符号化行列を取得し、決定した前記ブロックの縮小データサイズ分、前記データ蓄積手段から前記データを取得し、前記データ蓄積手段から取得した前記データにダミーデータを追加し、取得した前記符号化行列を用いて前記データ蓄積手段から取得してダミーデータを追加した前記データを符号化するデータ転送方法。

40

【請求項 5】

データを転送するデータ転送装置として機能するコンピュータを、
転送するデータの符号化に必要な複数種類のデータサイズのブロックが予め用意されており、前記複数種類のデータサイズのブロック毎に符号化行列を格納する符号化行列格納手段と、
前記データを蓄積するデータ蓄積手段に蓄積されたデータのデータサイズに基づき、前記ブロックをデータサイズの大きい方から順に決定する符号化サイズ決定手段と、
決定した前記ブロックに応じた前記符号化行列を取得し、取得した前記符号化行列を用

50

いて前記データ蓄積手段に蓄積されたデータを、決定した前記ブロックのデータサイズ毎に符号化する符号化手段と、

符号化された符号化データに、符号化に用いた前記符号化行列に関する符号化情報を付加したパケットを生成するパケット生成手段と、

生成した前記パケットを送信する送信手段として機能させ、

前記符号化サイズ決定手段は、データ蓄積手段に蓄積されたデータよりもデータサイズが小さくなる前記ブロックと、該ブロックのデータサイズを所定の割合だけ小さくした縮小データサイズと前記データ蓄積手段に蓄積されたデータのデータサイズとを比較して前記データ蓄積手段に蓄積されたデータのデータサイズよりも縮小データサイズが小さくなる前記ブロックとのうち、データサイズが最も大きい前記ブロックを決定し、

10

前記縮小データサイズの前記ブロックを決定した場合、前記符号化手段は、決定した前記ブロックの前記符号化行列を取得し、決定した前記ブロックの縮小データサイズ分、前記データ蓄積手段から前記データを取得し、前記データ蓄積手段から取得した前記データにダミーデータを追加し、取得した前記符号化行列を用いて前記データ蓄積手段から取得してダミーデータを追加した前記データを符号化するデータ転送プログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、データ転送装置、データ転送方法及びデータ転送プログラムに係り、特に符号化を行うデータ転送装置、データ転送方法及びデータ転送プログラムに関する。

20

【背景技術】

【0002】

近年、IPネットワーク等のネットワーク経由で画像や映像などのデータを転送することがよく行われるようになってきている。例えば遠隔地のクライアントからネットワーク経由でサーバの画面操作を行うASP(Active Server Pages)システムは、ネットワーク経由で画面データを転送している。

【0003】

ネットワーク経由でデータを転送する場合は、データの圧縮、データの符号化、データの復号、再送等の技術が一般的に利用されている(例えば特許文献1~4参照)。

30

【特許文献1】特開平10-285205号公報

【特許文献2】特開2007-274309号公報

【特許文献3】特開平10-65675号公報

【特許文献4】特開平11-17737号公報

【発明の開示】

【発明が解決しようとする課題】

【0004】

例えばIPネットワークにおけるデータ転送では、欠損部分の再送などの機能を持ち信頼性の高いTCPが用いられることが多い。しかし、TCPを用いるデータ転送は応答処理などのオーバーヘッドがあるため転送速度が比較的遅い。特にネットワーク間の距離を表わすRTT(Round Trip Time)の値が大きくなるほど、TCPを用いるデータ転送は低速になりリアルタイム性も低くなる。

40

【0005】

これに対して、信頼性が低くても問題のない音声のストリーミングなどのデータ転送の場合には、RTTの影響を受けにくいUDPを用いることによりリアルタイム性を高めている。

【0006】

ASPシステムはリアルタイム性が高いことが必要であるが、信頼性が低いと通信の結果(画面)が変わってしまうためTCPが用いられている。その為、ASPシステムでは前述のようにRTTの影響を受け、リアルタイム性が低くなる。

50

【 0 0 0 7 】

リアルタイム性が低くなるという問題を解決する方法としては、データ転送をUDPで行い、データ転送前、データに冗長度をつけることで欠損が発生しても、その欠損部分を訂正して信頼性を高める手法がある。なお、データを符号化する為には一定のデータサイズが必要となる。

【 0 0 0 8 】

したがって、データ転送をUDPで行い、データ転送前、データに冗長度をつけることで欠損が発生しても、その欠損部分を訂正して信頼性を高める手法は、一定量のデータが常に流れてくるシステムにおいて問題ない。

【 0 0 0 9 】

しかしながら、データ転送をUDPで行い、データ転送前、データに冗長度をつけることで欠損が発生しても、その欠損部分を訂正して信頼性を高める手法は、ASPシステムのように時間によって流れてくるデータの量が変化する場合、データを符号化する為に必要な一定のデータサイズとなるまで待ち時間が発生するという問題があった。

【 0 0 1 0 】

本発明の一実施形態は、上記の点に鑑みなされたもので、リアルタイム性を損なわずにデータロス対策と高速化の為のデータ符号化とを行うデータ転送装置、データ転送方法及びデータ転送プログラムを提供することを目的とする。

【課題を解決するための手段】

【 0 0 1 1 】

上記課題を解決するため、本発明の一実施形態は、データを転送するデータ転送装置であって、転送するデータの符号化に必要な複数種類のデータサイズのブロックが予め用意されており、前記複数種類のデータサイズのブロック毎に符号化行列を格納する符号化行列格納手段と、前記データを蓄積するデータ蓄積手段に蓄積されたデータのデータサイズに基づき、前記ブロックをデータサイズの大きい方から順に決定する符号化サイズ決定手段と、決定した前記ブロックに応じた前記符号化行列を取得し、取得した前記符号化行列を用いて前記データ蓄積手段に蓄積されたデータを、決定した前記ブロックのデータサイズ毎に符号化する符号化手段と、符号化された符号化データに、符号化に用いた前記符号化行列に関する符号化情報を付加したパケットを生成するパケット生成手段と、生成した前記パケットを送信する送信手段とを有し、前記符号化サイズ決定手段は、データ蓄積手段に蓄積されたデータよりもデータサイズが小さくなる前記ブロックと、該ブロックのデータサイズを所定の割合だけ小さくした縮小データサイズと前記データ蓄積手段に蓄積されたデータのデータサイズとを比較して前記データ蓄積手段に蓄積されたデータのデータサイズよりも縮小データサイズが小さくなる前記ブロックとのうち、データサイズが最も大きい前記ブロックを決定し、前記縮小データサイズの前記ブロックを決定した場合、前記符号化手段は、決定した前記ブロックの前記符号化行列を取得し、決定した前記ブロックの縮小データサイズ分、前記データ蓄積手段から前記データを取得し、前記データ蓄積手段から取得した前記データにダミーデータを追加し、取得した前記符号化行列を用いて前記データ蓄積手段から取得してダミーデータを追加した前記データを符号化する。

【 0 0 1 2 】

なお、本発明の一実施形態の構成要素、表現または構成要素の任意の組合せを、方法、装置、システム、コンピュータプログラム、記録媒体、データ構造などに適用したのもも本発明の態様として有効である。

【発明の効果】

【 0 0 1 3 】

上述の如く、本発明の一実施形態によれば、リアルタイム性を損なわずにデータロス対策と高速化の為のデータ符号化とを行うデータ転送装置、データ転送方法及びデータ転送プログラムを提供できる。

【発明を実施するための最良の形態】

【 0 0 1 4 】

10

20

30

40

50

次に、本発明を実施するための最良の形態を、以下の実施例に基づき図面を参照しつつ説明していく。なお、本実施例ではデータを転送するデータ転送システムとしてASPシステムを例に説明する。

【0015】

図1はASPシステムの一例の構成図である。ASPシステム1は、ASPサーバ10と、クライアント20と、データ転送装置30と、データ受信装置40と、ネットワーク50とを有する構成である。ASPシステム1は、遠隔地のクライアント20からネットワーク50経由でASPサーバ10の画面操作を行うものである。

【0016】

ASPサーバ10は、データ転送装置30と、データ受信装置40と、ネットワーク50とを經由して画面データをクライアント20に転送している。ASPサーバ10がクライアント20に転送している画面データは、単位時間ごとに見た場合、転送するデータサイズが一定でなく、データサイズの変動が大きい。なお、後述するように、データ転送装置30は画面データを転送する際、リアルタイム性を損なわずに、データロス対策と高速化の為のデータ符号化とを行う。

10

【0017】

データ転送装置30は、メモリ31と、タイマー32と、符号化装置33と、送信装置34と、受信装置35と、符号化行列データ36とを有する構成である。データ受信装置40は、受信装置41と、送信装置42と、復号装置43とを有する構成である。ASPサーバ10はクライアント20に転送する画面データ(以下、単にデータという)をメモリ31に順次、蓄積していく。

20

【0018】

データ転送装置30はメモリ31に蓄積されたデータを、後述のようにデータ受信装置40へ転送する。データ受信装置40はデータ転送装置30から転送されたデータをクライアント20へ後述のように転送する。なお、データ転送装置30及びデータ受信装置40の詳細は後述する。

【0019】

図2はデータ転送装置の一例のハードウェア構成図である。図2のデータ転送装置30はコンピュータ及びデータ転送プログラムによりデータ転送装置30を実現する例を表している。

30

【0020】

データ転送装置30は、それぞれバスBで相互に接続されている入力装置61, 出力装置62, ドライブ装置63, 補助記憶装置64, 主記憶装置65, 演算処理装置66及びインターフェース装置67を有する構成である。なお、データ転送装置30は入力装置61及び出力装置62の一方又は両方を有さない構成も可能である。

【0021】

入力装置61はキーボードやマウスなどで構成され、各種信号を入力するために用いられる。出力装置62はディスプレイ装置などで構成され、各種ウィンドウやデータ等を表示するために用いられる。インターフェース装置67は、モデム, ルータ, LANカードなどで構成されており、ネットワーク50に接続する為に用いられる。

40

【0022】

データ転送プログラムは、データ転送装置30を制御する各種プログラムの少なくとも一部である。データ転送プログラムは例えば記録媒体68の配布やネットワーク50からのダウンロードなどによって提供される。データ転送プログラムを記録した記録媒体68は、CD-ROM、フレキシブルディスク、光磁気ディスク等の様に情報を光学的、電氣的或いは磁氣的に記録する記録媒体、ROM、フラッシュメモリ等の様に情報を電氣的に記録する半導体メモリ等、様々なタイプの記録媒体を用いることができる。

【0023】

また、データ転送プログラムを記録した記録媒体68がドライブ装置63にセットされると、データ転送プログラムは記録媒体68からドライブ装置63を介して補助記憶装置

50

64にインストールされる。ネットワーク50からダウンロードされたデータ転送プログラムは、インターフェース装置67を介して補助記憶装置64にインストールされる。

【0024】

補助記憶装置64はインストールされたデータ転送プログラムを格納すると共に、必要なファイル、データ等を格納する。主記憶装置65は、コンピュータの起動時などに補助記憶装置64からデータ転送プログラムを読み出して格納する。そして、演算処理装置66は主記憶装置65に格納されたデータ転送プログラムに従って、後述するような各種処理を実現している。

【0025】

図3はデータ転送装置のメイン処理手順を表した一例のフローチャートである。例えばデータ転送装置30が起動されると、図3のフローチャートに示すメイン処理手順が開始される。

【0026】

ステップS1に進み、符号化装置33は「k」の値を0に設定する。ステップS2に進み、符号化装置33は送信パケットのパケットサイズ(例えば1000Byte)を決定する。なお、パケットサイズは、送信パケットのデータサイズではなく、送信パケットの送信データ部分のデータサイズ(符号化の単位:サブデータのデータサイズ)を表す。

【0027】

ステップS3に進み、符号化装置33は、利用する「n」の値(例えばn=8,128及び1024)を複数個、決定しておく。符号化装置33は、符号化に必要なデータサイズ(ブロックのデータサイズ)を「n*パケットサイズ」として決定する。例えばnの値が8,128,1024のとき、符号化に必要なブロックのデータサイズは、それぞれ8KB,128KB,1024KBとなる。以下では、符号化に必要なブロックのデータサイズが、8KB,128KB,1024KBである例を説明する。

【0028】

ステップS4に進み、符号化装置33は、後述するタイマー処理をタイマー32に開始させる。ステップS5に進み、メモリ31はクライアント20に転送するデータがASPサーバ10によって順次、蓄積される。ステップS6に進み、符号化装置33はメモリ31に蓄積されたデータのデータサイズが、ステップS3で決定したブロックのデータサイズのうち最も小さいデータサイズ(8KB)よりも大きいかを判定する。

【0029】

メモリ31に蓄積されたデータのデータサイズが、ステップS3で決定したブロックのデータサイズのうち最も小さいデータサイズよりも大きいと判定すると、符号化装置33はステップS7に進み、後述する符号化サイズ決定処理を行ったあと、ステップS5の処理に戻る。

【0030】

メモリ31に蓄積されたデータのデータサイズが、ステップS3で決定したブロックのデータサイズのうち最も小さいデータサイズよりも大きくないと判定すると、符号化装置33はステップS5の処理に戻る。

【0031】

図4はタイマー処理手順を表した一例のフローチャートである。タイマー32は前回の送信パケットが送信されてから又はデータ転送装置30の電源が投入されてから経過した時間(無送信時間)を計時している。ステップS11に進み、タイマー32は前回の送信パケットが送信されてから又はデータ転送装置30の電源が投入されてから規定の時間が経過したかを判定する。

【0032】

前回の送信パケットが送信されてから又はデータ転送装置30の電源が投入されてから規定の時間が経過していなければ、タイマー32はステップS11の処理に戻る。前回の送信パケットが送信されてから又はデータ転送装置30の電源が投入されてから規定の時間が経過していれば、タイマー32はステップS12に進み、後述する符号化サイズ決定

10

20

30

40

50

処理を符号化装置 3 3 に指示したあと、ステップ S 1 1 の処理に戻る。

【 0 0 3 3 】

図 5 ~ 図 8 は符号化サイズ決定処理手順を表した一例のフローチャートである。符号化装置 3 3 はステップ S 2 1 に進み、メモリ 3 1 に蓄積されたデータのデータ量 (データサイズ) が、ステップ S 3 で決定したブロックのデータサイズのうち最も大きいデータサイズ (1 0 2 4 K B) よりも大きいかを判定する。

【 0 0 3 4 】

メモリ 3 1 に蓄積されたデータのデータ量が、1 0 2 4 K B よりも大きければ、符号化装置 3 3 はステップ S 2 2 に進み、「 n 」の値を 1 0 2 4 に決定し、メモリ 3 1 に蓄積されたデータの先頭から 1 0 2 4 K B を「 k 」番目のブロックとする。

10

【 0 0 3 5 】

ステップ S 2 3 に進み、符号化装置 3 3 は「 k 」番目のブロックの符号化と送信処理を後述のように行う。ステップ S 2 4 に進み、符号化装置 3 3 はメモリ 3 1 に蓄積されているデータの先頭から 1 0 2 4 K B を削除する。即ち、「 k 」番目のブロックとされたデータをメモリ 3 1 から削除する。ステップ S 2 5 に進み、符号化装置 3 3 は「 k 」の値をインクリメントしたあと、ステップ S 2 1 の処理に戻る。

【 0 0 3 6 】

一方、符号化装置 3 3 はメモリ 3 1 に蓄積されたデータのデータ量が、1 0 2 4 K B よりも大きくなければステップ S 2 6 に進み、メモリ 3 1 に蓄積されているデータのデータ量が、9 1 2 K B (1 0 2 4 K B の 9 割のデータサイズ) よりも大きいかを判定する。

20

【 0 0 3 7 】

メモリ 3 1 に蓄積されているデータのデータ量が、9 1 2 K B よりも大きければ、符号化装置 3 3 はステップ S 2 7 に進み、「 n 」の値を 1 0 2 4 に決定し、メモリ 3 1 に蓄積されているデータの先頭から 9 1 2 K B を「 k 」番目のブロックとする。

【 0 0 3 8 】

ステップ S 2 8 に進み、符号化装置 3 3 は「 k 」番目のブロックの符号化と送信処理を後述のように行う。ステップ S 2 9 に進み、符号化装置 3 3 はメモリ 3 1 に蓄積されているデータの先頭から 9 1 2 K B を削除する。即ち、「 k 」番目のブロックとされたデータをメモリ 3 1 から削除する。ステップ S 3 0 に進み、符号化装置 3 3 は「 k 」の値をインクリメントしたあと、ステップ S 2 6 の処理に戻る。

30

【 0 0 3 9 】

一方、符号化装置 3 3 はメモリ 3 1 に蓄積されたデータのデータ量が、9 1 2 K B よりも大きくなければステップ S 3 1 に進み、メモリ 3 1 に蓄積されているデータのデータ量が、ステップ S 3 で決定したブロックのデータサイズのうち、2 番目に大きいデータサイズ (1 2 8 K B) よりも大きいかを判定する。

【 0 0 4 0 】

メモリ 3 1 に蓄積されているデータのデータ量が、1 2 8 K B よりも大きければ、符号化装置 3 3 はステップ S 3 2 に進み、「 n 」の値を 1 2 8 に決定し、メモリ 3 1 に蓄積されているデータの先頭から 1 2 8 K B を「 k 」番目のブロックとする。

【 0 0 4 1 】

ステップ S 3 3 に進み、符号化装置 3 3 は「 k 」番目のブロックの符号化と送信処理を後述のように行う。ステップ S 3 4 に進み、符号化装置 3 3 はメモリ 3 1 に蓄積されているデータの先頭から 1 2 8 K B を削除する。即ち、「 k 」番目のブロックとされたデータをメモリ 3 1 から削除する。ステップ S 3 5 に進み、符号化装置 3 3 は「 k 」の値をインクリメントしたあと、ステップ S 3 1 の処理に戻る。

40

【 0 0 4 2 】

一方、符号化装置 3 3 はメモリ 3 1 に蓄積されたデータのデータ量が、1 2 8 K B よりも大きくなければステップ S 3 6 に進み、メモリ 3 1 に蓄積されているデータのデータ量が、1 1 5 K B (1 2 8 K B の 9 割のデータサイズ) よりも大きいかを判定する。

【 0 0 4 3 】

50

メモリ31に蓄積されているデータのデータ量が、115KBよりも大きければ、符号化装置33はステップS37に進み、「n」の値を128に決定し、メモリ31に蓄積されているデータの先頭から115KBを「k」番目のブロックとする。

【0044】

ステップS38に進み、符号化装置33は「k」番目のブロックの符号化と送信処理を後述のように行う。ステップS39に進み、符号化装置33はメモリ31に蓄積されているデータの先頭から115KBを削除する。即ち、「k」番目のブロックとされたデータをメモリ31から削除する。ステップS40に進み、符号化装置33は「k」の値をインクリメントしたあと、ステップS36の処理に戻る。

【0045】

一方、符号化装置33はメモリ31に蓄積されたデータのデータ量が、115KBよりも大きくなければステップS41に進み、メモリ31に蓄積されているデータのデータ量が、ステップS3で決定したブロックのデータサイズのうち、3番目に大きいデータサイズ(8KB)よりも大きいかを判定する。メモリ31に蓄積されているデータのデータ量が、8KBよりも大きければ、符号化装置33はステップS42に進み、「n」の値を8に決定し、メモリ31に蓄積されているデータの先頭から8KBを「k」番目のブロックとする。

【0046】

ステップS43に進み、符号化装置33は「k」番目のブロックの符号化と送信処理を後述のように行う。ステップS44に進み、符号化装置33はメモリ31に蓄積されているデータの先頭から8KBを削除する。即ち、「k」番目のブロックとされたデータをメモリ31から削除する。ステップS45に進み、符号化装置33は「k」の値をインクリメントしたあと、ステップS41の処理に戻る。

【0047】

一方、符号化装置33はメモリ31に蓄積されたデータのデータ量が、8KBよりも大きくなければステップS46に進み、メモリ31に蓄積されているデータのデータ量が7KB(8KBの9割のデータサイズ)よりも大きいかを判定する。メモリ31に蓄積されているデータのデータ量が、7KBよりも大きければ、符号化装置33はステップS47に進み、「n」の値を8に決定し、メモリ31に蓄積されているデータの先頭から7KBを「k」番目のブロックとする。

【0048】

ステップS48に進み、符号化装置33は「k」番目のブロックの符号化と送信処理を後述のように行う。ステップS49に進み、符号化装置33はメモリ31に蓄積されているデータの先頭から7KBを削除する。即ち、「k」番目のブロックとされたデータをメモリ31から削除する。ステップS50に進み、符号化装置33は「k」の値をインクリメントしたあと、ステップS46の処理に戻る。

【0049】

一方、符号化装置33はメモリ31に蓄積されたデータのデータ量が7KBよりも大きくなければステップS51に進み、メモリ31にデータが残っているかを判定する。メモリ31にデータが残っていれば、符号化装置33はステップS52に進み、メモリ31に残っているデータを「k」番目のブロックとする。

【0050】

ステップS53に進み、符号化装置33は「k」番目のブロックを符号化せず、送信処理を後述のように行う。ステップS54に進み、符号化装置33は「k」の値をインクリメントする。そして、ステップS55に進み、符号化装置33はメモリ31に残っているデータを削除する。即ち、「k」番目のブロックとされたメモリ31に残っていたデータをメモリ31から削除したあと、符号化サイズ決定処理を終了する。なお、ステップS51においてメモリ31にデータが残っていなければ、符号化装置33は、そのまま符号化サイズ決定処理を終了する。

【0051】

10

20

30

40

50

図9はブロックの符号化と送信処理の手順を表した一例のフローチャートである。図9では「 n 」の値を符号化レベルと呼んでいる。ステップS61に進み、符号化装置33はブロックのデータサイズが符号化レベル n (1024、128、8)に必要なデータサイズ(1024KB、128KB、8KB)よりも小さい場合、末尾にダミーデータを追加する。

【0052】

ステップS61の処理は例えば図5～図8の符号化サイズ決定処理の場合、1024KB、128KB、8KBの9割のデータサイズである912KB、115KB、7KBのデータブロックにダミーデータを追加して1024KB、128KB、8KBのブロックとするものである。

10

【0053】

ステップS62に進み、符号化装置33は符号化レベル n (1024、128、8)のデータブロックを n (1024、128、8)個のサブデータに分割する。それぞれのサブデータのデータサイズは前述の packetsize となる。

【0054】

ステップS63に進み、符号化装置33は符号化レベル n の行列式(符号化行列データ36)を取得する。ここでは、符号化レベル n の行列式が「 $n \times m$ 」の行列式であるとして説明する。

【0055】

ステップS64に進み、符号化装置33はステップS63で取得した行列式から i 行目の行列を取り出す。「 i 」の初期値は「1」であるとする。ステップS64で取り出される行列は、例えば符号化レベル n が「8」である場合、8つの要素から成る。なお、各要素は「0」又は「1」の値を有する。各要素はサブデータと対応付けられている。ステップS64で取り出される行列は、「1」の値を有する各要素に対応するサブデータ同士の排他的論理和を計算することを表している。

20

【0056】

ステップS65に進み、符号化装置33はステップS64で取り出した行列の「1」の値を有する各要素に対応したサブデータ同士の排他的論理和を計算して、 i 個目の符号化データとする。

【0057】

ステップS66に進み、符号化装置33は i 個目の符号化データにステップS64で取り出された行列を符号化情報として付け加えた送信 packetsize を作成する。図10は送信 packetsize の一例の構成図である。図10に示した送信 packetsize は、ブロック番号、ブロックサイズ、符号化情報、符号化データを有するように構成される。

30

【0058】

ブロック番号は「 k 」の値となる。ブロックサイズは符号化データからダミーデータを除いたデータサイズとなる。符号化情報はステップS64で取り出した行列となる。符号化データはステップS64で取り出した行列の「1」の値を有する各要素に対応したサブデータ同士の排他的論理和を計算したものである。

【0059】

ステップS67に進み、符号化装置33は送信 packetsize を送信装置34にUDPで送信させる。ステップS68に進み、符号化装置33はタイマー32をリセットしてステップS64に戻る。なお、「 i 」の値はステップS64～S68の処理を1回実行する度に1ずつインクリメントされる。ステップS64～S68の処理は「 i 」の値が「 m 」になるまで繰り返し実行される。

40

【0060】

ここでは、図9に示したブロックの符号化と送信処理について図11を参照しつつ説明する。図11はブロックの符号化と送信処理の一例のイメージ図である。図11では説明を容易とするため、符号化レベル n が「4」である例を表している。

【0061】

50

符号化装置 33 はブロック 100 のデータサイズが符号化レベル $n = 4$ に必要なデータサイズ 4 K B よりも小さい場合、末尾にダミーデータ 101 を追加して 4 K B のブロック 100 とする。符号化装置 33 は符号化レベル $n = 4$ のデータブロックを $n = 4$ 個のサブデータ A ~ D に分割する。サブデータ A ~ D のデータサイズは、前述のパケットサイズとなる。

【 0062 】

符号化装置 33 は、符号化レベル $n = 4$ の行列式 102 を取得する。ここでは、符号化レベル $n = 4$ の行列式が「 4×5 」の行列式 102 であるとして説明する。符号化装置 33 は取得した行列式 102 から 1 行目の行列「1100」を取り出す。なお、行列式 102 の行の各要素は、左から順番にサブデータ A ~ D と対応付けられている。

10

【 0063 】

1 行目の行列「1100」は、「1」の値を有する各要素に対応するサブデータ A 及び B の排他的論理和を計算することを表している。符号化装置 33 は、計算したサブデータ A 及び B の排他的論理和である符号化データ 103 と、行列式 102 から取り出した 1 行目の行列 104 とを有するように送信パケット 105 を作成する。以下、同様にして、符号化装置 33 は 2 行目 ~ 5 行目までの行列を順番に取り出し、送信パケット 106 ~ 109 を作成する。

【 0064 】

図 12 は符号化しないで送信する処理の手順を表した一例のフローチャートである。ステップ S71 に進み、符号化装置 33 はブロックのデータサイズをパケットサイズ毎に分割する。例えばブロックのデータサイズが 2001 Byte、パケットサイズが 1000 Byte の場合、符号化装置 33 は 1000 Byte のサブデータ 2 個と、1 Byte のサブデータ 1 個とに分割する。

20

【 0065 】

ステップ S72 に進み、符号化装置 33 は分割したサブデータのうち、パケットサイズに満たないサブデータの最後の端数の部分をダミーデータで埋め、パケットサイズのサブデータとする。例えばブロックのデータサイズが 2001 Byte、パケットサイズが 1000 Byte の場合、符号化装置 33 は 1 Byte のサブデータの末尾に、999 Byte のダミーデータを追加し、パケットサイズのサブデータとする。

【 0066 】

ステップ S73 に進み、符号化装置 33 は単位行列式を取得する。単位行列式は i 行目の行列の要素のうち、 i 番目の要素の値が「1」であり、それ以外の要素の値が「0」となるものである。

30

【 0067 】

i 番目のサブデータであれば、符号化装置 33 は i 番目の要素の値が「1」、それ以外の要素の値が「0」である行列を符号化情報とし、その符号化情報を i 番目のサブデータに付け加えた図 10 のような送信パケットを作成する。

【 0068 】

ステップ S74 に進み、符号化装置 33 は送信パケットを送信装置 34 に TCP あるいは UDP で送信させる。ステップ S75 に進み、符号化装置 33 はタイマー 32 をリセットしてステップ S73 に戻る。ステップ S73 ~ S75 の処理は作成した送信パケットが無くなるまで繰り返し実行される。

40

【 0069 】

ここでは、図 12 に示した符号化しないで送信する処理について図 13 を参照しつつ説明する。図 13 は符号化しないで送信する処理の一例のイメージ図である。図 13 では説明を容易とする為、ブロックが 4 つのサブデータ A ~ D に分割された例を表している。

【 0070 】

符号化装置 33 はブロック 110 をサブデータ A ~ D に分割し、分割したサブデータのうち、パケットサイズに満たないサブデータ D の最後の端数の部分をダミーデータ 111 で埋めてパケットサイズのサブデータ A ~ D とする。

50

【 0 0 7 1 】

符号化装置 3 3 は、単位行列式 1 1 2 を取得する。符号化装置 3 3 は取得した単位行列式 1 1 2 から 1 行目の行列「 1 0 0 0 」を取り出す。なお、単位行列式 1 1 2 の行の各要素は、左から順番にサブデータ A ~ D と対応付けられている。

【 0 0 7 2 】

1 番目のサブデータ A であれば、符号化装置 3 3 は 1 行目の行列「 1 0 0 0 」を符号化情報 1 1 4 とし、その符号化情報 1 1 4 を 1 番目のサブデータ 1 1 3 に付け加えた送信パケット 1 1 5 を作成する。以下、同様にして、符号化装置 3 3 は 2 行目 ~ 4 行目までの行列を順番に取り出し、送信パケット 1 1 6 ~ 1 1 8 を作成する。

【 0 0 7 3 】

次に、データ受信装置 4 0 の処理手順について説明する。データ受信装置 4 0 の復号装置 4 3 はデータ転送装置 3 0 からの送信パケットを受信装置 4 1 経由で受信し、復号処理を行う。図 1 4 は復号装置の復号処理を表した一例のフローチャートである。

【 0 0 7 4 】

ステップ S 8 1 に進み、復号装置 4 3 はデータ転送装置 3 0 から転送された図 1 0 の送信パケットを受信装置 4 1 経由で受信する。ステップ S 8 2 に進み、復号装置 4 3 は送信パケットに含まれる符号化情報の先頭から i 番目に「 1 」がある場合、復号スタックの i 番目に、受信した送信パケットの符号化情報及び符号化データを保存する。

【 0 0 7 5 】

ステップ S 8 3 に進み、復号装置 4 3 は復号情報の i 番目のビットを「 1 」にする。復号情報は i ビットのビット列であり、そのビット列の初期値を「 0 」とする。ステップ S 8 4 に進み、復号装置 4 3 は復号情報のビット列を後述の ACK として送信装置 4 2 経由でデータ転送装置 3 0 に送信する。

【 0 0 7 6 】

ステップ S 8 5 に進み、復号装置 4 3 は既に復号スタックの i 番目に、符号化情報及び符号化データが保存されている場合、既に復号スタックの i 番目に保存されている符号化情報及び符号化データと、新たに受信した送信パケットの符号化情報及び符号化データとのそれぞれについて排他的論理和を計算する。

【 0 0 7 7 】

排他的論理和により計算された符号化情報の先頭から i 番目に「 1 」がある場合、復号装置 4 3 は、復号スタックの i 番目に、排他的論理和により計算された符号化情報及び符号化データを保存する。なお、復号装置 4 3 は既に復号スタックの i 番目に、符号化情報及び符号化データが保存されている場合、既に復号スタックの i 番目に保存されている符号化情報及び符号化データと排他的論理和により計算された符号化情報及び符号化データとのそれぞれについて排他的論理和を計算する。この処理は繰り返し行われる。

【 0 0 7 8 】

ステップ S 8 6 に進み、復号装置 4 3 は復号情報のビット列が全て「 1 」でなければステップ S 8 1 の処理に戻る。復号情報のビット列が全て「 1 」であれば、復号装置 4 3 はステップ S 8 7 に進む。ステップ S 8 7 では、復号装置 4 3 が、 i 番目の符号化情報の i 番目ビットだけが「 1 」になるように、 i 番目の符号化情報及び符号化データと他の符号化情報及び符号化データとのそれぞれについて排他的論理和を計算する。即ち、ガウス消去法を行う。ステップ S 8 8 に進み、復号スタックの 1 ~ i 番目に保存されている符号化データを合成することで、メモリ 3 1 に蓄積されていたデータを復号できる。

【 0 0 7 9 】

ここでは、図 1 4 に示した復号処理について図 1 5 ~ 図 1 8 を参照しつつ説明する。図 1 5 ~ 図 1 8 は復号処理の一例のイメージ図である。図 1 5 ~ 図 1 8 では説明を容易とするため、符号化レベル n が「 4 」である例を表している。

【 0 0 8 0 】

復号装置 4 3 はデータ転送装置 3 0 から転送された送信パケット 1 2 1 を受信装置 4 1 経由で受信する。送信パケット 1 2 1 には、符号化情報「 1 1 0 0 」が含まれる。復号装

10

20

30

40

50

置 4 3 は符号化情報「1 1 0 0」の先頭から 1 番目に「1」があるため、復号スタック 1 2 2 の 1 番目に、送信パケット 1 2 1 の符号化情報及び符号化データを保存する。復号装置 4 3 は復号情報 1 2 3 の 1 番目のビットを「1」にする。

【0 0 8 1】

送信パケット 1 2 1 の次にデータ転送装置 3 0 から転送された送信パケットはロスしたものとす。復号装置 4 3 はデータ転送装置 3 0 から転送された送信パケット 1 2 4 を受信装置 4 1 経由で受信する。送信パケット 1 2 4 には、符号化情報「0 0 1 0」が含まれる。復号装置 4 3 は符号化情報「0 0 1 0」の先頭から 3 番目に「1」があるため、復号スタック 1 2 2 の 3 番目に、送信パケット 1 2 4 の符号化情報及び符号化データを保存する。復号装置 4 3 は復号情報 1 2 3 の 3 番目のビットを「1」にする。

10

【0 0 8 2】

復号装置 4 3 はデータ転送装置 3 0 から転送された送信パケット 1 2 5 を受信装置 4 1 経由で受信する。送信パケット 1 2 5 には、符号化情報「1 0 0 1」が含まれる。復号装置 4 3 は符号化情報「1 0 0 1」の先頭から 1 番目に「1」があるが、復号スタック 1 2 2 の 1 番目に、符号化情報及び符号化データが既に保存されているため、送信パケット 1 2 5 の符号化情報及び符号化データと、復号スタック 1 2 2 の 1 番目の符号化情報及び符号化データとのそれぞれについて排他的論理和を計算する。

【0 0 8 3】

復号装置 4 3 は、排他的論理和の計算により符号化情報 1 2 6 及び符号化データ 1 2 7 を得る。復号装置 4 3 は排他的論理和により計算された符号化情報 1 2 6 「0 1 0 1」の先頭から 2 番目に「1」があるため、復号スタック 1 2 2 の 2 番目に、排他的論理和により計算された符号化情報 1 2 6 及び符号化データ 1 2 7 を保存する。復号装置 4 3 は復号情報 1 2 3 の 2 番目のビットを「1」にする。

20

【0 0 8 4】

復号装置 4 3 はデータ転送装置 3 0 から転送された送信パケット 1 2 8 を受信装置 4 1 経由で受信する。送信パケット 1 2 8 には、符号化情報「1 1 1 1」が含まれる。復号装置 4 3 は符号化情報「1 1 1 1」の先頭から 1 番目に「1」があるが、復号スタック 1 2 2 の 1 番目に、符号化情報及び符号化データが既に保存されているため、送信パケット 1 2 8 の符号化情報及び符号化データと、復号スタック 1 2 2 の 1 番目の符号化情報及び符号化データとのそれぞれについて排他的論理和を計算する。

30

【0 0 8 5】

復号装置 4 3 は、排他的論理和の計算により符号化情報 1 2 9 及び符号化データ 1 3 0 を得る。復号装置 4 3 は排他的論理和により計算された符号化情報 1 2 9 「0 0 1 1」の先頭から 3 番目に「1」があるが、復号スタック 1 2 2 の 3 番目に、符号化情報及び符号化データが既に保存されているため、符号化情報 1 2 9 及び符号化データ 1 3 0 と、復号スタック 1 2 2 の 3 番目の符号化情報及び符号化データとのそれぞれについて排他的論理和を計算する。

【0 0 8 6】

復号装置 4 3 は、排他的論理和の計算により符号化情報 1 3 1 及び符号化データ 1 3 2 を得る。復号装置 4 3 は排他的論理和により計算された符号化情報 1 3 1 「0 0 0 1」の先頭から 4 番目に「1」があるため、復号スタック 1 2 2 の 4 番目に、排他的論理和により計算された符号化情報 1 3 1 及び符号化データ 1 3 2 を保存する。復号装置 4 3 は復号情報 1 2 3 の 4 番目のビットを「1」にする。なお、図 1 7 の下段に示した復号スタック 1 2 2 は保存された符号化情報が上三角行列となっており、復号成功を表している。

40

【0 0 8 7】

復号装置 4 3 は復号情報 1 2 3 のビット列が全て「1」になった為、復号スタック 1 2 2 の 1 番目の符号化情報の 1 番目ビット、2 番目の符号化情報の 2 番目ビット、3 番目の符号化情報の 3 番目ビット、4 番目の符号化情報の 4 番目ビットだけが「1」になるように排他的論理和を計算する。

【0 0 8 8】

50

例えば図18の例では復号スタック122の2番目と4番目とで排他的論理和を計算することで2番目の符号化情報が「0100」となる。また、例えば図18の例では2番目と4番目とで排他的論理和を計算したあと、復号スタック122の1番目と2番目とで排他的論理和を計算することで1番目の符号化情報が「1000」となる。

【0089】

復号装置43は図18の下段に示した復号スタック122の1～4番目に保存されている符号化データA～Dを合成することにより、データ転送装置30のメモリ31に蓄積されていたデータを復号できる。

【0090】

図19は、符号化装置のACKに対する処理の手順を表した一例のフローチャートである。ステップS91に進み、符号化装置33は受信装置35経由でデータ受信装置40から図20に示すようなACKを受信する。図20はACKの一例の構成図である。図20に示したACKは、ブロック番号、復号情報を有するように構成される。

10

【0091】

符号化装置33は受信したACKから復号情報のビット列を取得する。ステップS92に進み、符号化装置33は取得したビット列のビットに「0」が含まれており、復号情報に対応する送信パケットの送信中でなければ、ステップS93に進む。符号化装置33は符号情報のビット列のビットのうち、「0」であるビットに対応するサブデータ全ての排他的論理和を計算し、送信パケットで送信する。ステップS94に進み、符号化装置33は符号情報のビット列のビットのうち、「0」であるビットに対応するサブデータを、それぞれ送信する。

20

【0092】

(まとめ)

IPネットワークにおけるASPシステム1のデータ転送において、転送するデータを順次メモリ31に蓄積し、符号化に必要なデータ量については何段階か用意しておくことで、蓄積により発生する遅延を削減する。最大のデータ量を必要とする段階までデータが蓄積された場合には直ちに符号化してデータを転送する。また、最大のデータ量を必要とする段階までデータが入力されていなかった場合でも、あらかじめ決めておいた時間が経過した場合には、その段階で符号化を行ないそのデータを転送する。もし、最小の段階までデータが入力されていない場合には、そのデータは符号化せずにTCPもしくはUDPを用いた方法で転送を行なう。

30

【0093】

蓄積されたデータを符号化するには、蓄積されたデータを符号化の段階ごとに決められた数n個に分割し、n個のデータ同士で排他的論理和を計算することでn×m個の符号化されたデータを作成する。この時、データ同士の組み合わせ方法はn×mの行列式として用意することが可能であるため、あらかじめ消失に対する性能のよい行列を複数個求めておく。例えば、n=8、16、32、64、128、256、512、1024の8種類の行列を用意しておく。パケットサイズを1000Byteとすると、それぞれ符号化の為には最低で8000、16000、32000、64000、128000、256000、512000、1024000Byteのデータサイズのデータの塊(ブロック)が必要である。これまで、上記のデータサイズに満たない場合には、残りをダミーデータで埋めて符号化を行なっていたが、それでは送信するデータ量が増えるため効率が悪い。

40

【0094】

そこで蓄積されたデータのデータサイズを大きいブロックサイズの符号化から順に適用していき、どれにも満たない場合に符号化を行わない方法で無駄を削減する。例えば蓄積されたデータのデータサイズが1234567Byteであった場合、先頭の1024000Byteはn=1024の行列で符号化して送信する。次に残りの210567Byteの先頭の128000Byteはn=128の行列で符号化して送信する。同様に残りの82567Byteの先頭の64000Byteをn=64で符号化、16000Byteをn=16で符号化して送信する。最後に残った2567Byteについては符

50

号化しないで送信する。

【 0 0 9 5 】

データの組み合わせを表わす行列の行はビット列に変換してデータを転送する際に一緒に送信する。n個分は排他的論理和を取らずに元のn個のデータにし、m個の符号化されたデータだけを作成することも可能であり、符号化するより前にn個のデータを転送することが可能であるため遅延をより削減できる。しかし、mの数が小さい場合には消失に対する性能が悪くなることもある。符号化されないデータの場合には、分割したデータが何番目のデータであるかをヘッダに付加して転送する。

【 0 0 9 6 】

受信側のデータ受信装置40では受けとった符号化データの復元を試み、復元状況を返
10
信する。復元状況を受けとった送信側のデータ転送装置30では、受信側のデータ受信装置40がまだ復元できていないデータを、もう送信側のデータ転送装置30が送信していない場合に再送が必要と判断し、復元できていないデータの再送を行なう。この時n+m個のすべてのデータを送信するのではなく、受信した復元状況を元にして復元できていない部分だけを再送する。

【 0 0 9 7 】

本実施例によれば、従来のTCPを使ったデータ転送方法と比較して、RTTが大きい
20
時に、リアルタイム性を確保したまま転送速度を高速化できる。また、本実施例は符号化行列を使い分けることで、符号化して送信する処理系統と符号化しないで送信する処理系統とを同じように構成でき、データを復号する処理系統も同じように構成できる。

【 0 0 9 8 】

本発明は、以下に記載する付記のような構成が考えられる。

(付記1)

データを転送するデータ転送装置であって、

転送するデータの符号化に必要な複数種類のデータサイズのブロックが予め用意されており、前記複数種類のデータサイズのブロック毎に符号化行列を格納する符号化行列格納手段と、

データ蓄積手段に蓄積されたデータのデータサイズに基づき、前記ブロックをデータサイズの大きい方から順に決定する符号化サイズ決定手段と、

決定した前記ブロックの前記符号化行列を取得し、取得した前記符号化行列を用いて前
30
記データ蓄積手段に蓄積されたデータをデータサイズ毎に符号化する符号化手段と、

符号化された符号化データに、符号化に用いた前記符号化行列に関する符号化情報を付加したパケットを生成するパケット生成手段と、

生成した前記パケットを送信する送信手段と
を有するデータ転送装置。

(付記2)

前記符号化サイズ決定手段は、データ蓄積手段に蓄積されたデータよりもデータサイズが小さくなる前記ブロックのうちデータサイズが最も大きい前記ブロックを決定し、

前記符号化手段は、決定した前記ブロックの前記符号化行列を取得し、決定した前記ブ
40
ロックのデータサイズ分、前記データ蓄積手段から前記データを取得し、取得した前記符号化行列を用いて前記データ蓄積手段から取得した前記データを符号化する

付記1記載のデータ転送装置。

(付記3)

前記符号化サイズ決定手段は、前記ブロックのデータサイズを所定の割合だけ小さくした縮小データサイズと、前記データ蓄積手段に蓄積されたデータのデータサイズとを比較して、前記データ蓄積手段に蓄積されたデータのデータサイズよりも縮小データサイズが小さくなる前記ブロックのうちデータサイズが最も大きい前記ブロックを決定し、

前記符号化手段は、決定した前記ブロックの前記符号化行列を取得し、決定した前記ブ
50
ロックの縮小データサイズ分、前記データ蓄積手段から前記データを取得し、前記データ蓄積手段から取得した前記データにダミーデータを追加し、取得した前記符号化行列を用

いて前記データ蓄積手段から取得してダミーデータを追加した前記データを符号化する付記 1 又は 2 記載のデータ転送装置。

(付記 4)

前記パケットが連続して送信されない無送信時間を計時するタイマー手段と、

前記データ蓄積手段に蓄積されたデータのデータサイズが前記ブロックのデータサイズのうち最も小さいデータサイズより大きくなったとき又は前記パケットが連続して送信されない無送信時間が所定時間を経過したとき、前記符号化サイズ決定手段に前記ブロックのデータサイズを決定させる符号化サイズ決定指示手段とを有する付記 1 乃至 3 何れか一項記載のデータ転送装置。

(付記 5)

前記符号化サイズ決定手段は、前記データ蓄積手段に蓄積されたデータのデータサイズが前記ブロックのデータサイズのうち最も小さいデータサイズより大きくならず、前記パケットが連続して送信されない無送信時間が所定時間を経過したとき、前記データ蓄積手段に蓄積されたデータを符号化しないブロックとし、

前記パケット生成手段は、符号化しないブロックの前記符号化行列として単位行列を取得し、前記データ蓄積手段に蓄積されたデータに、前記符号化情報として前記単位行列に関する情報を付加したパケットを生成する付記 4 記載のデータ転送装置。

(付記 6)

データを転送するデータ転送装置のデータ転送方法であって、

転送するデータの符号化に必要な複数種類のデータサイズのブロックが予め用意されており、符号化行列格納手段が、前記複数種類のデータサイズのブロック毎に符号化行列を格納される符号化行列格納ステップと、

符号化サイズ決定手段が、データ蓄積手段に蓄積されたデータのデータサイズに基づき前記ブロックをデータサイズの大きい方から順に決定する符号化サイズ決定ステップと、

符号化手段が、決定した前記ブロックの前記符号化行列を取得し、取得した前記符号化行列を用いて前記データ蓄積手段に蓄積されたデータをデータサイズ毎に符号化する符号化ステップと、

パケット生成手段が、符号化された符号化データに、符号化に用いた前記符号化行列に関する符号化情報を付加したパケットを生成するパケット生成ステップと、

送信手段が、生成した前記パケットを送信する送信ステップとを有するデータ転送方法。

(付記 7)

データを転送するデータ転送装置として機能するコンピュータを、

転送するデータの符号化に必要な複数種類のデータサイズのブロックが予め用意されており、前記複数種類のデータサイズのブロック毎に符号化行列を格納する符号化行列格納手段と、

データ蓄積手段に蓄積されたデータのデータサイズに基づき、前記ブロックをデータサイズの大きい方から順に決定する符号化サイズ決定手段と、

決定した前記ブロックの前記符号化行列を取得し、取得した前記符号化行列を用いて前記データ蓄積手段に蓄積されたデータをデータサイズ毎に符号化する符号化手段と、

符号化された符号化データに、符号化に用いた前記符号化行列に関する符号化情報を付加したパケットを生成するパケット生成手段と、

生成した前記パケットを送信する送信手段として機能させるためのデータ転送プログラム。

【0099】

本発明は、具体的に開示された実施例に限定されるものではなく、特許請求の範囲から逸脱することなく、種々の変形や変更が可能である。データ転送装置 30 はデータ転送プログラムにより制御されたコンピュータであってもよいし、専用装置、半導体チップ又は PC カードなどであってもよい。

10

20

30

40

50

【0100】

また、転送するデータは画面データの他、監視カメラの画像データや音声データ等のリアルタイム性が高いことが求められ、且つ時間によって流れてくるデータの量が変化する如何なるデータであってもよい。

【図面の簡単な説明】

【0101】

【図1】ASPシステムの一例の構成図である。

【図2】データ転送装置の一例のハードウェア構成図である。

【図3】データ転送装置のメイン処理手順を表した一例のフローチャートである。

【図4】タイマー処理手順を表した一例のフローチャートである。

10

【図5】符号化サイズ決定処理手順を表した一例のフローチャート(1/4)である。

【図6】符号化サイズ決定処理手順を表した一例のフローチャート(2/4)である。

【図7】符号化サイズ決定処理手順を表した一例のフローチャート(3/4)である。

【図8】符号化サイズ決定処理手順を表した一例のフローチャート(4/4)である。

【図9】ブロックの符号化と送信処理の手順を表した一例のフローチャートである。

【図10】送信パケットの一例の構成図である。

【図11】ブロックの符号化と送信処理の一例のイメージ図である。

【図12】符号化しないで送信する処理の手順を表した一例のフローチャートである。

【図13】符号化しないで送信する処理の一例のイメージ図である。

【図14】復号装置の復号処理を表した一例のフローチャートである。

20

【図15】復号処理の一例のイメージ図(1/4)である。

【図16】復号処理の一例のイメージ図(2/4)である。

【図17】復号処理の一例のイメージ図(3/4)である。

【図18】復号処理の一例のイメージ図(4/4)である。

【図19】符号化装置のACKに対する処理の手順を表した一例のフローチャートである。

【図20】ACKの一例の構成図である。

【符号の説明】

【0102】

- 1 ASPシステム
- 10 ASPサーバ
- 20 クライアント
- 30 データ転送装置
- 31 メモリ
- 32 タイマー
- 33 符号化装置
- 34 送信装置
- 35 受信装置
- 36 符号化行列データ
- 40 データ受信装置
- 41 受信装置
- 42 送信装置
- 43 復号装置
- 50 ネットワーク
- 61 入力装置
- 62 出力装置
- 63 ドライブ装置
- 64 補助記憶装置
- 65 主記憶装置
- 66 演算処理装置

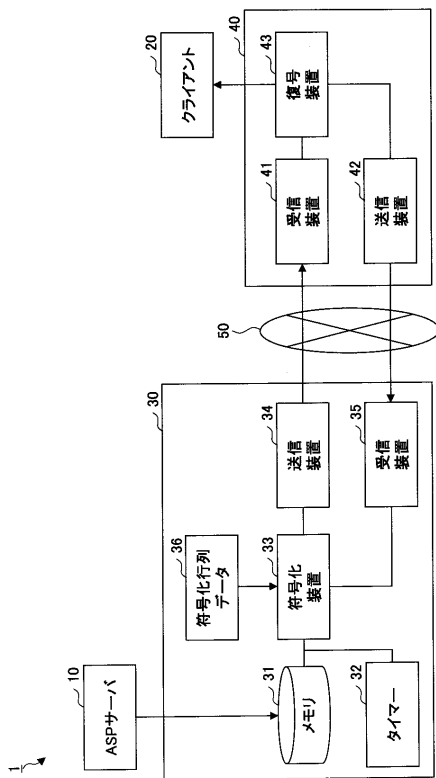
30

40

50

67 インターフェース装置

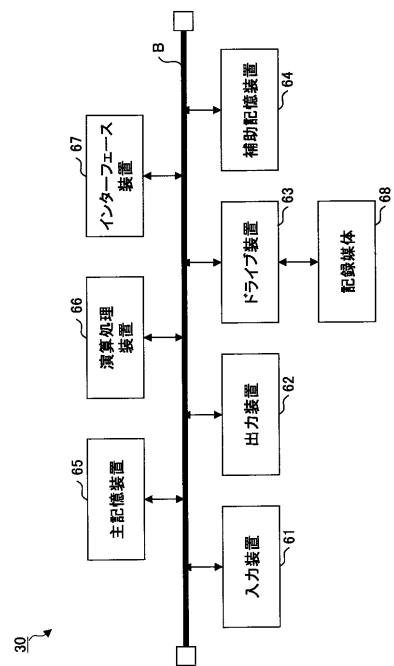
【図1】



ASPシステムの一例の構成図

【図2】

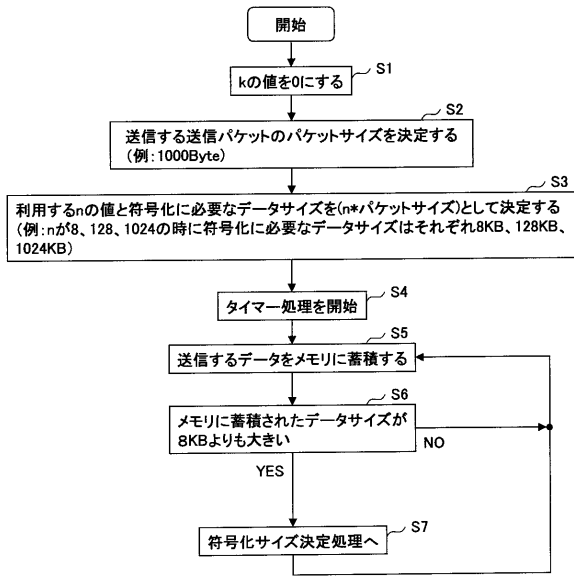
データ転送装置の一例のハードウェア構成図



30

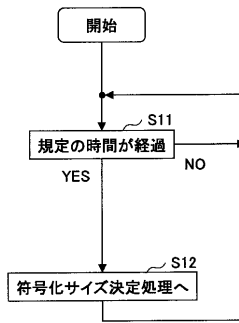
【図3】

データ転送装置のメイン処理手順を表した一例のフローチャート



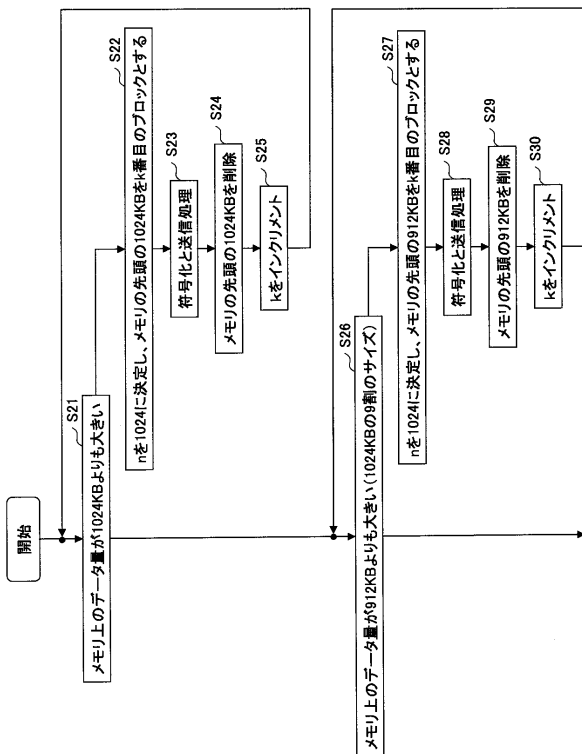
【図4】

タイマー処理手順を表した一例のフローチャート



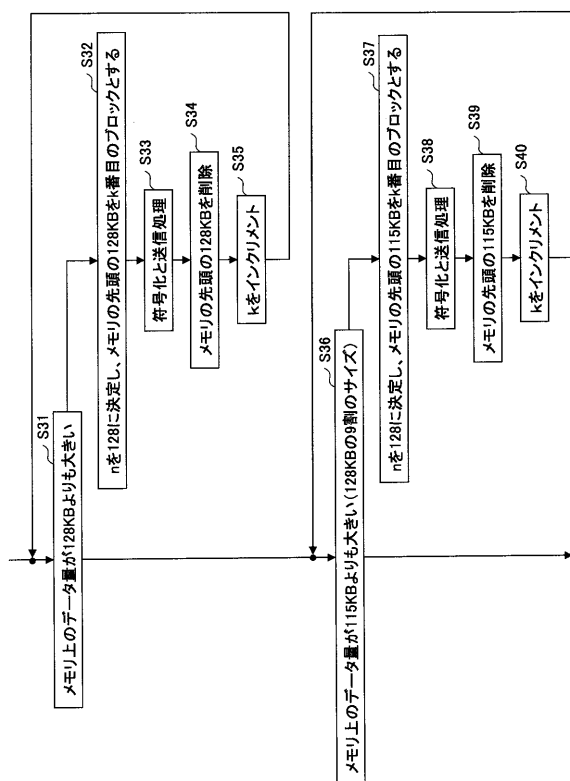
【図5】

符号化サイズ決定処理手順を表した一例のフローチャート(1/4)



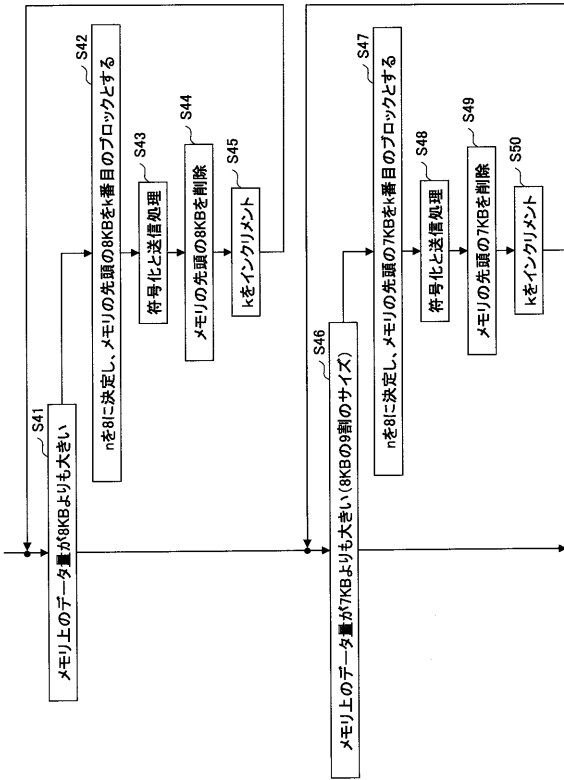
【図6】

符号化サイズ決定処理手順を表した一例のフローチャート(2/4)



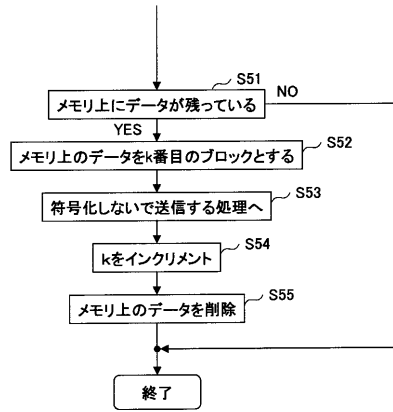
【図7】

符号化サイズ決定処理手順を表した一例のフローチャート(3/4)



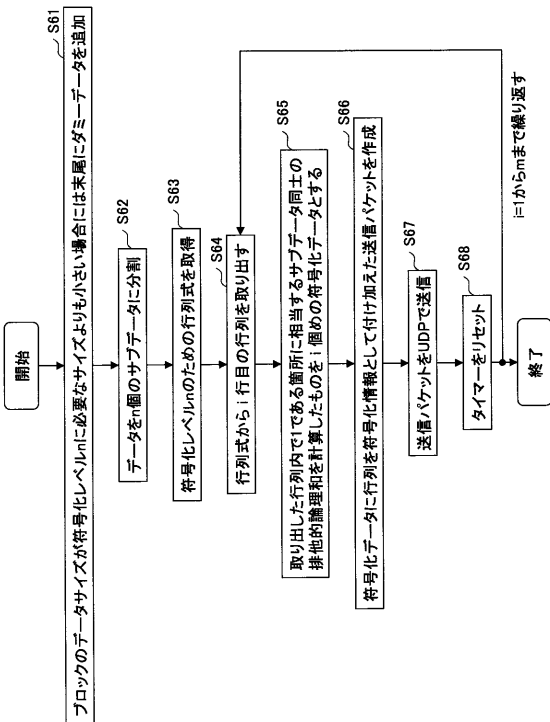
【図8】

符号化サイズ決定処理手順を表した一例のフローチャート(4/4)



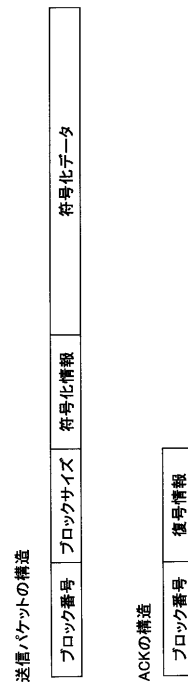
【図9】

ブロックの符号化と送信処理の手順を表した一例のフローチャート



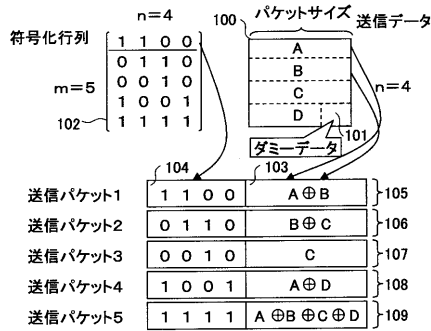
【図10】

送信パケットの一例の構成図



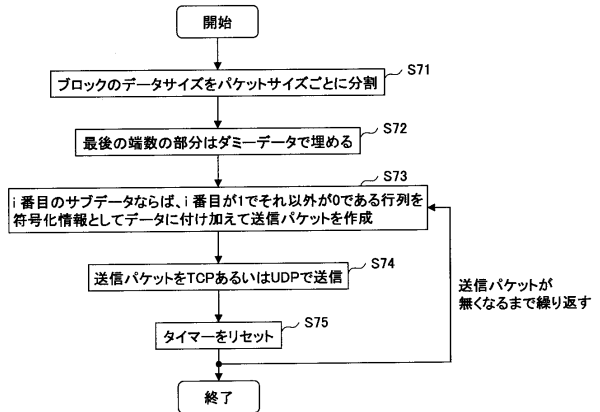
【図11】

ブロックの符号化と送信処理の一例のイメージ図



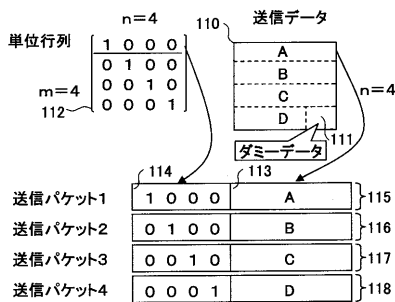
【図12】

符号化しないで送信する処理の一例のフローチャート



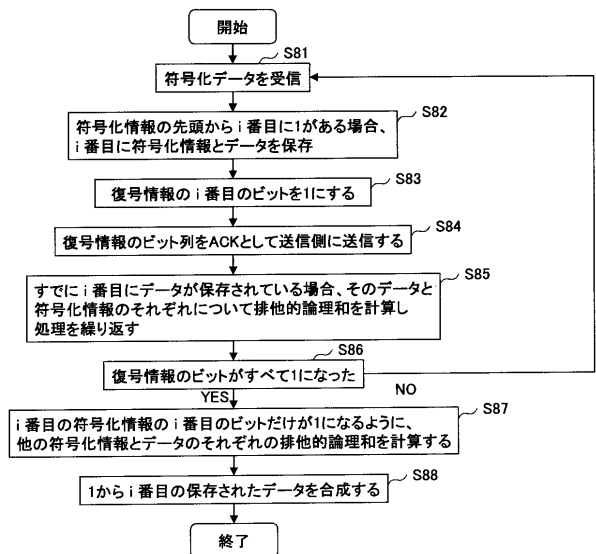
【図13】

符号化しないで送信する処理の一例のイメージ図



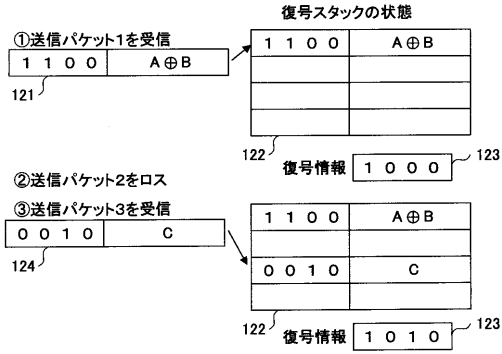
【図14】

復号装置の復号処理を表した一例のフローチャート



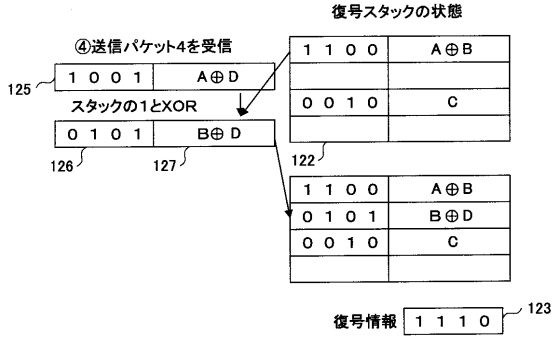
【図15】

復号処理の一例のイメージ図(1/4)



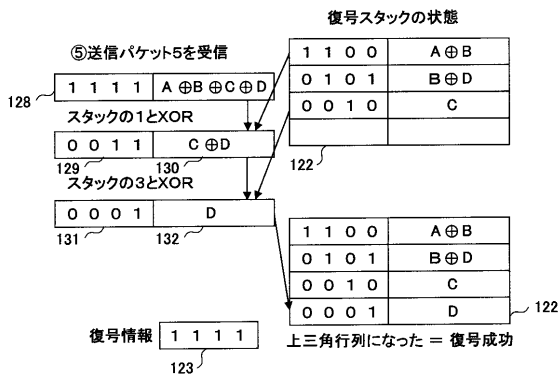
【図16】

復号処理の一例のイメージ図(2/4)



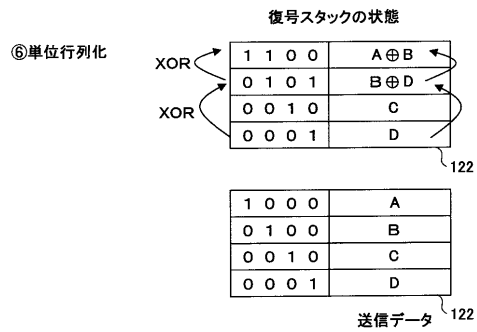
【図17】

復号処理の一例のイメージ図(3/4)



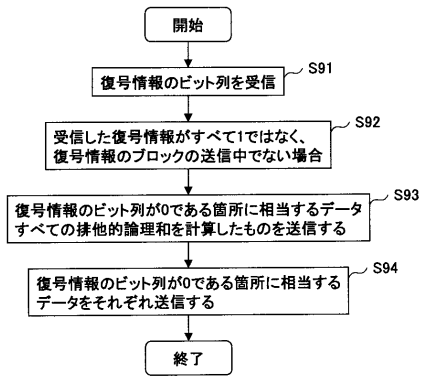
【図18】

復号処理の一例のイメージ図(4/4)



【図19】

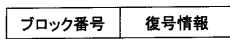
符号化装置のACKに対する処理の手順を表した一例のフローチャート



【図20】

ACKの一例の構成図

ACKの構造



フロントページの続き

- (56)参考文献 特開2006-157525(JP,A)
特開2006-325113(JP,A)
特開2005-347927(JP,A)
特開平10-285147(JP,A)
米国特許出願公開第2008/0151776(US,A1)
特開2005-175837(JP,A)

(58)調査した分野(Int.Cl., DB名)

H04L 1/00
H04L 12/70
H04L 29/08