

## 2.2 コンピュータネットワーク

田中英彦

田中英彦：正員 東京大学工学部電気工学科  
 Computer Networks. By Hidehiko TANAKA, Regular Member  
 (Faculty of Engineering, the University of Tokyo, Tokyo).  
 資料番号：昭 50-52 [特集-9]

### 1. はじめに

最近、複数の計算機システムを相互に結合して1つの系を構成するシステムが検討され実用に供され始めている。この電子計算機網(Computer Network)は、データ伝送・交換技術および、計算機タイムシェアリング利用技術の発達に因るところが大きい。しかし、この言葉の意味するところは様々で一般には、(i) データ交換網、(ii) リソース共有目的の複数計算機システム結合網、(iii) プロセッサが複数で1つのシステムを構成する計算機複合体、などを意味するようである。(i) は計算機間のデータ授受に使われ得るという意味で用いられ、諸国で検討されている新データ網がこれに当る。(ii) の代表はアメリカの ARPANET であり、又 (iii) はカーネギーメロン大学の Cmpmp に代表されるが、複数のプロセスを用いて処理を行うという意味からは (ii) と (iii) は同様である。このようなシステム以外にも、鉄鋼生産や化学プラントのプロセス制御システムなど、計算機を複数結合したものはあるが、処理形態と目的が固定的であるものが多く計算機網とは呼ばないのが普通である。

ここでは計算機網として上述の (ii) を想定するが、結合には一般に通信回線が使われることが多く通信系構成に関しては (i) と密接な関係を持ち、又、処理系に関しては (iii) と密接な関係にあるので、技術上の問題は複数計算機結合システムに共通するところが少なくない。

又、計算機網の解説は、既に本学会誌上で取上げられており<sup>(1)</sup>、目的・分類・実例について詳しく取扱われているのでそれらについては余り触れず、構成技術・問題を中心に概説する。

## 2. 計算機網

### 2.1 目的

計算機網の目的<sup>(1),(3)</sup> はリソース共有にある。例えば、計算負荷の共有による能率向上、特徴ある諸計算機が使える、自分の利用目的と要求される処理能力に適合したシステムが選べること、高価な特殊機器でも共有すれば安価であること、特徴ある共有ソフトウェアが使えること、データベースの共有が行えること、複数計算機による処理ができること、信頼性が向上すること、複数ユーザによる共同作業ができ、研究および

ソフトウェア開発の重複が避け易いこと、機種種のグレードアップに際し、旧機種が網内の他所にあれば新システムへの移行が容易であることなどである。

実際に網を利用しての報告<sup>(13)</sup>によると、自分のところの大形計算機のレンタルを解約し網を通して他所の同機種を使うことによりコストが40%に減少し、又、信頼性やサービスレベルも向上したこと、大規模計算のバッチ処理や会話形処理それぞれに便利な機種が明らかになり業務に応じて使い分けると便利であることなどが報告されている。又、複数計算機利用による大規模な実験も行われている他、利用の経験から、網内の人々間の連絡が緊密になり協調が容易になったと言われている。

2.2 網の概念

計算機網における網の概念は2つのレベルがある。1つは通信系で、実際の物理的伝送路による接続の表現として、又、データの送・受信者間関係の表現としての網概念であり、通信接続トポロジー、交換制御などに関連する。2つ目は処理系で、異なった計算機内にあるプロセス間の協調関係、分散データベース間の相互関係、プロセスに対するリソースの割当関係など、いずれも処理上の関係 (Association) を表現するものとしての網概念である。表1に代表的な計算機網(稼動中、製作中を含めて)の名前と場所、所属をあげる。これらの詳細については文献<sup>(1)~(3), (9), (29), (30)</sup>を参照されたい。

3. 通信系構成

3.1 システム構成

計算機網の構成要素を、処理システム、通信システム、ユーザ端末と大きく分ければ図1のように見ることができよう。

計算機間通信の伝送路としては、通常次のようなものが使われている。

- ① チャネル結合 (OCTOPUS, EPICS など)
- ② ポイントーポイント回線結合 (全銀ネット, TUCC)
- ③ データ交換専用・公衆網 (ARPANET, PCI) 網トポロジーとしては基本的に、星形、分散形 (リング, 蜂の巣形, 不規則形), メッシュがあり, この上に局階位を設けることが考えられる。1つの有用リソースを多くの計算機が共有する場合や, 処理の階層構成の場合は星形がとられることが多く (TUCC, Net/440), 同規模の計算機が地理的に離れ, 高い信頼

表1 計算機網の例

網名称	所在	所属	備考
ARPANET	米, 欧州	ARPA	本格的網
MERIT	米	ミシガン州3大学	三角形
TUCC	米	ノースカロライナ大など	360/370
OCTOPUS	米	LLL	バス結合
DCS	米	UCI	リングインタフェース
CYBERNET	米	CDC	商用網
Net/440	米	IBM	IBM 実験網
IBM TSS	米	IBM, カストマ	360/67 間
TYMNET	米, 欧州	Tymshare Inc.	商用網
GE Inf. Service	米	GE	最大の商用網
INFONET	米, カナダ etc	Computer Science Co.	商用網
JIPNET	日	情報開発センタ	パケット交換
EPICS	日	電総研	星形・バス結合
全銀ネットワーク	日	電電公社	全国銀行間
KUIPNET	日	京大	バス・インハウス
TECNET	日	東大	回線・インハウス
CYCLADES	仏	仏政府	パケット網
SITA	欧州	SITA	航空会社網
POLI	伊	諸大学	仮想端末結合

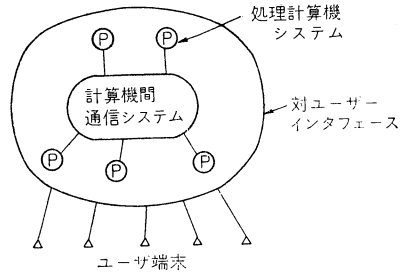


図1 計算機網

性が要求される場合にはその地理的配置を考慮した適当な分散形がとられる (ARPANET, CYBERNET). 又, 計算機が地理的に近く (同一建屋) にある場合には, 交換専用の計算機を配置し, 必要に応じて星形やメッシュに近い接続が取られることもある (GE スーパーセンタ, OCTOPUS).

網の利用者が網へアクセスするところの構成は, 基本的には図2のような4とおりが考えられる。③の例は ARPANET のミニホスト ANTS (ARPA Network Terminal System) であって, これは TIP<sup>(7)</sup> (端末 IMP) の持つ能力としての端末制御機能と, 計算機間メッセージ授受の基本制御機能とを併せ持つものである。④は処理計算機間通信網と端末網とが明確に区別されないもので, TSS サービスの発展形と考えられる (TYMNET)<sup>(4)</sup>。

3.2 通信制御

(1) 伝送制御

従来までによく使用されているデータリンク制御方式は, EBCDIC 符号や ASCII 符号を用いた BSC

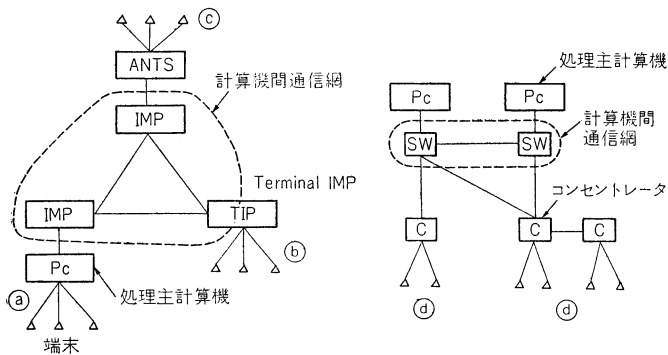
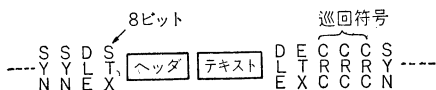
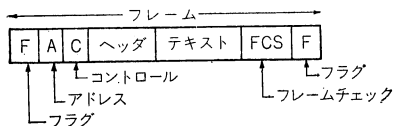


図2 種々のユーザインタフェース



(a) ARPANETの packets 形式



(b) HDLC方式のフレーム構成

図3 パケット(メッセージ)形式の2例

(Binary Synchronous Communication) 方式や、ISO 1 ビット符号を用いたベーシックモード伝送制御手順で、この他に新しい方式としてハイレベルデータリンク制御方式 (HDLC) がある。HDLC は伝送制御専用なので能率的で融通性に豊み、将来の広い利用が期待されている。図3に符号トランスパレントモードの伝送メッセージ形式の2例を示す。

(2) 交換方式

回線交換, メッセージ交換, パケット交換の3つがある。回線交換は、伝送路の設定時間が長いと大量のデータを長時間にわたって高速で伝送する場合に向いた方式であり、パケット交換は少ない情報を遅延少なく伝送するのに向いた方式である。例えば、MERIT は回線交換とメッセージ交換の組合せを用い、ARPANET はパケット交換を、CYBERNET, DCS, GE Net などはメッセージ交換を採用している。パケット網の場合交換網と処理計算機間の情報授受単位としては、メッセージとパケットとが考えられるが、パケットの再結合機能を前者では交換網が、後者では処理計算機が持つことになる。

データ伝送経路を定めるルーティングについては、

この特集3.4章(ネットワークにおける確率的問題)で扱われているので省くが、固定経路方式では信頼性に乏しいため、特に高信頼性が要求されたり網利用能率を重んじるには回方式(GE Net, TYMNET), 適応方式(ARPA-NET)などが使われている。又その制御は、TYMNET の場合は集中管理方式、ARPANET では分散制御方式である。

(3) フロー制御

計算機間通信はプログラム上から見るとプロセス間通信とみなすことができるが、通信系や計算機リソースを有効に利用するためには、この通信のフロー制御が不可欠である。これには2つのレベルが考えられる。1つは各計算機内プログラムにより通信フローを制御するもので、他は通信系自体が行うフロー制御である。前者が理想的に動作していれば後者の必要性は必ずしもないが、計算機の障害、処理の遅れによって通信系が影響を受けてしまうので、ある程度のフロー制御を持つことが望ましい。ここでは通信系のフロー制御<sup>(4)</sup>について述べる。

計算機間通信では、一般に、短い会話形メッセージを遅延少なく伝達せねばならないという要求と、送・受信者間のリンク設定時間は長くてもいいが、大量のデータを高帯域で伝送することの要求とがある。フロー制御はこれらを損なうものであってはならない。又パケット網の場合は、幾つかのパケット交換機内バッファが一杯になることによって通信が停滞する現象(ロックアップ)があり得るので、これを避ける対策が必要である。1つはリアセンブリロックアップで、これを避けるにはマルチパケットメッセージを送出する前にあて先交換機に対してバッファの予約をする必要がある。又、隣接する交換機内のバッファが互いに相手へ中継するパケットで満杯である場合のデッドロック(ストアアンドフォワードロックアップ)を避けるには、1つの出線にすべてのパケットバッファを割当てないとか、異常時用のパケットバッファを2つ用意しておく(オーバフロー網と呼ぶ)などの対策が必要である。

3.3 通信系の制御装置

通信系で利用される制御装置には次のものがある。

- ① 処理計算機と回線とのインタフェース装置
- ② データ交換装置(メッセージ/パケット交換機)

③ 端末制御装置

④ データコンセントレータとマルチプレクサ

計算機間の1対1接続には①が用いられ、通信制御装置 (CCU) と呼ばれている多くのものがこれに当る。

②はデータ交換機で、ARPANET の IMP<sup>(6)</sup> はこの②と①とを組合せたものと考えられる。③、④は端末と計算機網とのインタフェースで用いられるが、②と③、④を組合せたものが TIP<sup>(10)</sup> であり、③、④と基本的なホスト機能を組合せたものが ANTS である。これらの装置は従来までハードワイヤードなものが多く利用されているが、最近ではミニコンピュータやマイクロプロセッサをベースにして作りあげることが多くなっている。制御装置をプログラマブルにすることの利点は、主プロセッサの負担減少、変更に対する柔軟性、可用性の向上などである。このような目的で主計算機のフロントエンドとして使われるプロセッサをフロントエンドプロセッサ<sup>(5)</sup> (FEP) と呼んでいる。

3.4 計算機網間プロトコル

計算機網が各地に構成されていった時、これらを結合する要求は当然生じるであろう。しかし全網が共通の通信プロトコルを用いるとは考えられず、一般には異なるものと思われる。従って、これを予期したある程度の考慮や標準化はできるだけ早期から準備しておくべきであろう (現在、CYCLADES と NPL 間で実験中)。相互接続用プロトコルとして考えられているのは主としてパケット網の相互結合を対称としたもので<sup>(11),(12)</sup>、相互接続時の計算機相互間の通信手法やプロセス通信プロトコルは未検討である。

異なる網を結合するには、パケットのサイズと形式、アドレッシング、メッセージ形式とシーケンシング、伝送障害対策、エラー対策、フロー制御、End-to-End 制御、論理的なプロセス間接続の設定と解消、網間制御情報の連絡法、などの不一致に対処する必要が生じる。すなわち、全網上で共通に通用するフィーチャをどう設定するかが問題である。この設定の仕方にも、できる限り各網の自主性と独立性を尊重しようという行方<sup>(12)</sup>と、能率よい網を構成するために、ある程度の制約を設けようという行方<sup>(11)</sup>があるが、一

般に考えられている相互結合網形態は図4のようにゲートウェイ (GW) によって接続をするものであるが、網間の情報授受単位としてメッセージを用いる方式とパケットを用いる方式とが検討されている。

4. 処理系構成

計算機網をその目的どおりに有効に働かすためには、様々なソフトウェア上の問題を解決し又便利な制御システムを開発することが必要である。まず最も基本的に要求されることは、メッセージを諸計算機間で自由にやり取りできることであり、次に各計算機間のユーザプログラムと計算機間メッセージとの間の連絡を取る手法を定め、様々な計算機網利用形態に対して便利な援助プログラムを開発することが必要である。

4.1 計算機網のプロトコル

独立な計算機同志が協調しあうために、最も基本となるのが計算機間メッセージのやり取りの基本ルールで、一般にはその上に計算機内の処理単位であるプロセス間同志の通信基本ルールが設定される。これらはそれぞれ HOST/HOST プロトコル<sup>(3)</sup>、プロセス間通信プロトコルと呼ばれているが、通常の計算機網利用のための便宜はすべてこれらの上に構成されるものと考えることができよう。現在動いている諸計算機接続は必ずしもこれらの構造を意識して作られているわけではなく、HOST/HOST プロトコルとプロセス間通信プロトコルの区別が明確でないものや、通信しあうプロセスとしては非常に特殊化されたものしか対象にならないものが多い。しかし、計算機網のシステムソフトウェアを考える時はこのような分割による見方は有用である。

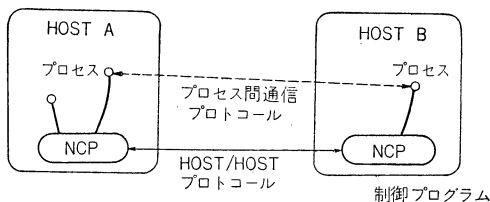


図5 計算機間通信の基本プロトコル

(1) 基本プロトコル

計算機内処理から見た通信プロトコルは2とおりの見方ができる。まず、一連のメッセージをあるプロセス間で通信しあう場合、メッセージ単位ごとに通信契約を交わすか<sup>(16)</sup>、コネクション関係を設定しその後は切断要求まで自由にメッセージを送るか<sup>(17)</sup>、の方式選択がある。前者は会話形の利用に向けた方法で、

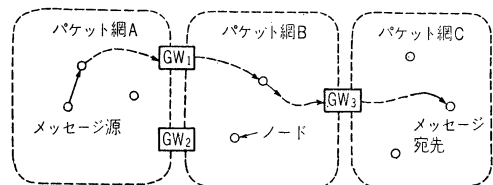


図4 パケット網の相互結合

後者は大量データの伝送に向けた方法であり、ARPANET では後者が用いられている。又、これら両者の中間に位置するモード変換形プロトコル<sup>(20)</sup>も提案されている。

次に、通信制御の形で分類すれば、共有メモリを介して通信しあうもの（集中形）と、明白な形でメッセージの交換が行われるもの（分散形）とがある。集中形プロトコルではその共有メモリに対するアドレッシングのルールを規定するという形をとり、分散形ではメッセージのアドレス・形式などを規定するものになる。集中形の例は GE Net<sup>(9)</sup> の MARK III の他、様々なディスク共有、メモリ共有による計算機結合方式があり、分散形の典型的な例は ARPANET である。

## (2) アプリケーションプロトコル

プロセス間通信プロトコルの上に様々な応用プログラムが作られるが、それにはまずジョブの生成、ログイン、ログアウト、割込などのサービスをする機能が必要で（ログプロセス）、次に基本的な計算機網利用形態（RJE、ファイルアクセス etc）に対するサービス用プロセスが作られる。これには、例えば次のようなものがある<sup>(19)</sup>。

- ① 初期コネクション関係設定用プロトコル
- ② 遠隔計算機端末からの会話形ジョブ用の仮想端末制御プロトコル（cf. ARPANET の TELNET）
- ③ ファイル転送プロトコル
- ④ リモートジョブエントリプロトコル
- ⑤ グラフィックスプロトコル

これらの働きをするプロセスの実装法として、各 OS を変更してシステムプロセスを作る方法と、ユーザプロセスとして付加する方法があるが、既存の OS を変更することが望ましくない場合は後者の手段がとられる。

## 4.2 データ記述

計算機網を構成する要素計算機のハードウェアや OS を、同種のもので構成すれば一般に問題は少なくより容易である。しかし異なったシステムが結合されているということは計算機網のかなり本質的な目的に関係することで、重要な利点の1つでもある。更に、同種計算機でも一般には接続装置や OS が異なることも多いであろう。従って異なったシステムの相互結合ということは利点を持つと同時に難しい問題を内在させることになる。

一般にデータ記述メッセージは3種ある。1つは言語プロセッサやライブラリプログラムなどで必要な変数とデータ構造についての詳細を記述するもの（変数の語長、次元 etc）で、2つ目は外部媒体と主記憶間のデータ転送用のシステム入出力ルーチンで使用される情報（物理レコードの最大サイズ、数 etc）で、3つ目はリソース割当をするシステムルーチンが必要とするデータ記述（データセットネーム、デバイスタイプ）である。

問題は、データ表現の差異とリソース割当手順などのファイル扱い手続きの差異であり、これに対する対策は次の2とおりが考えられる<sup>(20)</sup>。

- ① 各利用プログラム内でそのレコード変換をする。
- ② 各データセットに、標準のデータ記述言語でそのデータ記述をしたものを付けておく（cf. MERIT 網での確率応用プログラム向けの MICIS 標準）。

## 4.3 網利用上のハイレベルサービス

ハイレベルサービスとしては、計算機間の差異を緩和するためのサービス、リソース共有の自動化の他に、網情報の管理・測定<sup>(27)</sup>、運営機構などがある。以下、ARPANET などで実施（試み）されているその幾つかについて述べる。

### (1) データ再編サービス（DRS）<sup>(23), (24)</sup>

データ記述の対策は前述の他に、データ変換のサービスプログラムを置きインタプリティブに動かす方法がある。ARPANET ではこのメッセージのリフォーマットサービスを適当なホスト計算機上で行うようにしており、Data Reconfiguration Service と呼んでいる。このサービスでは、必要なデータ再編の記述（フォームと呼ぶ）を DRS 言語を用いて書き、それにネームを付けてコンパイルして記憶する。後程、そのデータ変換が必要になった時点でこのフォームをネームで呼び、データの供給プロセスと受領プロセスとを指定すれば、DRS インタプリタは、コンパイルされたフォームを取出して供給プロセスからの入力データ

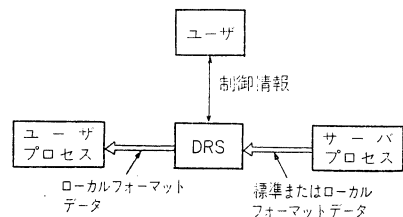


図6 データ再編サービス

トリームを処理し再編して出力ストリームを作り受領側に渡すものである。

#### (2) リソース共有の自動化

ユーザから見れば、ある問題に対して、諸計算機リソースのプログラミング言語、ファイル構成、ライブラリ、特殊機能の他、経済性、信頼性を考慮して使用する網内リソースを決定することが必要であるが、この決定を自動的にしようという研究がある。

##### ① RSEXEC (Resource Sharing Executive)<sup>(25)</sup>

ページングハードウェアを備えた PDP-10 用 OS (TENEX) 上で走る分散エクゼクティブシステムで、TENEX OS の諸機能を単一マシンの境界を越えて拡張し、多くの TENEX 計算機を含んだマルチホスト TENEX を可能にするものである。例えば、単一計算機内のファイルアクセス用のトリー構造が、全 TENEX 計算機内のファイルアクセストリーに拡張されており、ローカルファイルと同様にアクセスが可能である。又、特定のホスト計算機を持たない TIP ユーザに対し、プリログイン状態でも直接利用可能なコマンドを提供している。

##### ② 分散管理およびロード制御

計算処理の内、プログラミングシステムのユーザインタフェース部と、情報検索・管理部、計算部とをそれぞれモジュールとして分離し、これらの機能各々に対して適当な網内計算機を選ぶといった試みが ARPANET 上でなされている。又、リソースアロケーションを分散化したシステム実験に、カリフォルニア大学 (UCI) の DCS がある。

更に全ジョブのターンアラウンド時間を短く、又、全リソースが有効に使われるように、各計算機に加わるジョブのトラフィックを制御するセンタを設け、混んだシステムへのジョブは他の空いた計算機にまわすという方式の提案<sup>(26)</sup>や制御アルゴリズムの検討がある。

##### (3) McROSS (Multi-Computer-ROSS)<sup>(28)</sup>

分散処理に関する実験の1つとして、地理的に離れた計算機各々の中で走る多くの部分からなる処理をプログラミングし、更にそれを計算機網上で走らせることを可能にするマルチコンピュータプログラミングシステムである。具体的には、BBN社で開発された航空トラフィックモデリング用の、単一計算機によるシミュレーションシステム ROSS (Route Oriented Simulation System) の拡張で、ARPANET 上の TENEX ホスト計算機を要素計算機とするものである。

#### 4.4 データベースの構成

##### (1) ファイルの配置と網の構成

分散データベースの構成を考える場合、各データベース要素としてのファイルをどの計算機システムに配置するか、又、通信網を作る場合は伝送路をどのように張り伝送容量を幾らにすべきかが問題となる。この設計評価基準としては、例えば回線利用コストとファイルコストの和が考えられよう。従って、設計としては、諸計算機システムの位置(ノード)が与えられ、諸ファイルが与えられた場合に各ノードから各ファイルへのアクセスラヒック量を想定し、単位情報転送当りの通信回線コストを与えて、最も経済的なファイル配置と通信回線設定を定めることになり、種々の発見的手法が提案<sup>(21)</sup>されている。

##### (2) ファイルディレクトリ及びデータの完全性

情報アクセスの基本は、所望データに至るパスを与えるファイルディレクトリを用いることであろう。従って、ローカルには存在しないリモートファイルへのアクセス用のディレクトリをどう構成し、その内どれだけの情報をローカルに持つべきかが問題になる。

又、データファイルアクセスの基本パターンは、問合せと更新であるが、広くファイルを共有させると同一ファイルに対する問合せと更新が同時に発生し、ファイル内容の一貫性が取れなくなる恐れがある。従って、これを避けるための手段が必要である。

##### (3) データコンピュータ

データベースを実現する独立なシステムとして、例えばデータコンピュータ<sup>(22)</sup>が考えられる。これには、網ユーザがリモートデータにアクセスすることを可能にするデータ言語の設計開発、システムの基本記憶として何をどう用いるか (cf. レーザメモリ)、ローカル、リモートユーザに対する1つのデータユティリティとしてのデータコンピュータシステムの使用を検討することなどが問題になる。

## 5. おわりに

計算機網について、網の見地からその通信系構成、処理系構成について述べた。現在、計算機ハードウェアの構成に関しては、複数プロセッサシステムが注目をあびており、一方、データ通信については、計算機と結合し複数の計算機システムを利用した処理が現実のものになりつつある。いずれにせよ複数の処理システムを結合するシステムであって、この傾向は今後ますます強まるであろう。又、ヨーロッパでその兆が見

えているように、国の境界を越えた計算機網が構成されてゆくものと思われる。計算機網などのようなシステムは、それが開発され便利に発展することによってそれまでには見られなかった需要を喚起するものであり、現時点でその要求が多くないからといって開発を無意味とするのは早計であろう。特に、情報・知識の共有は文明発達の基本となっているものであって、これが効果的に実現されることの利点は測り知れないものがある。もちろん、情報の技術は両刀の剣であり、人権や機密を保護することは十分に考慮されねばならない。しかし、計算機網が広く普及するか否かは、それが如何に容易に且つ便利に、又、経済的に取扱えるかという点にかかっている。我が国における計算機網の今後の発展を望みたい。

### 文 献

- (1) 高月敏晴：“広域電子計算機網 [I], [II], [III]”, 信学誌, 57, 5~7 (昭 49-05~07).
- (2) D.W. Davies and D.L.A. Barber: “Communication networks for computers”, John Wiley & Sons Inc. (1973).
- (3) 元岡 達 (編): “計算機システム技術”, オーム社 (昭 48).
- (4) Special Issue on Computer Communications, IEEE Trans., COM-20, 3 (June 1972).
- (5) Special Issue on Computer Communication, Proc. IEEE, 60, 11 (Nov. 1972).
- (6) F.E. Heart: “Interface message processors for the ARPA computer network”, BBN, Quarterly Technical Report, No. 1-19.
- (7) D.C. Walden: “Users guide to the terminal IMP”, NIC-10916 (Sept. 1972).
- (8) P. Hirsch: “SITA-rating a packet-switching network”, DATAMATION, p. 60 (March 1974).
- (9) R.R. Hench and D.F. Foster: “Toward an inclusive information network”, FJCC 1972, AFIPS, p. 1235.
- (10) G.L. Fultz: “Adaptive routing techniques for message switching computer-communication networks”, Ph.D. Dissertation UCLA (1972).
- (11) V.G. Cerf and R.E. Kahn: “A protocol for packet network intercommunication”, IEEE Trans., COM-22, 5, p. 637 (May 1974).
- (12) L. Pouzin: “A proposal for interconnecting packet switching networks”, Eurocomp, Brunel Univ. (May 1974).
- (13) M.S. Sher: “A case study in networking”, DATAMATION, p. 56 (March 1974).
- (14) W.J. Barr: “Cost effective analysis of network computers”, Univ. of Illinois, UCDCS-R-72-538 (Aug. 1972).
- (15) E.W. Dijkstra: “Co-operating sequential processes”, Programming Languages, Academic Press, New York (1968).
- (16) D.C. Walden: “A system for interprocess communication in a resource sharing computer network”, Comm. ACM, 15, 4 (April 1972).
- (17) A. Mckenzie: “HOST/HOST protocol for the ARPA network”, NIC-8246 (Jan. 1972).
- (18) D.B. McKay and D.P. Karp: “Protocol for a computer network”, IBM Syst. J., 12, 1 (1973).
- (19) S.D. Crocker, et al.: “Function oriented protocol for the ARPA computer network”, SJCC 1972, AFIPS, p. 271.
- (20) D.H. Fredricksen: “Describing data in computer networks”, IBM Syst. J., 12, 3, p. 257 (1973).
- (21) R.G. Casey and T.D. Friedman: “Design techniques for data base oriented computer network”, IBM J., AD-763791 (May 1973).
- (22) T. Marill: “Network data handling system”, CCA Semiannual Technical Report, AD-730724 (Sept. 1971).
- (23) E.F. Harslem and J.F. Heafner: “The data reconfiguration service-an experiment in adaptable process/process communication”, ARPA Report, AD-737318 (Nov. 1971).
- (24) E.F. Harslem, J. Heafner and T.D. Wisniewski: “Data reconfiguration service compiler”, ARPA Report, AD-745751 (April 1972).
- (25) R.H. Tomas: “Resource sharing executive for the ARPANET”, NCC 1973, AFIPS, p. 155.
- (26) J.T. Fitzgerald: “Load regulation and dispatching in a network of computers”, Univ. of Illinois, UCDCS-R-72-537 (Aug. 1972).
- (27) L. Kleinrock and W.E. Naylor: “On measured behavior of the ARPA network”, NCC 1974, p. 767.
- (28) R.H. Thomas and D.A. Henderson: “McROSS-a multi-computer programming system”, SJCC 1972, AFIPS p. 281.
- (29) 坂井利之, 田畑孝一, ほか: “インハウス・コンピュータ・ネットワーク KUIPNET について”, 信学会電子計算機研資 EC 73-56 (1973-12).
- (30) 田中英彦, 元岡 達: “研究用電子計算機網 TEC-NET”, 同上, EC 73-57.

(文献 (1), (3) 内で引用されている文献は、直接必要なもの以外は上記から省略した)