ICOT Technical Report: TR-280

TR-280

# KBMS PHIにおける
# 分散問い合せ処理方式

高杉哲朗，羽生田博美，宮崎収兄
(沖電気)
伊藤英則

June, 1987



ICOT
Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan
(03) 456-3191～5
Telex ICOT J32964

Institute for New Generation Computer Technology

# KBMS PHIにおける分散問い合せ処理方式

高杉　哲朗*　　羽生田　博美*　　宮崎　収兄*　　伊藤　英則**

*：沖電気工業（株）　　**：（財）新世代コンピュータ技術開発機構

分散知識ベースシステムKBMS PHIにおける分散処理方式，特に問い合せ処理方式について報告する．PHIでは大量の共有知識の処理をデータベース技術を用いて効率的に行う方法の検討を行い，分散演繹データベースを中心にした実験システムの開発を行っている．

本報告では分散関係データベースと演繹データベースの検討をもとにそれらを統合した分散問い合せ処理方式について述べる．分散問い合せ処理方式としては先に同報型LAN に適した動的最適化を行うステージング方式を提案した．ここではこの方式を再帰問い合せに拡張した２段階ステージング方式を提案する．また通信管理やトランザクション管理などの分散制御方式の概要を述べる．

## Distributed Query Processing in KBMS PHI

Tetsuro TAKASUGI*　Hiromi HANIUDA*　Nobuyoshi MIYAZAKI*　Hidenori Itoh**

*:Oki Electric Industry Co. Ltd.,
Systems Lab., 4-11-22 Shibaura, Minato-ku, Tokyo, 108 JAPAN
**:Institute for New Generation Computer Technology,
Mita Kokusai Bldg. 21F, 1-4-28, Mita, Minato-ku, Tokyo, 108 JAPAN

This paper discusses the distributed query processing and distributed control in KBMS PHI. PHI is an experimental distributed knowledge management system that efficiently handles a large shared knowledge using the database technology. The kernel of PHI is a distributed deductive database system.

A distributed query processing strategy is developed based on the distributed relational database and the deductive database technologies. The paper describes this two phase staging strategy that is an extension of the staging strategy proposed by the authores to dinamically optimize query processing in the system connected by a broadcast-type local area network. It also briefly discusses the distributed control in PHI such as the communication management and transaction management.

## １．はじめに

知識情報処理分野においては，問い合せに対する演繹処理を行うデータベース（ＤＢ）システム，即ち，演繹ＤＢシステムに基づいた知識ベース管理技術の研究が注目されている．知識ベース管理技術における課題としては，大容量知識ベースの管理技術，処理の高速化技術や分散化技術がある．

第五世代コンピュータ・プロジェクトでは，これらの技術的課題に対する研究の１つとして演繹ＤＢに基づいた分散知識ベースシステムPHI の研究を行っている．演繹ＤＢの実現方式として，ホーン節と関係代数とが同じ論理的な基礎を持つことに着目し，ホーン節を関係演算に変換し，その関係演算を実行することにより処理の効率化を図る方式がある．PHI はこの方式を基礎とし，演繹ＤＢ機能を核とする知識管理部と分散ＤＢ機能を核とする分散制御部とからなる．

本稿ではPHI におけるホーン節問い合せに対する分散問い合せの処理の一方式として，２段階ステージングに基づく方式を報告する．この方式は[吉田86]で報告した方式を再帰問い合せ処理へ拡張したものである．まず，２章でPHI のシステム構成について，３章で分散問い合せを行う上での環境を設定する為の分散制御方式について，４章で分散問い合せの処理概要について，最後に５章で分散問い合せ処理について述べる．

## ２．ＰＨＩのシステム構成

PHI では分散環境下での大容量知識ベースの管理技術の確立を目標としており，そのシステム形態は分散した複数台の逐次型推論マシン(PSI) 上に個々の知識管理システムを配置した分散型の知識ベースシステムである[伊藤86]．個々の知識管理システムは関係ＤＢを核とした演繹ＤＢを基本に実現する[宮崎86]．2.1節でPHI の物理構成，2.2節でPHI の論理構成を述べる．

### ２．１　ＰＨＩの物理構成

PHI は物理的には図１に示す様に各サイトを狭域ネットワーク(LAN) で結合した構成となる．各サイトはPSI であり，それらを結ぶネットワークは分散問い合せを効率よく処理する為の３種類の通信形態を持つLAN (ICOT-LAN)[TAGU84]である．各サイトは自律しており独自の演算処理が可能であるとともに，必要に応じて他のサイトとの協調した処理を行うことも可能である．

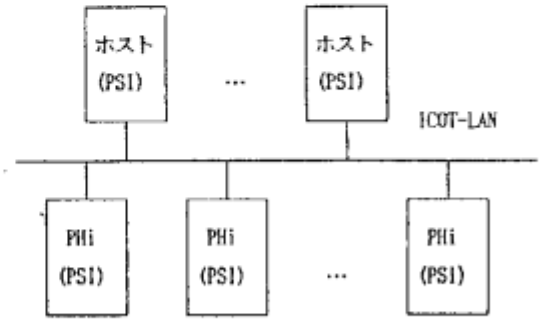


図１　PHIの物理構成

### ２．２　ＰＨＩの論理構成

PHI における論理構成を図２に示す[羽生87]．各サイトは機能的には知識管理層，分散制御部及びＤＢ管理層からなる．知識管理層はコンパイル法[BANC86]に基づく問い合せの処理を行いシステム内の知識への問い合せの管理，知識の一貫性を制御する．分散制御部は分散ＤＢに基づく処理を行い，ユーザに対してデータの透過性，各サイト間のＤＢの一貫性の制御を提供する．ＤＢ管理層は各サイトに閉じた関係ＤＢの管理を行う．ホーン節のファクトに相当する部分を外延ＤＢと呼び，同一名のファクト集合を１つの関係に対応させる．また，ルールに相当する部分（内包ＤＢ）は関係ＤＢのヴューの拡張に当り知識管理層で処理する．

PHI における問い合せの処理は，各サイト毎に各々のサイトが管理する知識ベースに対する処理を自律的に他のサイトと協調しながら行うことを基本としている．このような処理を実現する為，応用プログラムと

のインタフェース部となりシステム全体に共通した処理を行うグローバル知識ベース・マネージャ(GKBM)と，サイトに依存した処理を他サイトと協調して行うローカル知識ベース・マネージャ(LKBM)を動的に生成する．GKBMは応用プログラムに対応してシステム内に１つ生成され，LKBMは問い合せで使用する知識を持つサイト毎に１つ生成される．このモデルは[吉田86]におけるトラザクション・マネージャ(TM)－データ・マネージャ(DM)モデルを演繹問い合せに適するように拡張を行ったものである．

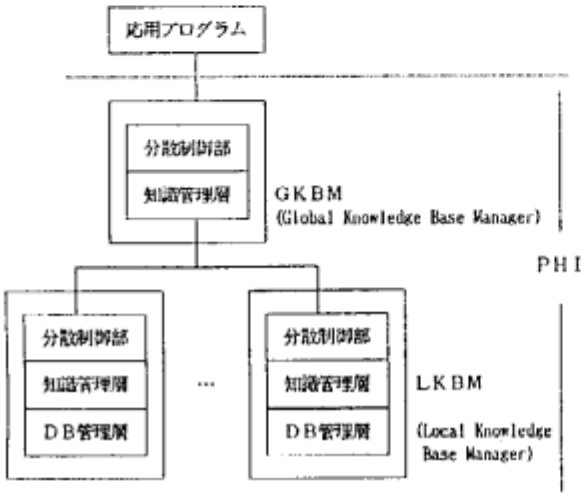


図2　PHIの論理構成

## ３．ＰＨＩにおける分散制御

PHI の分散制御について述べる．本章で述べる各制御方式は，４章以降に示す分散問い合せ処理を行う上で必要とされる環境を設定している．本章では分散制御部の諸機能の中で(a)通信管理，(b)データ管理，(c)カタログ管理，そして(d)分散トランザクション管理について述べる．

### ３．１　通信管理

本システムで重要となる通信管理方式について述べる．分散システムの場合，サイト間の通信を効率的に行うことが必須である．通常の１対１通信のみでは不十分であり，PHI で用いるLAN では同報機能を含む通信機能を提供する．PHI におけるネットワークの通信形態としては(a) １対１通信，(b)グループ通信，(c)同報通信がある．各通信形態はそれぞれに適した通信に用いられる．次に各通信形態の使用例を示す．

(a)LKBM間の関係の転送，もしくは送信の相手先が明確な場合の通信，
(b)内部問い合せの発行，問い合せ処理の中間的な結果の通知など処理サイトが限定できる通信，
(c)トランザクション制御（トランザクションの開始，終了）などシステム全体への通信．

### ３．２　データ管理

各サイト内のデータ（知識）の格納方式について述べる．応答速度の向上の為に関係を複数のサイトに複製し重複配置する方式がある．重複配置した場合の各重複データ間の一貫性の管理の問題と実現性とを考慮し，非重複分散配置とした．この場合でもネットワークの形態と同報機能により重複分散配置の場合と比べて著しい性能低下が生じないと思われる．

### ３．３　カタログ管理

応用プログラムから見たデータの位置透過性について述べる．ユーザが分散システムを利用する際にはデータ位置を指定しなくとも利用可能であることが望ましい．この要求に対し，PHI ではディレクトリをユーザ毎に決められたメイン・サイト上に置くことにより位置透過性を実現した．ディレクトリは関係のユーザ定義名とシステム内のグローバルな関係名との対応表である．トランザクション開始時に，GKBMはディレクトリを参照し，ユーザ定義名をグローバルな関係名に変換する．ディレクトリは同報通信を用いることを前提としている為，非重複分散型とする．

### 3．4　分散トランザクション管理

データベースの処理単位であるトラザクションの分散管理方式を述べる．複数サイトにまたがるようなトランザクションを管理する上で問題となるのが同時実行制御及びローカルＤＢ間の一貫性の保証である．PHI の研究では，分散知識ベース管理の基礎技術の確立を中心課題としている．従ってトランザクション管理方式はPHI が利用するネットワークの特徴を生かして不必要に複雑になることなくかつ十分な機能を発揮することを目標として次のような方式を採用した．GKBM がトランザクション開始時に事前に使用する関係名を同報しロックを得るとともに，該当する関係を格納しているサイトをメンバとしてトランザクション通信グループを形成する．トランザクション終了時には使用した関係をアンロックする．これは狭義２フェーズロック方式に基づいている．またコミットメント制御も同報機能に基づく集中型２フェーズコミットプロトコルを用いることとした．

## 4．問い合せ処理概要

PHI の研究では演繹ＤＢの処理技術そのものの確立と，それを分散環境下に適応させるという２つの課題がある．これらの課題はそれぞれが数多くの問題を含んでおり両課題を同時に統一的に検討することは困難であった．そこでPHI では分散ＤＢにおける効率的な問い合せ処理に対する検討と演繹ＤＢ処理に対する検討とをそれぞれ行い，最終的にそれらを統合するアプローチを採用した．本章ではこれらの各分野における問題点とその対処法，そしてそれぞれの方式を基礎とし両者を統合したPHI システム全体での問い合せ処理について述べる．

### 4．1　分散データベースにおけるステージング方式

分散ＤＢにおける問い合せ処理の問題点は，複数のサイトにまたがる演算からなる問い合せの最適化である．複数サイトにまたがる演算では関係の転送を必要とし，この転送コストは処理効率に大きな影響を与える．従って，各演算の実行サイトや演算の実行方式の決定とその関係の転送先サイトを決定することが重要となる．

従来この問題点に対しては，事前に演算の実行順序とそのサイトを決定するコンパイル方式による静的なアプローチが行われていた．しかしながら選択度のような情報により事前に戦略を決定する方式では非定型的問い合せに対し最適な実行方法が選択できるとは限らない．

この為PHI における分散ＤＢ処理ではLAN の同報機能を活用し，問い合せの実行時に最適な戦略を決定する動的なアルゴリズムを検討し，ステージング方式を提案した[吉田86]．この方式ではサイト間でそれらが非同期に処理を進めるのではなく，同期／協調の単位（ステージ）に分割しておきそのステージに基づく処理を行う．

ステージング方式では，まず問い合せ全体を１つのグラフと見て，各ノードを２項演算単位で葉ノードから１つずつ分解する．この１つのノードをステージと呼ぶ．このとき１つの子ステージを共有する複数のステージが存在する場合がある．この場合にはこれらのステージ間でその共有ステージの結果（関係）の転送プランを決定し，問い合せ処理の最適化を図る必要がある．この複数ノードからなるステージを複合ステージと呼ぶ．

ステージング処理が終った後，この問い合せは各ローカルサイトに対して発行される．各ローカルサイトはステージ内で自サイトで処理できるものから実行する．複合ステージを含め他サイトとの同期処理が必要な演算については該当するサイト間でメッセージを交換し，関係の転送先サイト等を決定しながら処理を行う．

### 4．2　演繹データベースにおける最小不動点演算

演繹問い合せが再帰的でない場合は，通常の関係問

い合せと完全に等価であり，その処理方式も関係ＤＢと同様に行うことができる．演繹問い合せ処理で問題となるのは再帰問い合せをいかに効率的に処理するかである．単純な再帰問い合せに対しては，効率的処理方法がほぼ確立しており，複雑な再帰問い合せに対しても幾つかの方式が提案されている[BANC86]．再帰問い合せ処理方式にはボトムアップ型とトップダウン型の処理があり，ボトムアップ型においては問い合せを関係演算に変換し，関係演算の実行により問い合せを処理するという特徴がある．トップダウン型には制約条件の伝播を行い必要なタプルのみを計算できるという特徴がある．

PHI ではこれらを踏まえてボトムアップ型の処理を基本として最小不動点(LFP )演算法の改良により再帰問い合せを処理する．この方式では，問い合せを述語の再帰性に基づいて幾つかの部分問い合せに分解し，これらの部分問い合せ間で制約条件の伝播を行い，各部分問い合せには差分展開法，制約付LFP 演算等の方式を適用し，再帰問い合せを処理する[MIYA86，羽生87，宮崎87]．このような方式を採用することにより，従来のＤＢ技術を有効に利用することも可能となる．

## ４．３　分散演繹データベース処理方式

本章では先に述べた分散ＤＢと演繹ＤＢとを統合した分散演繹処理方式について述べる．PHI 全体での問い合せ処理モデルについていては[羽生87]に報告したが，演繹ＤＢにおいて生成される関係代数をそのまま分散ＤＢにおいて実行した場合の再帰的処理の問題が残っている．即ち，再帰的処理が複数のサイトにおいて処理される場合を想定すると，4.1節で述べた分散ＤＢの処理では再帰を対象としていないので，例えば，同一の関係を繰り返しの各ステップで複数回転送するといった冗長な処理を行うことがあり効率的ではない．従って，再帰問い合せを考慮した処理戦略が必要になる．この点を考慮すると従来の関係ＤＢでのステージング処理だけでは効率化が望めない．

PHI ではホーン節レベルで問い合せを，相互に依存する述語からなる強連結成分に分解した依存グラフに基づいて，一段階目のステージング（マクロステージング）を行い，その各成分内での関係代数に対する二段階目のステージング（ミクロステージング）を行う２段階ステージングとする．この方式だとマクロステージングにおいて再帰的処理を１つの成分とすることにより再帰を意識しない処理単位に分類できる．各成分内では再帰に対応する繰り返し処理の実行サイトを指定し，繰り返しを必要しない処理は4.1節で述べた方式を行いサイト間にまたがる演算の効率化を行う．この方式により分散演繹ＤＢにおける再帰的問い合せ処理を効率的に行うことが期待できる．

## ５．分散問い合せ処理

本章では２段階ステージング方式の詳細を述べる．PHI では問い合せをまず，相互再帰を１つのまとまった成分として扱い，実行の単位（マクロステージ）に分ける処理，マクロステージングを行う．この後各マクロステージに対して関係代数を生成し，その関係代数に対し演算の実行同期単位（ミクロステージ）に分ける処理，ミクロステージングを行う．最後にステージ毎に関係代数の実行順序を動的に決定し実行する．以下２段階ステージングの詳細を述べる．

### ５．１　マクロステージング

マクロステージングでは強連結成分に対するステージ分割を行う．マクロステージの定義を定義１に示す．

［定義１］　マクロステージ

強連結成分分解された依存グラフに対し，ゴールまたは再帰述語を含む成分が依存している再帰述語を含まない部分グラフを１つのノードに縮退させたグラフをＧｒとする．Ｇｒ中の各ノードを１つのマクロステージとする．

依存グラフ[羽生87]は内包ＤＢ内のルールから述語間の（ヘッドからボディ述語への）依存関係を表す有向グラフであり，強連結成分は相互に依存し合う述語からなる部分グラフである．依存グラフを強連結成分分解したグラフでは，各ノードは相互再帰述語の集合，外延ＤＢ中の関係に対応する述語，非再帰述語またはゴールのいずれかとなっており，強連結成分によっては複数のサイトに関連する述語を含む場合がある．この場合にはさらに成分内でのステージ分割（ミクロステージング）が必要となる(5.3節参照)．

又，このとき問い合せが再帰処理を含まなければ，マクロステージングを適用するグラフＧｒはゴールノードと縮退ノード１個となり，これに対してミクロステージングが行われる．この場合，問い合せ処理は分散ＤＢで検討したステージング処理がそのまま適用可能である．

## ５．２　マクロステージの処理順序の決定

マクロステージング後の問い合せに対して，各マクロステージの実行順序を決定する．この順序の決定は，各マクロステージ間での制約条件の伝播に基づいて行う．問い合せを例に説明する（図３参照）．ｐ１及びｐ２に対応する関係が外延ＤＢ中に存在すると仮定する．

```
?-q(A, taro).
q(X,Y) :- p1(X,Y).
q(X,Y) :- r(X,Z), p1(Z,Y).
r(X,Y) :- p2(X,Y).
r(X,Y) :- p2(X,Z), r(Z,Y).
```

図３　問い合せ例

ステージングの対象となるグラフは図４のようになりｑ，ｒ，ｐ１，ｐ２がそれぞれマクロステージとなっている．この時与えられた制約条件(Y=taro)は最初ｐ１へ，ついでｐ１(A,taro)としてｒへ伝播する[羽生87]．従ってマクロステージの実行順序としてｐ１＞ｒが決定される．

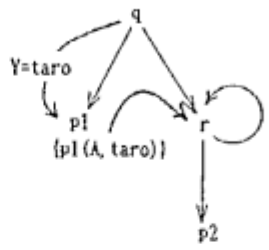


図４　制約条件伝播例

## ５．３　ミクロステージング

ミクロステージングは5.1節で述べたマクロステージ毎に行う．ミクロステージングは4.1節で述べた関係代数演算列に対するステージングを基本として，これを再帰述語処理に対して拡張したステージングである．ミクロステージとミクロステージングの際発生する複合ステージの定義を定義２，３に示す．複合ステージの概念を導入するのは，１つの関係が複数の演算で使用される場合の処理を効率化するためである．

［定義２］ミクロステージ

与えられた関係代数列に対し各演算を１つのノードとし，グラフを生成する．このグラフに対し葉ノードから２項演算毎に番号をつける．２個以上の経路がある場合にはその長いものをとる．この番号づけされたノードをミクロステージとする（図５参照）．

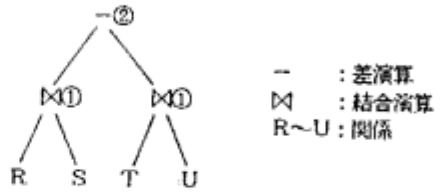


図5　ミクロステージ例

［定義３］複合ステージ

同一の番号を持つミクロステージの中で，１つの子ノードを共有するノードがある場合，このノード集合

を複合ステージとする（図6参照）．

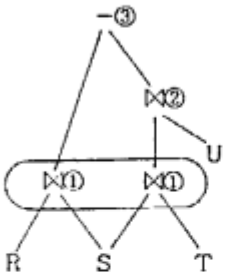


図6　複合ステージ例

ミクロステージングでは各マクロステージ内での関係代数列に対するステージ分割を行う．マクロステージが再帰処理を含む場合にはその再帰処理を実行するサイトの決定が必要となる．この点を考慮し各マクロステージにおいて，それらが再帰処理を含むか，含まないかにより以下の場合に分けて処理する．

(a)再帰処理を含む場合

まず，マクロステージに対応する関係代数演算列を生成し，ステージ分割を行う．このとき同一サイトに閉じた処理は1つのノードに縮退させて考える．次に，繰り返し処理は処理性能を大きく左右する為，繰り返し処理を行うサイトを決定する．

(b)再帰処理を含まない場合

まず，マクロステージに対応する関係代数演算列を生成し，この演算列に対して4.1節で述べたステージングと同様のステージングを行う．このとき同一サイトに閉じた処理は1つのノードに縮退させて考える．

## 5．4　関係代数の実行

ステージング処理が適用された問い合せは，マクロステージのレベルでは（必要に応じて）部分的に実行順序が規定され，ミクロステージのレベルでは，繰り返し処理の実行サイトが規定され，部分的に実行順序が規定されている．各LKDMはこの順序に従ってマクロステージの実行制御を行いながら，マクロステージ内での関係代数列を処理する．関係代数列はマクロステージ単位で生成され，マクロステージ単位で実行される．どのマクロステージから実行してゆくかは，5.3節で順序づけられたものはその順序で，順序関係のないものは動的に決定する．決定されたマクロステージ内の各関係代数列の実行について以下に述べる．

各マクロステージ内での関係代数演算は，(a)ローカル実行可能な演算，(b)他サイトとの同期を必要とする演算，(c)複合ステージ内演算，(d)繰り返し処理演算に分れる．これら各演算に対してミクロステージで決定された順序で実行してゆく．(a)〜(c)の演算については，必要に応じて動的な最適化，即ち演算の実行サイトや演算の実行方法の決定を行いながら問い合せを処理する．(d)の演算に対しては事前に実行サイトが5.3節のステージングにおいて決定されている為，ここではそのプランに基づいた実行を行う．(a)〜(c)は次に示すように実行される．

(a)ローカル実行可能な演算は単項演算もしくは自サイト内に閉じた2項演算である．この場合は他のサイトと関係なくその実行を行う．

(b)他サイトとの同期を必要とする演算は(c)の複合ステージの場合を除き，2サイトにまたがる2項演算である．例えば，2サイト間のジョイン演算がある．この場合はこの2サイトが互いに自サイト内の関係のサイズ等を交換しその結果により演算実行サイト，実行方法を決定する．

(c)複合ステージの場合は例えば3サイトにある関係を2個の演算が1つを共有しながら実行する場合である．この場合は3サイト間で必要な関係情報を交換し，その結果により演算実行サイトを決定する．この場合同報による転送回数の低減化効果及び並列処理の効果，もしくは1サイトに集めることによる転送量の減少効果が期待できる．

## 6．おわりに

分散知識ベースシステムPHI で検討中の２段階ステージング方式による分散問い合せ処理について述べた。２段階ステージング方式では，ホーン節問い合せに対する依存グラフのマクロステージング，複数のマクロステージ間の実行順序の決定，各マクロステージ内の関係代数列に対するミクロステージング，及び関係代数列の動的実行により分散演繹問い合せを処理する。本方式を用いることにより，演繹問い合せの特徴の１つである再帰述語の処理を分散環境下で効率よく処理することが期待できる。

今後は本方式の改良を行うとともに，他の方式との比較を行い本方式の有効性，妥当性を検討する予定である。

［参考文献］


[BANC86]：Bancilhon, F., Ramakrishnan, R. : An Amateur's Introduction to Recursive Query Processing Strategies, ACM SIGMOD, pp.16-52 (1986)

[MIYA86]：Miyazaki, N., et. al : Compiling Horn Clause Queries in Deductive Databases:A Horn Clause Transformation Approach, ICOT Technical Report TR-183 (1986)

[TAGU84]：A.Taguchi et.al:INI:Internal Network in the ICOT Programming Laboratory and Its Future, ICCC, Sydney, Australia, October (1984)

[伊藤86]：伊藤他：KBMS PHI(1):分散知識ベースシステムのシステム構成方式，情報処理学会第32回（昭和61年度前期）全国大会，2M-5 (1986)

[羽生87]：羽生田他：KBMS PHI，情報処理学会第34回（昭和62年度前期）全国大会，3K-5～9 (1987)

[宮崎86]：宮崎他：KBMS PHI(2):知識とデータの扱いに関する一考察，情報処理学会第32回（昭和61年度前期）全国大会，2M-6 (1986)

[宮崎87]：宮崎，伊藤：演繹データベースにおける制約付最小不動点，ICOT Technical Report (1987)

[吉田86]：吉田他：KBMS PHI(3):分散知識ベース制御方式，情報処理学会第32回（昭和61年度前期）全国大会，2M-7 (1986)