

2017 OR学会

SCSR研究部会

計画系システム開発に於けるソフトウェア・アーキテクチャ とOR アルゴリズムの連携の重要性

-- 開発事例の紹介 --

数理モデリング研究所

野末 尚次

URL: www.math-model.co.jp

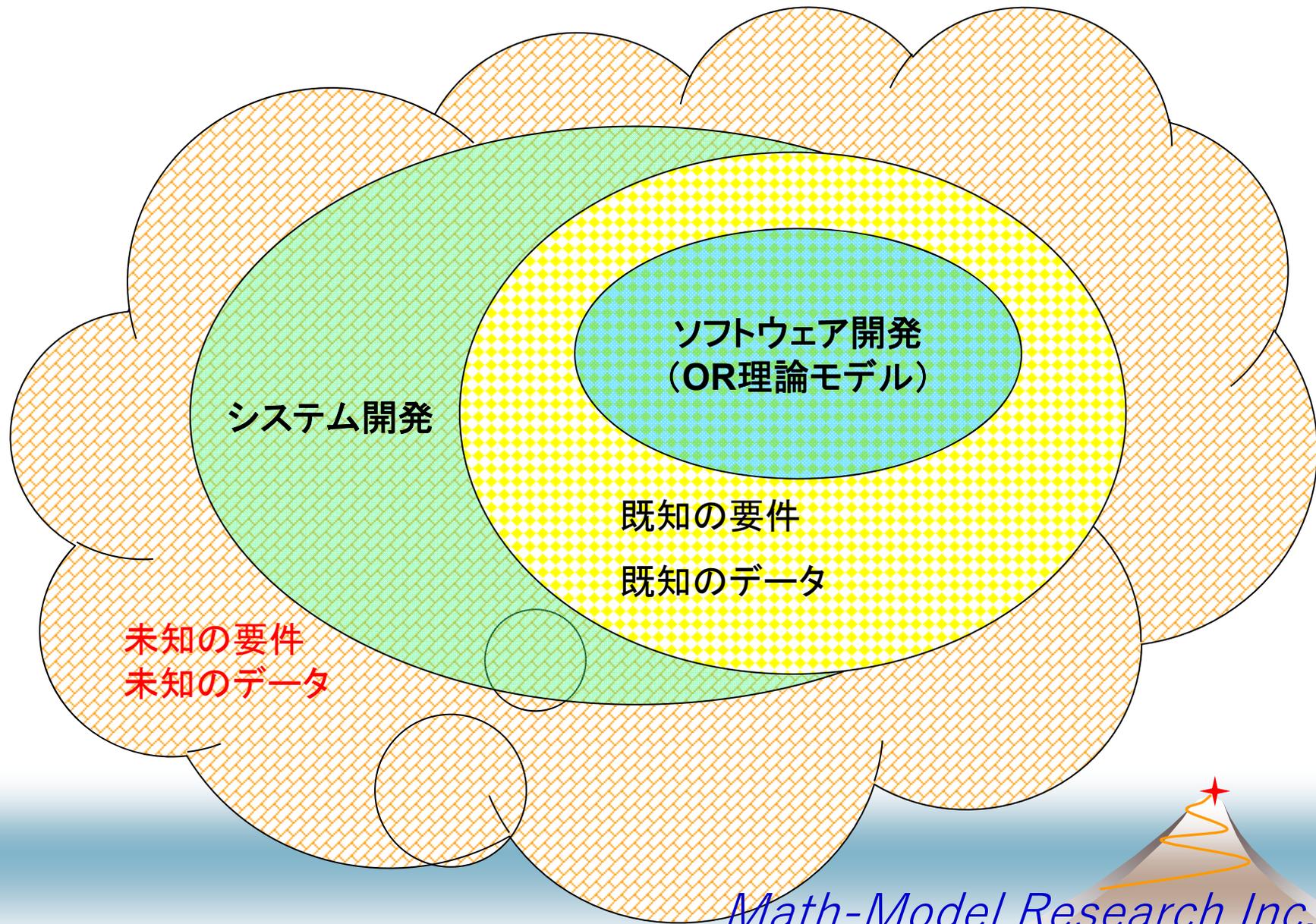


Math-Model Research Inc.

講演の概要

- ソフトウェア開発 (OR理論) とシステム開発の違いは？
- システム開発の課題：
 - 効率の良い頑健なアルゴリズムの考案
 - 開発効率の良いシステム・アーキテクチャの採用
 - メンテナンスを考えたソフトウェア・アーキテクチャの採用
- 非構造的な問題とこれに対するアプローチ法
- 大規模ソルバーに対する緩和アプローチ
- 宣言型システムの有効性とその実現アルゴリズムの開発
- 開発事例
- ORの理論によるモデル化は、なぜ上手く行かないのか？

ソフトウェア開発(OR理論)とシステム開発の違いは？



私の主な開発例

・汎用パッケージ

- ・ 非線形最適化理論
- ・ 最適化+確率効用理論
- ・ 地理情報処理(GIS)と国勢調査データ
- ・ データベース+AI

・交通計画関係

- ・ GIS+商圈分析モデル
- ・ GIS+交通需要予測モデル
+代替経路生成+幹線純流動調査
+仮想質問+主観的データ
- ・ 実績時系列データ+需要予測+GUI
- ・ 予約履歴データ+需要予測
- ・ ネットワーク(k-thパス)+鉄道営業規則
- ・
- ・ 知識工学+数理計画
- ・ GIS+リモートセンシング

・輸送計画関係

- ・ 制約ベースの計画ロジック
- ・ 制約論理と制約プログラミング

・宣言型汎用システム開発

- ・ 運賃規則の宣言的表現+Solverの開発

・信頼性管理関係

- ・ 時系列データ+磨耗予測

・ロジスティックス関係

- ・ メタヒューリスティックス
- ・ 混合整数計画法

・Deep Learning

- ・ フレームワークの利用法と適用可能性

- ・ 非線形最適化パッケージ
- ・ 非集計行動モデル推定システム
- ・ 地理情報処理システム
- ・ 交通知識ベースシステム

- ・ 商圈分析支援用DSS
- ・ バス路線計画支援システム
- ・ 全国新幹線網計画システム
- ・ MAGLEV(札幌~千歳空港)需要予測システム
- ・ 実績データによる波動予測・列車設定支援システム
- ・ 予約履歴ベースの動的列車設定計画のDSS
- ・ 鉄道運賃経路生成システム
- ・ 定期運賃計算システム
- ・ 列車乗継案内システム
- ・ 鉄道3D路線計画支援システム

- ・ 列車ダイヤ作成システム
- ・ 新幹線運転整理計画支援システム
- ・ 車両運用計画のDSS
- ・ 勤務割当計画のDSS

- ・ 宣言型運賃計算システム

- ・ 車載消耗品管理システム
- ・ 信頼性管理システム

- ・ 配送計画のDSS
- ・ シフト勤務計画のDSS
- ・ プロ野球のCSナンバー計算方式

- ・ 現在進行中

システム開発の課題

- 1) 効率の良いモデル化と頑健な**アルゴリズム**の考案
求解時間、解の品質、解の修正等に対応可能なアルゴリズムの採用
- 2) 開発効率の良い**システム・アーキテクチャ**の採用
分散処理など、システムの開発環境に合ったアーキテクチャの採用
制約プログラミング・混合整数計画等のソフトウェア・パッケージの利用
- 3) メンテナンスを考えた**ソフトウェア・アーキテクチャ**の採用
実使用のシステムは、環境の経年変化に適応する必要がある

システム開発の課題(その2)

1) 効率の良いモデル化と頑健なアルゴリズムの考案

- ・非構造的問題に対する緩和問題を利用したアルゴリズム
- ・AIのエキスパート・システムを利用した問題のチューニング
- ・地理情報処理に特化したGUIシステム
- ・非集計型需要予測システム

2) 開発効率の良いシステム・アーキテクチャの採用

- ・ワークフローと並列プロセスによる独立性の確保したアーキテクチャ
- ・測定系とネットワークで繋がる管理システムのアーキテクチャ
- ・AIのエキスパート・システムと連携したアーキテクチャ
- ・制約プログラミング・混合整数計画等のパッケージの利用

3) メンテナンスを考えたソフトウェア・アーキテクチャの採用

- ・宣言型の定義によるソフトウェアの構築
- ・混合整数計画による宣言的な開発
- ・制約プログラミングによる宣言的な開発
- ・ユーザーによるロジックの変更可能なShellによる開発

実用規模の計画問題の特徴

計画問題

機械・人間・材料・在庫・納期等の制約条件の下で、

- 制約条件を満たす計画案(実行可能案)を作成する。
- 実行可能案の中で、最も効率の良い案を求める。



- どのようなタイプの制約条件があるか？
 - 絶対守らなければならない制約(ハードな制約)
 - ペナルティは有るが、緩和が可能な制約(ソフトな制約)
- 評価指標はなにか？
 - 実行可能解が得られれば良い(制約充足問題)。
 - 実行可能解の中から、(準)最適解を求める(最適化問題)。

最適化問題の分類(形式的)

問題の定義要因	性質	分類	英語表記(略称)
変数のタイプ	実数	連続型	Continuous
	実数と整数	混合整数型	Mixed Integer (MIP)
	整数	整数型	Integer Programming(IP)
	記号	非数値型	Symbolic
制約条件 目的関数	線形	線形計画	Linear Programming(LP)
	2次式	2次計画	Quadratic Programming(QP)
	非線型	非線形計画	Non-linear Programming(NLP)
	論理/非数値	シンボリック	Symbolic
制約条件の有無	制約なし	非制約最適化	Unconstrained
	制約あり	制約付最適化	Constrained
データ	確定	確定的	Deterministic
	不確定	確率的	Stochastic Programming

最適化のアルゴリズム

探索方式	アルゴリズム	特徴
逐次的探索	ローカルサーチ メタ・ヒューリスティックス (タブ・サーチ、ガイドド・サーチ) エキスパート・システム	<ul style="list-style-type: none"> ・色々な問題に適用可能で、比較的簡単 ・最後の方になってダメと判明する！) ・探索空間が大きくなり、時間がかかる。 ・局所的な解に陥りやすい。
数理的探索	数理計画法(LP、QP、NLP) ネットワーク理論	<ul style="list-style-type: none"> ・方程式系により大域的な実行可能性を保証 ・探索空間の縮小が可能 ・適用可能な制約条件に制限がある。 ・専門的な知識による抽象的な定式化が必要 ・大規模な問題に適用可能
	混合整数計画(Branch and Bound 法) Column Generation 法	
制約充足的探索	制約論理 CHIP(COSYTEC) 制約ツールキット SOLVER(ILOG)	<ul style="list-style-type: none"> ・大域的制約条件の導入と制約伝播による実行可能性の確保 ・制約伝播による探索空間の縮小 ・複雑な制約条件に対応可能 ・多少の専門知識で、具象的な定式化が可能 ・大規模な問題には、適用が難しい。
確率的探索	遺伝子アルゴリズム シミュレーティッド・アニーリング	<ul style="list-style-type: none"> ・確率的に多くの解を生成して、良いものを求める。 ・実行可能解が沢山あり、簡単に求まる場合に適用が可能 ・遺伝子の定義が難しい。 ・最適化の達成度が判らない。

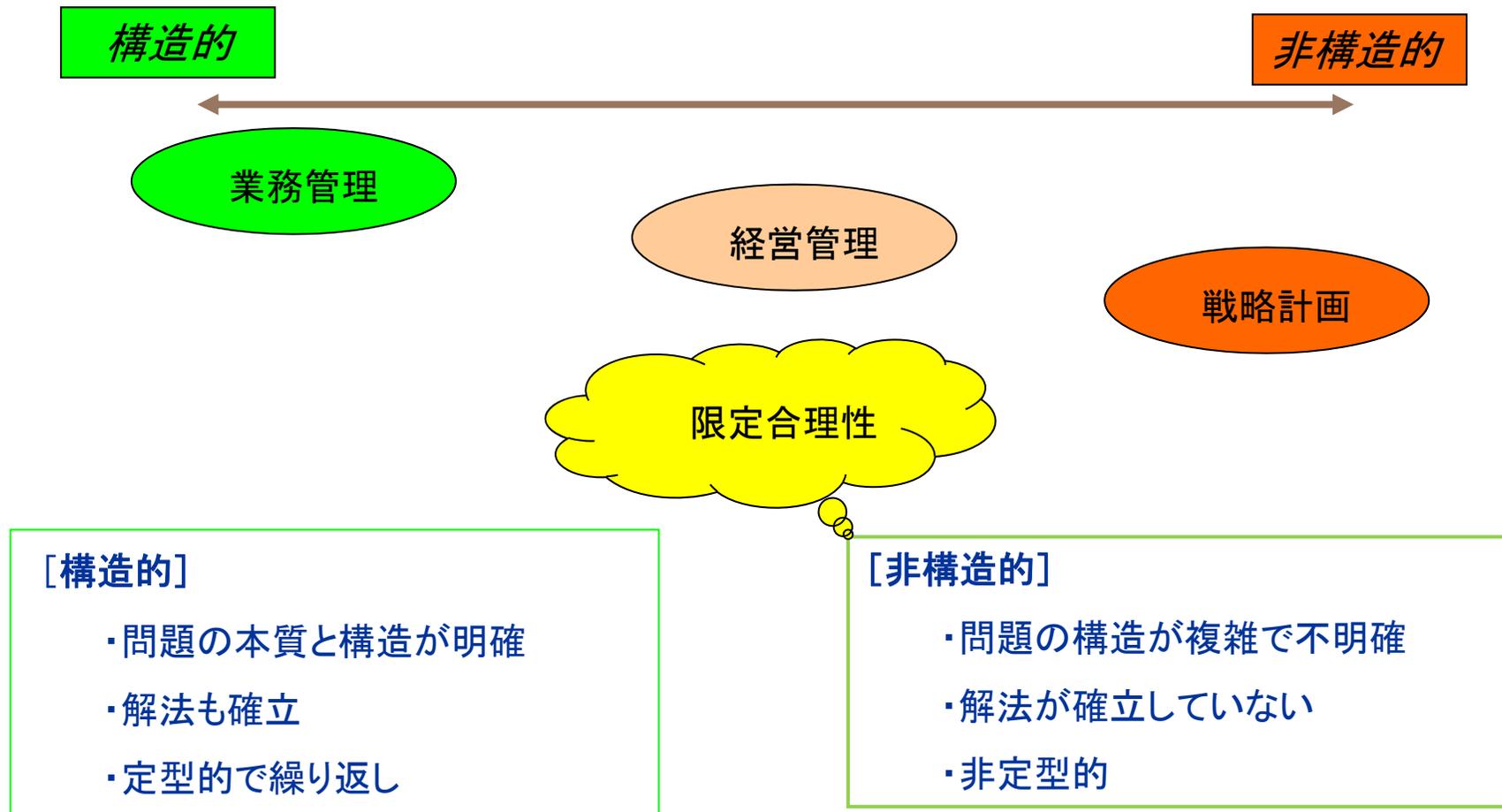
アルゴリズムは有効か

これらのアルゴリズムで、**実用できる解**が得られるか？

- 1) 実用上の制限時間内に、解が得られない。
- 2) 問題の定式化が不十分で、解が実使用できない。
- 3) 解法に合わせた定式化で、解が実使用できない。
- 3) 解法のコントロールが出来ず、メンテナンスが出来ない。

アルゴリズムの実使用は困難

意思決定問題の分類



意思決定とDecision Support System(DSS)

[Simon] 人間の意思決定過程は、次の3段階で構成される。

(1) 発見過程:

- ・関連情報の収集
- ・それに基づいた問題点の発見

(2) 設計過程:

- ・問題の定式化
- ・可能解の生成
- ・実現可能性のチェックによる代替案の設定

(3) 選択過程:

- ・複数の代替案の中から1つの案の選択
- ・実行案の作成

DSSは、これらの3段階を通じて、人間との共同作業を行うシステムである

非構造的な問題に対するアプローチ

- 計算モデル
- 1)モデル化
- 2) 計算が容易な**緩和問題**を定式化し、その目的関数の上/下限値の計算
- 3)メタヒューリスティックスによる目的関数の上/下限値を考慮した**複数代替案生成**
- 4)重複案の削除、
- 5)**2個の代替案**に対する**比較関数**の設定
- 6)**全ての代替案を比較関数でソート**
- 7)**最適案**の取り出し
- グラフィカル・ユーザーインターフェース
- 8)解の**グラフィカルな表示と問題点の抽出**

最適解探索アルゴリズムが**不要!**

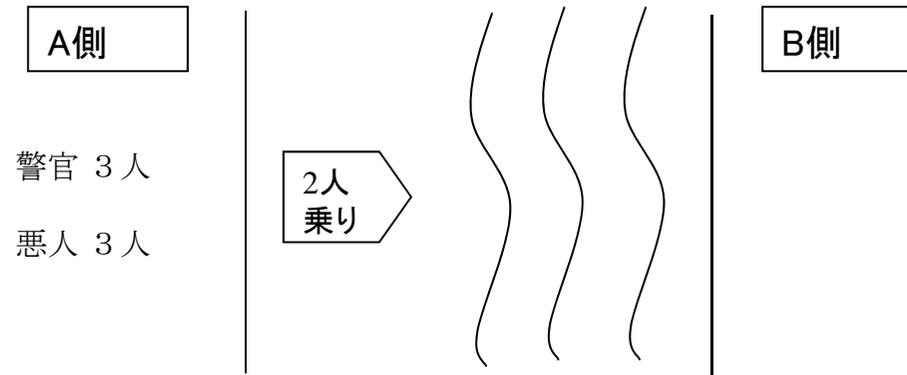
緩和問題の例:

混合整数計画: 整数条件の除外 → 通常のLP

XからYへ行く複数ルートの生成;

中間Mを経由するルートの長さ \geq (X->Mの最短距離 + M->Yの最短距離)

河渡り問題とモデル化

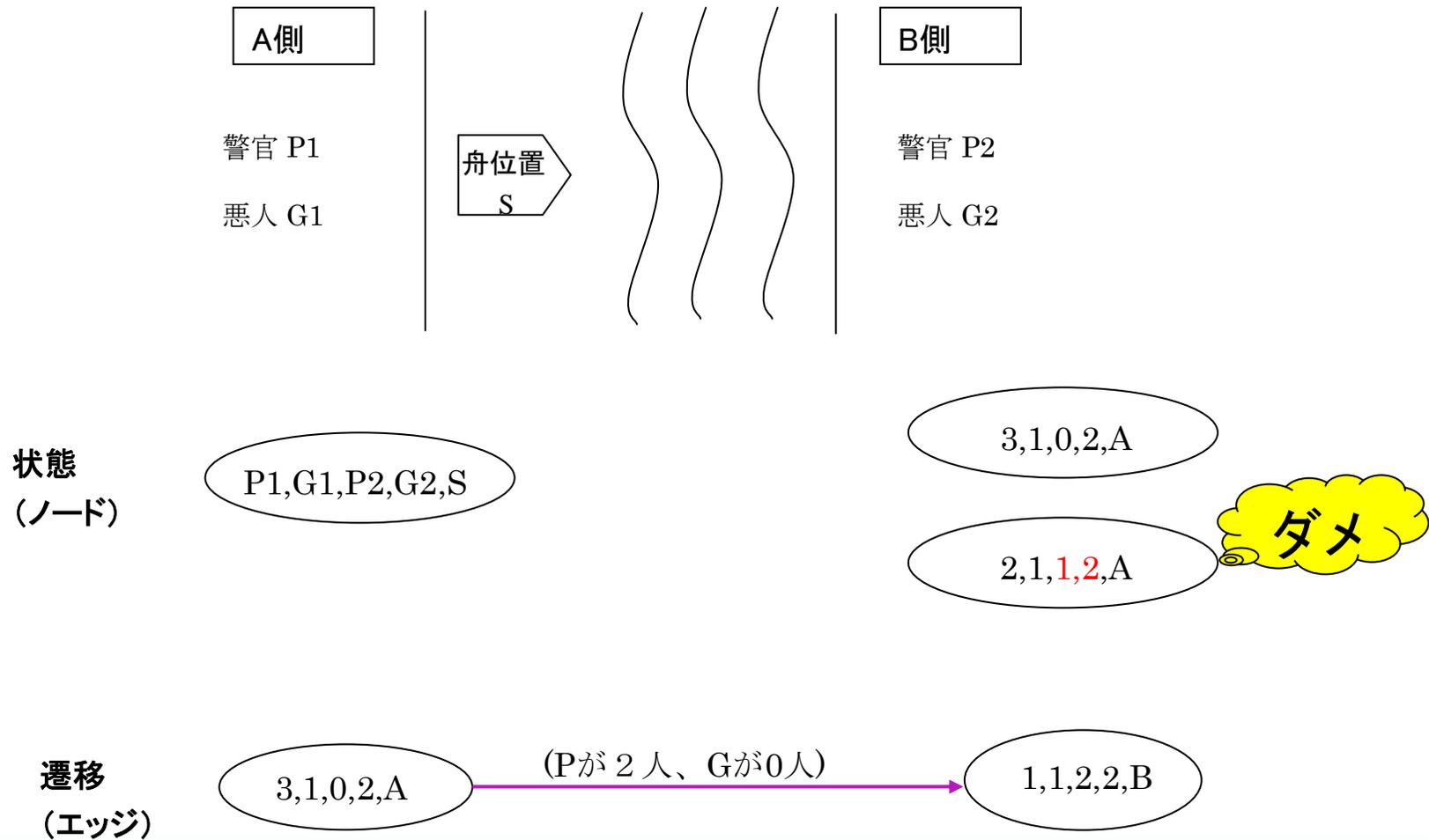


問題: 警官3人と悪人3人が、舟(2人乗り)でA側→B側へ渡りたい

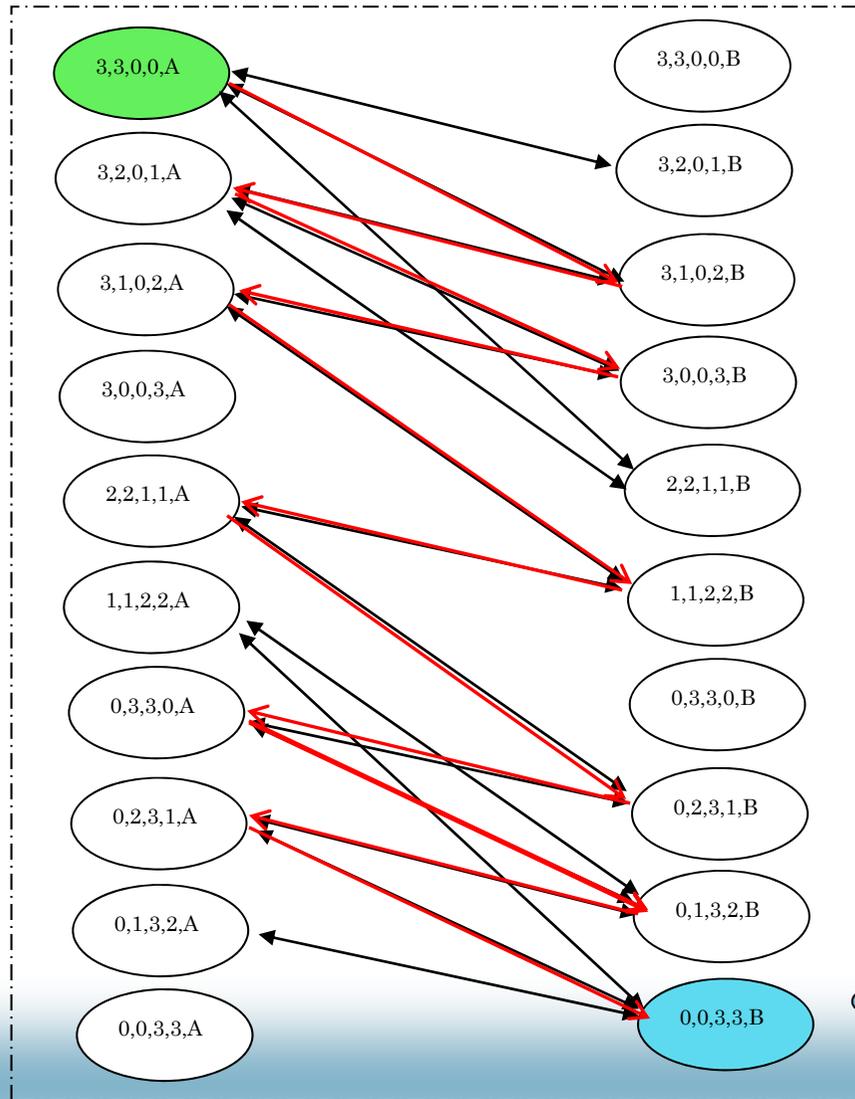
条件: 中間の状態では、次の条件の何れかが成立する必要がある。

- (1) 両方の岸とも、警官の数 \geq 悪人の数
- (2) 片側の岸は悪人だけ、反対側の岸は、警官の数 \geq 悪人の数

グラフによるモデル表現



モデル空間(全体のグラフ)



解法

(3,3,0,0,A)から
(0,0,3,3,B)への
道を見つける

宣言的アプローチ

「警官の移動が最小
の方法は」
枝の距離を警官の
数とし、最短ルート

最短路を使用した緩和問題:

全国幹線網需用予測システム: 鉄道・航空・自家用の全国260ゾーンを対象とした需用予測

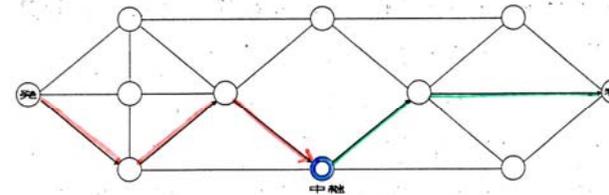
整備新幹線需要予測: 現況再現

対象年次	対象地域	鉄道パターン	航空パターン	道路パターン
1995	hokuriku	base 1995	base 1995	base 1995

分布交通量: 199512.h07

MEMO

現況再現を行い、実績値と比較する。
分布交通量を対象のフォルダに入れる。
*new_estV対象地域_dirV対象年VbaseVz_genkyou
例: C:\Crew\home2\New_est\Hokuriku_dirV1995VbaseVz_genkyou
ネットワーク:
鉄道(base_1995): 山形、秋田、北越、智頭開業
航空(base_1995):
道路(base_1995):

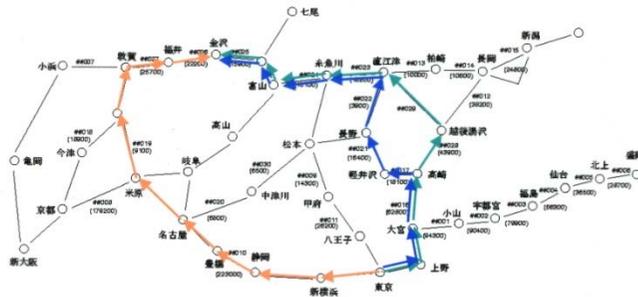


代替ルートの生成

$$\text{Min}\{\text{中継ノードを経由するルートの距離}\} \geq$$

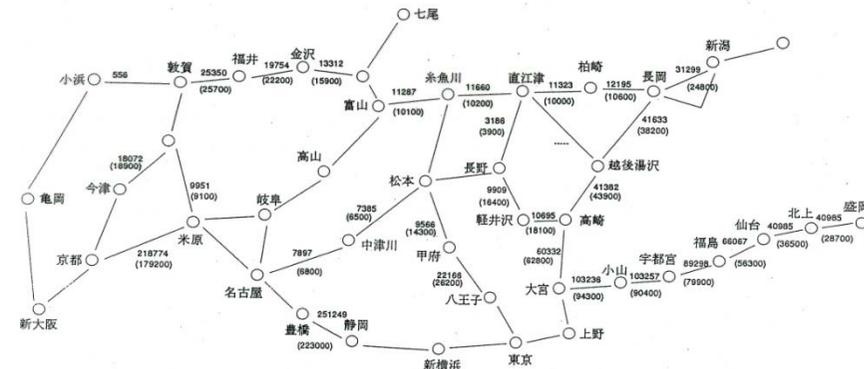
$$\text{発から中継ノードへの最短距離} + \text{着から中継ノードへの最短距離}$$

径路の自動生成と LOS の自動計算による多経路配分



計算条件: 東京～金沢 (長野新幹線なし、北越北線あり)

- 上越新幹線・北越経由 : 7.9%
- 東海道新幹線・米原経由 : 1.4%
- 信越線経由 : 7%



- ・全国260ゾーンを対象に、鉄道・航空・自家用の経路・運賃・時間・最適列車乗継案の自動積算、
- ・旅客純流動調査データを使用
- ・分担モデルを**AWK**で記述し、ユーザーが修正可能

Math-Model Research Inc.

大規模ソルバー利用時の緩和アプローチ

- 1) 制約条件を緩和して、得られた解の制約違反を解消する制約を追加
 - ・配送計画におけるサブツアーの禁止
- 2) 解空間を狭く緩和して、得られた解をベースに解空間を拡張する
 - ・カラムジェネレーションによる解空間の拡張
- 3) 非線形制約を解消するために、一部の 변수を固定して解を求める。
 - ・クリンチナンバー計算に於ける有効試合数のパラメータ化
- 4) 解空間を狭めるための動的な制約伝搬
 - ・MIPにおけるカットの追加
 - ・制約プログラミングの大域的制約

サブツアー禁止の緩和

2日間集荷ルート決定問題

一台のトラックにより、2日間の集荷を行うが、毎日の農家と1日のみの農家がある。

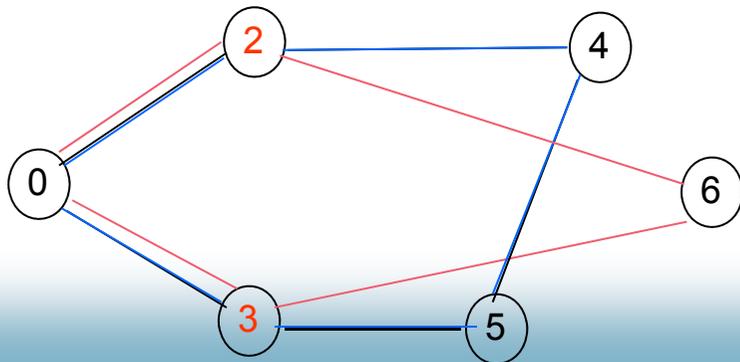
トラック: 容量 C 、移動コスト C_{ij}

農家: 全体の集合 N 、毎日: $N1$ 、1日: $N2$

集荷量 K_i

X_{ij}^k : k 日目に (i, j) 間を運行

Y_i^k : k 日目に農家 i を集荷



$$\text{Min} \sum_k \sum_{i < j} C_{ij} * X_{ij}^k$$

$$\sum_i K_i Y_i^k \leq C \quad k=1,2$$

$$Y_i^1 + Y_i^2 = 1 \quad i \in N2$$

$$\sum_{j:j>i} X_{ij}^k + \sum_{j:j<i} X_{ji}^k = 2 \quad k=1,2, i \in M$$

$$\sum_{j:j>i} X_{ij}^k + \sum_{j:j<i} X_{ji}^k = 2Y_i^k \quad k=1,2, i \in N2$$

$$X_{ij}^k \leq Y_i^k \quad k=1,2, i, j \in N2, i < j$$

$$X_{ji}^k \leq Y_i^k \quad k=1,2, i \in N2, j \in N, j < i$$

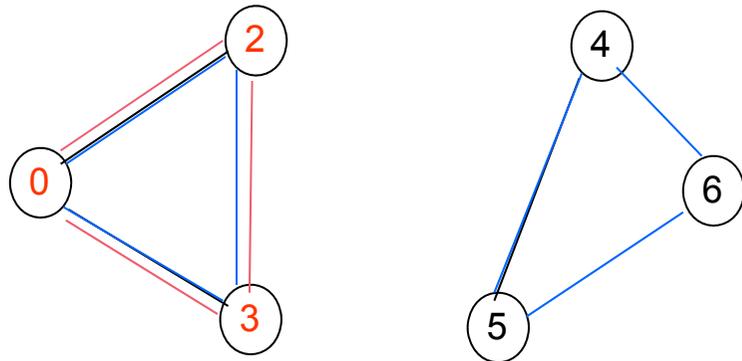
$$X_{ij}^k \in \{0,1\} \quad k=1,2, i, j \in N, i < j$$

$$Y_i^k \in \{0,1\} \quad k=1,2, i \in N$$

制約追加アプローチ：制約式が不完全

問題点：

流れの均衡式だけでは、サブツアーが発生する。



サブツアーの禁止制約：ノード0を含まない任意のNの部分集合 S に対して、

$$\sum_{ij:i \in S, j \in S, j > i} X_{ij}^k + \sum_{ji:i \in S, j \in S, j < i} X_{ji}^k \leq |S| - 1$$

制約が多すぎる！

初期設定：

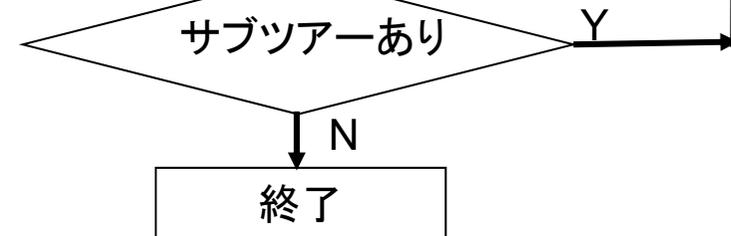
サブツアー禁止制約リスト: PS = \emptyset

制約緩和問題：

前頁のLPに、PSの制約を付加して解く

サブツアーのチェック：

サブツアーがあれば、禁止制約リストPSに追加



2日間集荷ルート決定問題: $N=21$ 、 $N1=10$ 、capacity:80

```
>Release¥Milk_Collection.exe 1
```

```
***** iter: 1
```

```
Optimal value: 12160
```

```
Day:0 size: 15
```

```
Base_Node:0 -> 7 -> 8 -> 9 -> 0
```

```
Base_Node:1 -> 4 -> 17 -> 1
```

```
Base_Node:2 -> 15 -> 12 -> 3 -> 18 -> 2
```

```
Base_Node:5 -> 6 -> 19 -> 5
```

```
Day:1 size: 16
```

```
Base_Node:0 -> 9 -> 5 -> 6 -> 10 -> 8 -> 20 -> 7 -> 11 -> 3 -> 14 -> 2 -> 13 -> 4 -> 1 -> 16 -
```

```
> 0
```

```
***** iter: 2
```

```
Optimal value: 12189
```

```
Day:0 size: 17
```

```
Base_Node:0 -> 1 -> 16 -> 5 -> 19 -> 6 -> 9 -> 0
```

```
Base_Node:2 -> 4 -> 13 -> 15 -> 12 -> 3 -> 18 -> 2
```

```
Base_Node:7 -> 8 -> 20 -> 7
```

```
Day:1 size: 14
```

```
Base_Node:0 -> 1 -> 17 -> 4 -> 2 -> 14 -> 3 -> 11 -> 7 -> 8 -> 10 -> 6 -> 5 -> 9 -> 0
```

```
***** iter: 3
```

```
Optimal value: 12293
```

```
Day:0 size: 15
```

```
Base_Node:0 -> 9 -> 5 -> 6 -> 8 -> 20 -> 7 -> 12 -> 3 -> 15 -> 18 -> 2 -> 4 -> 1 -> 16 -> 0
```

```
Day:1 size: 16
```

```
Base_Node:0 -> 1 -> 17 -> 4 -> 13 -> 2 -> 14 -> 3 -> 11 -> 7 -> 8 -> 10 -> 6 -> 19 -> 5 -> 9 -
```

```
> 0
```

解空間の拡張: Column Generationアプローチ:

Cutting Stock問題:

次の表の板を,幅Wの原板から製造したい。

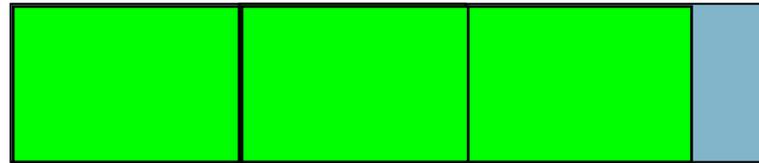
目的:原板の枚数を最小にする切り出しパターン?

幅	必要量
W1	d1
W2	d2
W3	d3
W4	d4

文献1)

切り出しパターン: $P = (P1, P2, P3, P4)$

$$P0 = (3, 0, 0, 0)$$



$$P1 = (2, 3, 0, 0)$$



有効なパターンをどのように生成するか?



Column Generationアプローチ: 列ベクトルが不完全

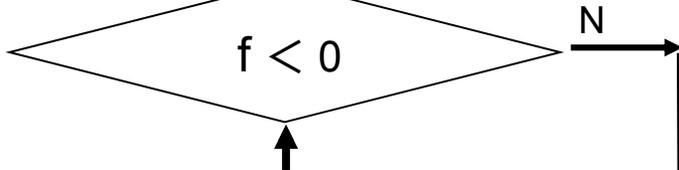
初期パターン群の生成: PS(0)
[同一サイズのみを切り出すパターン]
t=0

パターン y を列に追加する
 $PS(t+1) = PS(t) \cup \{y\}$
t=t+1

緩和親問題: cplex1
 $\sum Z_k \rightarrow \text{最小}$
sub to
cst[i]: $\sum_k P_{jk} Z_k \geq d_j \quad k \in PS(t)$
 $Z_k \geq 0$: 整数条件を緩和

Pricing 問題: cplex2
(列: パターンの生成)
o $f = 1 - \sum \pi_j y_j \rightarrow \text{最小}$
sub to
 $\sum W_j y_j \leq W$
 $y_j \geq 0$: 整数

Shadow priceの取得
 $\pi_i = \text{cplex1.getDual}(\text{cst}[i])$



未知のパターン(列ベクトル) y
で
Reduced Costが最小を求める

**親問題: 緩和なし
(不完全)**

文献5)

切り出し問題: Column Generation: ILOG Sample Program(cutstock.cpp)を実行

原板:110

幅	必要
20	48
45	35
50	24
55	10
75	8

幅	P0	幅	P2	幅	P4
20	5	20	0	20	0
45	0	45	0	45	0
50	0	50	2	50	0
55	0	55	0	55	0
75	0	75	0	75	1

幅	P5
20	1
45	2
50	0
55	0
75	0

幅	P6
20	1
45	0
50	0
55	0
75	1

幅	P7
20	3
45	0
50	1
55	0
75	0

幅	P1	幅	P3
20	0	20	0
45	2	45	0
50	0	50	0
55	0	55	2
75	0	75	0

初期パターン

パターン	最適製造量
P0	0
P1	0
P2	8
P3	5
P4	0
P5	18
P6	8
P7	8

親問題

非線形性の回避:プロ野球のCS クリンチ・ナンバー決定

プロ野球では、3位までのチームによるクライマックス・シリーズがあり、出場に必要な今後の勝利数、クリンチ・ナンバーが重要な指標である。

順位の決定法

レギュラーシーズンの順位は勝率を基準とする。

同一順位の解消法

勝率が並んだ場合は以下のように順位を決める。

セ・リーグ

勝利数が多い球団

直接対決で勝ち越している球団

前年度順位の上位球団

パ・リーグ

直接対決で勝ち越している球団

セ・パ交流線を除いたリーグ戦(全120試合)における勝率

前年度順位の上位球団

・この問題は、複数の球団の勝敗が入組んでおり、膨大な組合せが発生するが、チームの負け数をKeyにして、その時の最悪順位を混合整数計画法で求めることにより、クリンチ・ナンバーが計算できる。

チーム名	勝利数	負け数	引分数	残試合数	1位-確保	-可能	-可能	2位-確保	-可能(勝)	-可能	3位-確保	-可能(勝)	-可能(負)
広島	56	80	2	6	-1	-1	-1	-1	-1	-1	-1	-1	-1
中日	79	61	3	1	1	0	1	0	0	1	0	0	1
巨人	76	63	1	4	-1	-1	-1	-1	0	4	0	0	4
ヤクルト	68	66	4	6	-1	-1	-1	-1	-1	-1	-1	6	0
阪神	74	61	3	6	-1	6	0	5	1	5	1	0	6
横浜	47	91	1	5	-1	-1	-1	-1	-1	-1	-1	-1	-1

注1) 可能(勝):これより勝数が少ないと、この順位は不可能である。
 注2) 可能(負):これより負け数が多いと、この順位は不可能である。

非線形問題を回避した緩和問題の導入

この問題は、最初の決定因子の勝率が、全試合を集計したとき、引分けが除外されるため、

$$A\text{チームAの勝率} = A\text{チームの勝数} / \{A\text{チームの勝数} + A\text{チームの負数}\}$$

AチームがBチームより勝率が良い:

$$A\text{チームの勝数} / \{A\text{チームの勝数} + A\text{チームの負数}\}$$

$$> B\text{チームの勝数} / \{B\text{チームの勝数} + B\text{チームの負数}\}$$

→ **非線形問題**となる。

Aチームと他チームとの順位を決める場合に、Aチームの(勝数、負数、引分数)をパラメータにして計算すれば、Aチームが順位上:

$$A\text{チームの勝数(定数)} \times B\text{チームの負数} > A\text{チームの負数(定数)} \times B\text{チームの勝数}$$

→ 線形不等式となり、混合整数計画の**緩和問題**が得られ、**確保順位**、**可能順位**が求まる。

$$A\text{勝数(定)} * B1\text{負数} \leq A\text{負数(定)} * B1\text{勝数} + M * \lambda 1$$

$$A\text{勝数(定)} * B2\text{負数} \leq A\text{負数(定)} * B2\text{勝数} + M * \lambda 2$$

:

$$A\text{勝数(定)} * B5\text{負数} \leq A\text{負数(定)} * B5\text{勝数} + M * \lambda 5$$

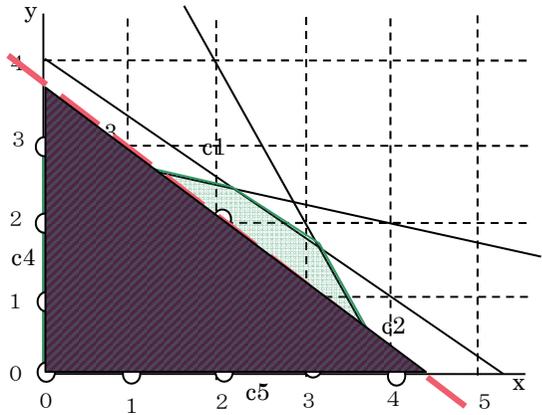
$$A\text{チーム確保順位} \doteq 6 - \text{Min} \{ \lambda 1 + \lambda 2 + \dots + \lambda 5 \}$$

(要:同率の場合の付加ルールに対する処理を追加)

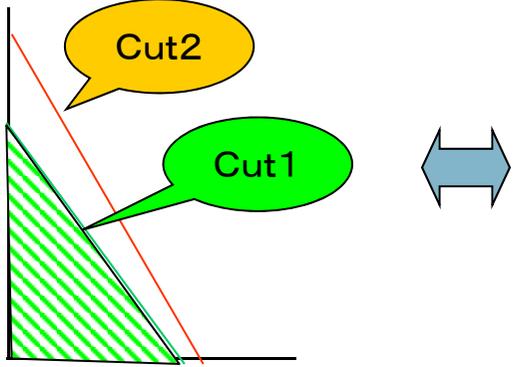
Aチームのクリンチナンバー: 3位以内を確保するケースで、引分なしの**最小勝数**がクリンチナンバー

動的な解空間の縮小: Valid inequality: Cut

緩和問題の領域を削るが、許容整数解は削除しない制約



Cover Cut: x_1, x_2, x_3 は、0-1変数
 制約条件: $4x_1 + 3x_2 + 2x_3 \leq 5$
 Minimal cover: 変数の値を1にした時に、
 制約条件が無効となる変数群の極小集合
 $S_1 = \{x_1, x_2\}$ $S_2 = \{x_1, x_3\}$
 Cover cut: $x_1 + x_2 \leq 1$
 $x_1 + x_3 \leq 1$



Cutの優劣:
 2個の $cut1 (\sum a_i x_i \leq b)$ と $cut2 (\sum c_i x_i \leq d)$ に対して、
 $cut1$ が $cut2$ より優勢である

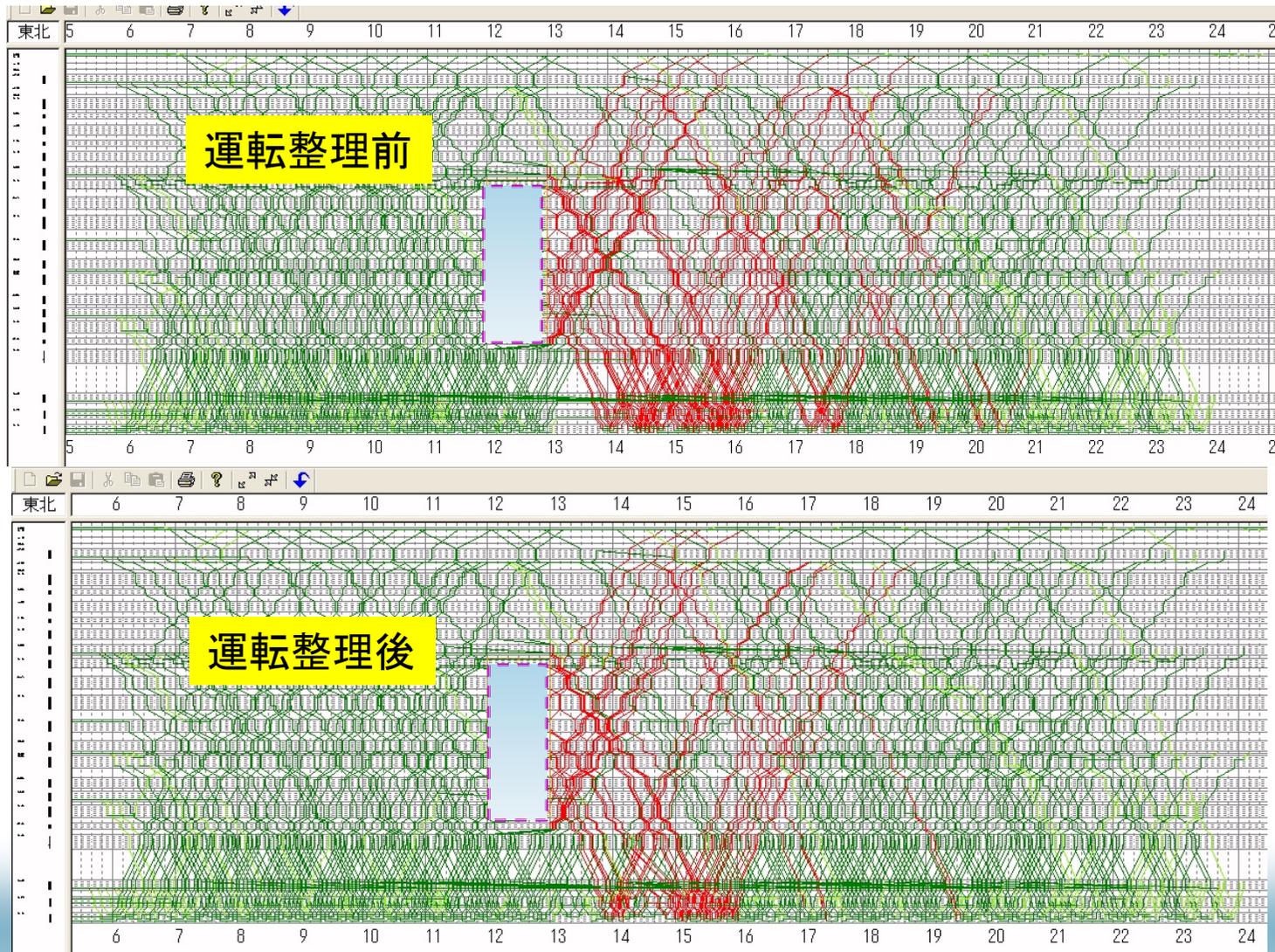
或る $z > 0$ が存在して、 $a_i \geq z c_i$ 、 $b \leq z d$ とできる。

文献3)

優勢なcutのみを使用する

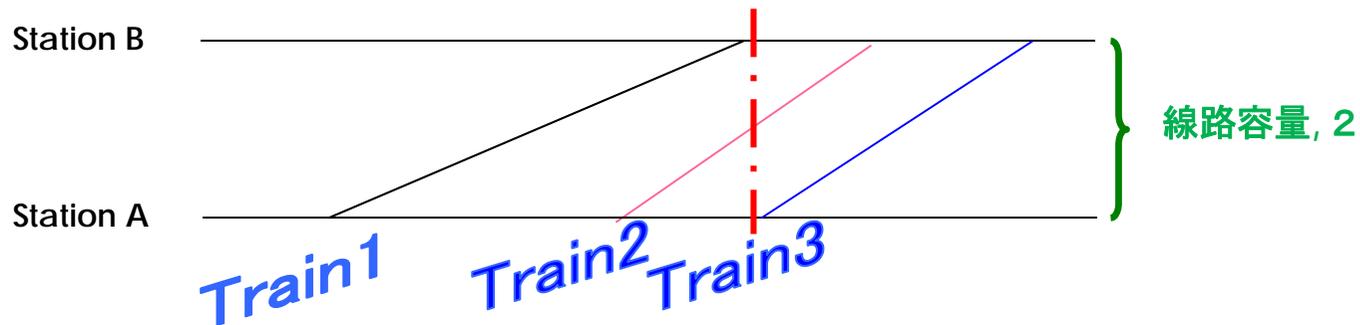
制約プログラミングによる動的な解空間の縮小: 新幹線運転整理システム

(小山~仙台間が12:00~13:00まで通行不可)



駅間の線路容量と制約伝搬

- 駅間には、同時に k 本以下の列車が走行可能



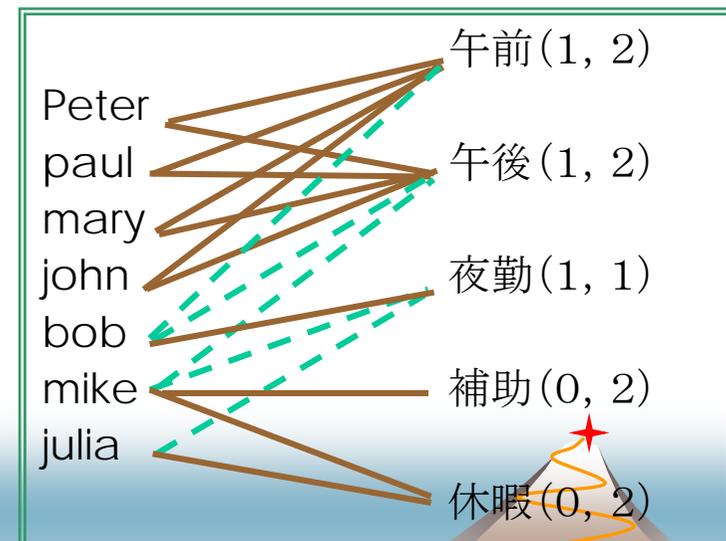
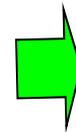
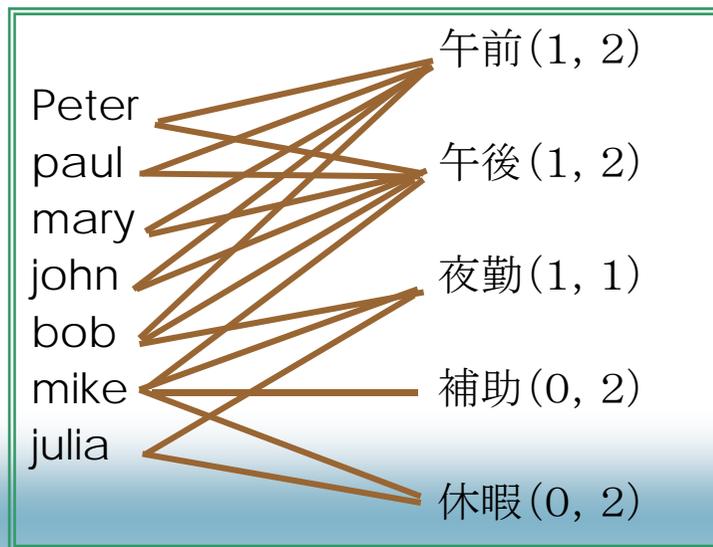
制約伝搬: $\text{Train1_Arrive_StB} \leq \text{Train3_Depart_STA}$

Train1が出発可能なためには
→ B駅に着くまでに、B駅の番線が空くこと
Train2が出発可能なためには、
→ Train1がB駅に到着していること
未来が決まらないと、現在の判断が出来ない
⇒シミュレーションは不可能

制約(論理)プログラミングだけでは、**解けない**

- 制約伝播の不完全性
 - 制約(論理)プログラミングの固有の制約伝播だけでは、複雑な制約条件に対して、完全な制約伝播を行うことは、計算時間上、不可能である。
- 大域的な制約条件の導入
 - オペレーションズ・リサーチ等の理論を応用することにより、効率良く制約伝播が可能な汎用の**大域的制約条件**の導入 → **メーカーの差異化**
- 例:勤務計画で、勤務毎の人数の制約: 勤務(最小、最大)

最大フロー問題を利用した制約伝播: J. Regin (ILOG)



Math-Model Research Inc.

計画問題へのアプローチ（開発手法）

- 計画問題に対する2つのアプローチ
 - 手続的なアプローチ(従来のプログラム)
 - 宣言的なアプローチ(制約の定義と解探索の分離)

百足もいるよ
!

つる・かめ算の計算

鶴と亀が足して5匹、足の合計が14本の時、鶴と亀の数は？

手続的アプローチ

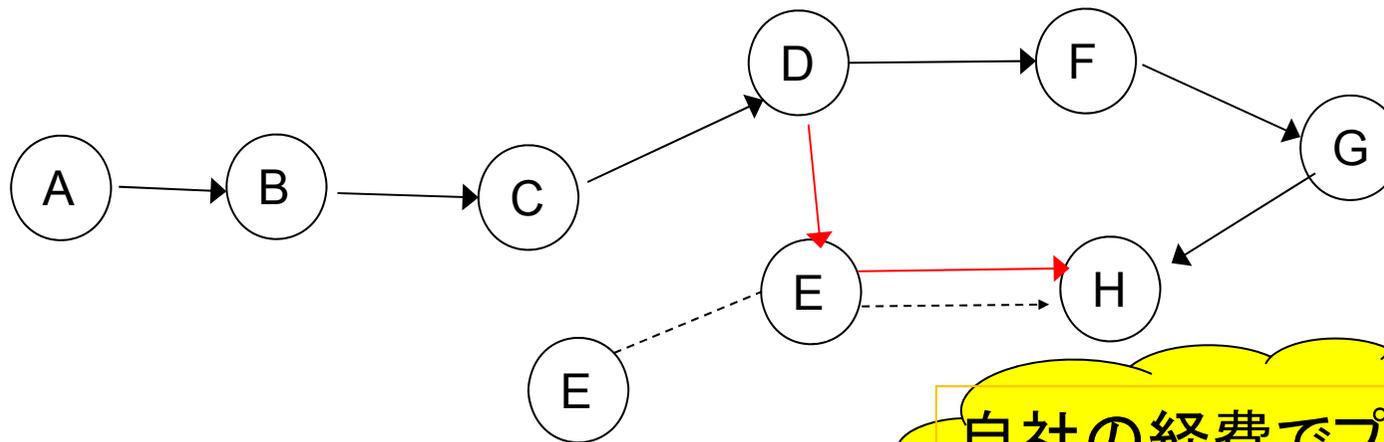
- 1) 全部鶴と仮定すると、足は10本
- 2) 余った足の数 $14 - 10 = 4$ 本
- 3) 亀の数 = $4本 / 2 = 2$ 匹
- 4) 鶴の数 = $5 - 2 = 3$ 匹

宣言的アプローチ

- 1) 鶴と亀の数を X 、 Y とする
- 2) $X + Y = 5$ $2X + 4Y = 14$ (定義)
- 3) 連立方程式を解く (探索)
- 4) $X = 3$ $Y = 2$

宣言型の定期運賃計算システムの開発

- 従来の方法:
- 1) ベテランのノウハウで、与えられた発駅と着駅に対して、運賃計算用のルートを作成する。
- 2) ベテランのノウハウで、このルートに適用する割引規則群を選択して、最終運賃を決定する。



- 当初ルート: A->B->C->D->F->G->H
- **D->E**開通後: A->B->C->**D->E->H**

他社の新駅設置、運賃規則変更

自社の経費でプログラム改修

宣言型定期運賃計算システム

	A	B	C	D	E	F	G	H	I	J	K	
2	\$会社リスト											
3	会社名	会社コード	略号	分類コード								
4	阪神	147	阪神	4						最新情報抽出		
5	JR西日本	5	JR西日本	1								
6	大阪市営	141	大阪市営	3								
7	神戸高速	146	神戸高速	3								
8	神鉄	156	神鉄	4						路線情報抽出		
9	阪急	148	阪急	4								
10	山陽	157	山陽	4								
11	近鉄	161	近鉄	4								
12	京阪	155	京阪	4								
13	南海	151	南海	4								
14	神戸市交	142	神戸市交	3						クライアント情報生成		
15	神戸新交通	144	神戸新交通	4								
16	阪神バス	201	阪神バス	14								
17	阪急バス	202	阪急バス	14								
18	尼崎市バス	203	尼崎市バス	13								
19	神戸市バス	204	神戸市バス	13						クライアント・テスト		
20	\$END											
21	# 鉄道の分類コードは、9以下、バス会社は11以上とする。											
22												

\$営業制度	2										
\$阪神-鉄道	1	単純計算方式	乗車経路	切上		10					
運賃定義名	優先順序	専属適用線区定	制限条件	パラメータ1	パラメータ2	パラメータ3	換算係数	運賃計算			
阪神全線	1	あり	なし	なし	なし	なし		1	営業キロ		
#											
通用期間	3										
期間定義名	処理種別	対象データ	運賃比率	加算修正額	端数処理	単位					
1か月	データ										
3か月	計算	1か月	2.85	0	切上		10				
6か月	計算	1か月	5.4	0	切上		10				
#											
計算運賃	= (対象データ*運賃比率 + 加算修正額) %端数処理 (単位)										
#											
発売種別	4										
券種/レベル	大小/区分数	基本(通勤/通学)	基本(大人/小児/小中)	高校	実習	介護	障害				
標準レベル	7	1	2	3	3	3	3				
通勤	大人	◎	◎	×	×	×	介護A	障害			
通勤	小児	◎	小児	×	×	×	介護B	障害			
通学	大人	◎	◎	小中	高校	実習	×	障害			
通学	小児	◎	小児	小中	×	×	×	障害			
#											
基本データ	2										
略称	期間	基本(通勤/通学)	基本(大人/小児/小中)	高校	実習	介護	障害				
通勤・大人・1	1か月	通勤	大人	なし	なし	なし	なし				
通学・大人・1	1か月	通学	大人	なし	なし	なし	なし				
#											
追加割引区分	8										
追加割引定義名	レベル	会社制度	処理種別	対象データ	運賃比率	加算修正額	端数処理	単位			
小児	2	定義	計算	大人	0.5	0	切上				
小中	3	仮想	なし	*	*	*	*	*			
高校	3	仮想	なし	*	*	*	*	*			
実習	3	定義	なし	*	*	*	*	*			
介護A	3	定義	計算	上位	0.5	0	切上				
介護B	3	定義	なし	*	*	*	*	*			
障害A	3	定義	計算	上位	0.5	0	切上				
障害B	3	定義	なし	*	*	*	*	*			



会社間割引規則の適用

#	野田阿波中央線 - 2	鉄道発売	発着	鉄道連絡	連絡発売	なし	◎大物	野	2	阪神	大阪市営	
		地域定義	5									
		開始地域	地域 A	1	区間 A							
		終了地域	地域 B	1	区間 B							
		経由地域	地域 C	1	区間 C							
		禁止地域	地域 D	2	区間 D	区間 E						
		内部地域	地域 E	2	区間 F	区間 G						
		区間定義	7									
		駅間指定	区間 A	阪神	阪神西大阪線	出来島	西九条	*	*	*	内	
		駅間指定	区間 B	大阪市営	大阪市営中央線	阿波座	阿波座	*	*	*	除	
		駅間指定	区間 C	大阪市営	大阪市営千日前	野田阪神	阿波座	*	*	*	内	
		駅間指定	区間 D	大阪市営	大阪市営千日前	阿波座	西長堀	*	*	*	内	
		駅間指定	区間 E	阪神	阪神西大阪線	尼崎	出来島	*	*	*	内	
		駅間指定	区間 F	大阪市営	大阪市営中央線	阿波座	阿波座	*	*	*	除	
		駅間指定	区間 G	大阪市営	大阪市営千日前	野田阪神	阿波座	*	*	*	内	
		接続定義	1									
		駅間接続	接続 A	対称	阪神	阪神本線	野田	大阪市営	大阪市営千	野田阪神	0	*
		券種定義	1		券種	大人・小児	高校・小中	介護・障害	なし			
		券種指定	券種 A	適用	*	*	*	*				
		期間定義	1									
		期間指定	期間 A		3	1か月	3か月	6か月				

#	社間乗継											
	ルール 1 1	乗継 2 個	発着	乗継割引	割引個別	*	*		2	阪神	神戸高速	
		地域定義	3									
		開始地域	地域 A	1	区間 A							
		終了地域	地域 B	2	区間 B	区間 C						
		経由地域	地域 C	2	区間 B	区間 D						
		区間定義	4									
		駅間指定	区間 A	阪神	阪神本線	芦屋	春日野道	*	*	*	内	
		駅間指定	区間 B	神戸高速	神戸高速東西線	高速神戸	西代	*	*	*	内	
		駅間指定	区間 C	神戸高速	神戸高速東西線	元町	高速神戸	*	*	*	内	
		駅間指定	区間 D	神戸高速	神戸高速東西線	元町	高速神戸	*	*	*	内	
		接続定義	1									
		駅間接続	接続 A	対称	阪神	阪神本線	元町	神戸高速	神戸高速東	元町	0	*
		積算定義	1									
		換算距離	距離 A	1	地域 C	1.0						
		券種定義	1		券種	大人・小児	高校・小中	介護・障害	なし			
		券種指定	券種 A	適用	*	*	*	*				
		期間定義	1									
		期間指定	期間 A		3	1か月	3か月	6か月				
		割引定義	12									
		割引個別	阪神	通勤・大人・1	均一方式	-	1	0				
		割引個別	阪神	通勤・大人・3	均一方式	-	1	0				
		割引個別	神戸高速	通学・大人・3	距離方式	距離 A	2	650	1	3	650	
		割引個別	神戸高速	通学・大人・6	距離方式	距離 A	2	1240	1	3	1240	
		排他定義	2									
		会社排他	阪神	*	全体							
		会社排他	神戸高速	*	全体							
		配分定義	1									
		表績配分										

宣言型システムの開発プロセス

- 1) 対象ネットワークのEXCEL表現とネットワーク生成
 - ・各会社の路線情報
 - ・会社間の駅接続情報
- 2) 鉄道営業規則のEXCEL上の表現形式の考案
 - ・各会社内の運賃規則の表現
 - ・連絡発売規則の表現
 - ・連絡運賃割引規則の表現
 - ・サイバネ規則の表現
- 3) EXCELで表現された規則のネットワーク上の処理方式
 - ・ルートをトラバースして、集計情報を作成し、適用可能ルールを判断
- 4) 発売規則に合致した複数ルートの生成
 - ・下限値を利用した経路生成と発売規則のチェック
- 5) ルートに適用可能な割引規則の抽出
- 6) 最大割引可能な規則群の抽出(最大独立集合)
- 7) ルートの最安運賃の決定
- 8) 対象ルート群の最安運賃の決定

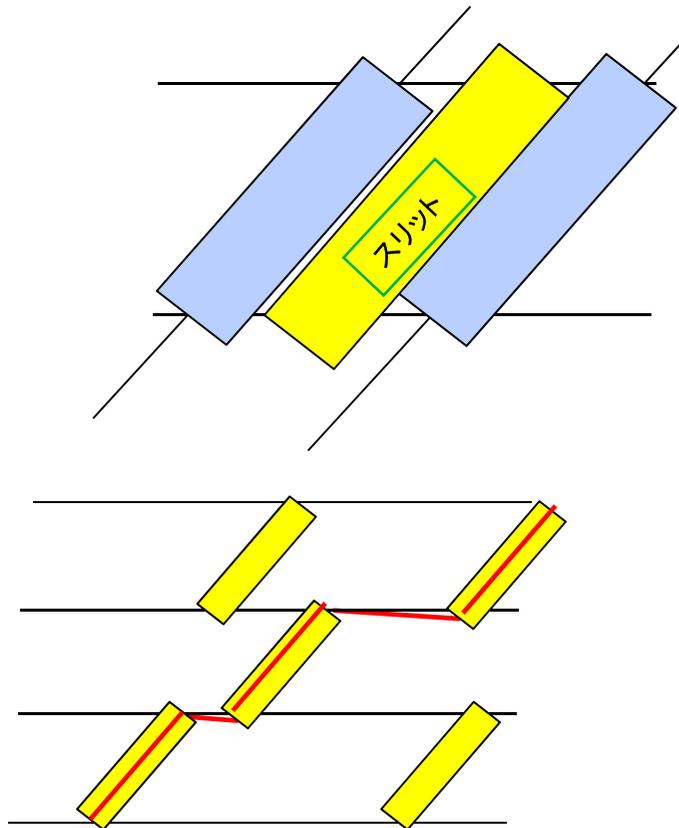
開発事例



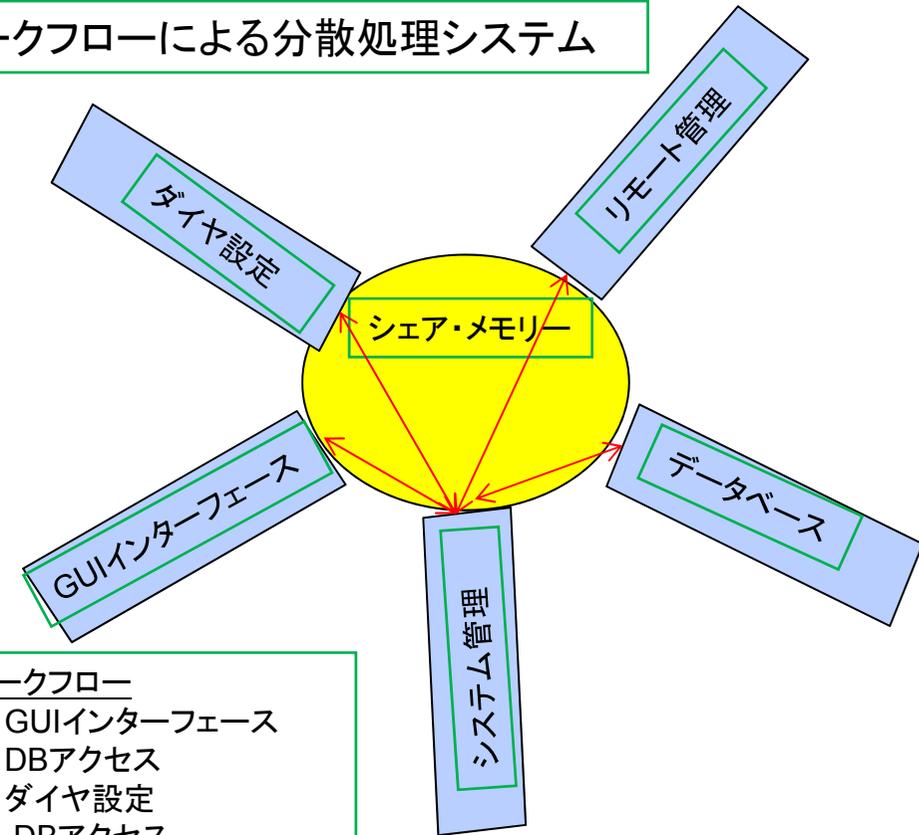
Math-Model Research Inc.

分散処理アーキテクチャ：制約ベースの列車ダイヤ作成システム

制約ベースの列車ダイヤ設定とワークフローによる分散処理システム



列車設定：始発駅から終着駅迄のスリットのルートを見つけるアルゴリズム



ワークフロー

1. GUIインターフェース
2. DBアクセス
3. ダイヤ設定
4. DBアクセス
5. リモート管理
6. GUIインターフェース

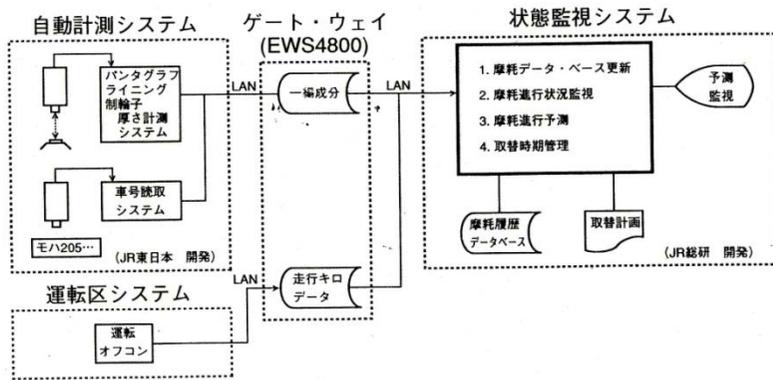
ワークフロー：**改良が容易**

分散処理：サブシステムのバグが局所かされ、更新も容易に行える。

Math-Model Research Inc.

分散処理アーキテクチャ: 車両消耗品管理システム

パンタ・すり板、制輪子、ブレーキ・ライニングの遠隔測定データ履歴より、取替計画を作成



取替え支援システムの全体構成

山15 編成 取替計画

車号	部品	位置	残km	厚さ	摩耗率	推定日	計画
モハ204045	制輪子	5	0km	18.6mm	6.482	12/31	12/30
モハ204044	制輪子	1	781km	20.3mm	3.613	01/01	12/30
モハ205015	制輪子	1	5758km	21.4mm	2.410	01/13	12/30
モハ204043	制輪子	2	7449km	23.4mm	4.556	01/16	12/30
モハ205015	ライニング	1	7822km	12.0mm	2.540	01/17	12/30
モハ205029	ライニング	7	9276km	13.3mm	3.584	01/21	12/30
モハ205015	ライニング	4	11018km	12.2mm	1.975	01/25	12/30
モハ205029	ライニング	3	11196km	14.6mm	4.141	01/25	12/30
モハ205015	ライニング	1	13882km	12.9mm	2.086	01/31	12/30
モハ205015	ライニング	5	14505km	14.9mm	3.406	02/01	>12/30
モハ205030	制輪子	3	16007km	22.8mm	1.776	02/05	
モハ205015	ライニング	5	18121km	14.8mm	2.663	02/10	
モハ205015	制輪子	2	19828km	23.1mm	1.566	02/14	
モハ204044	制輪子	2	23479km	27.7mm	3.288	02/22	
モハ205045	制輪子	5	25632km	34.7mm	5.733	02/27	
モハ204043	制輪子	1	25848km	29.7mm	3.741	02/27	
モハ205044	パンタ	1	28156km	3.8mm	0.296	03/04	
モハ204015	ライニング	2	29429km	18.5mm	2.887	03/07	

計画を入力 順序[キロ]/短縮[既計画外]
 表示順序 短縮表示 拡大表示 複写 取消 取替済 終了

図 1. 3 編成別の取替計画作成

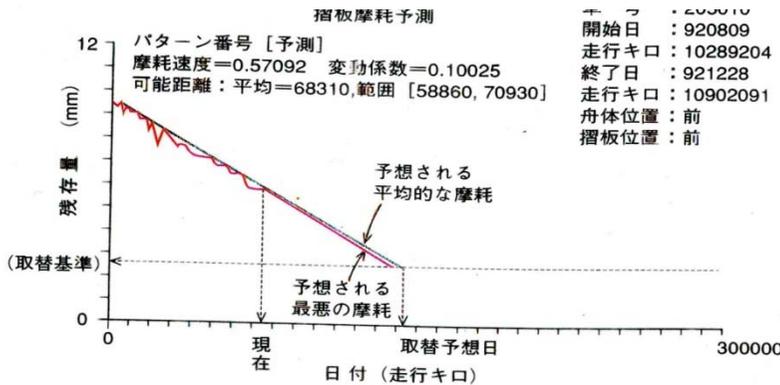


図 5 すり板摩耗予測

- ・分散処理方式で、独立したシステム開発が可能
- ・取替日から測定日までの日数の逆数の重みを付けた回帰分析で、直近の摩耗速度を重視する

AIとORの結合アーキテクチャ: 列車乗継案内システム

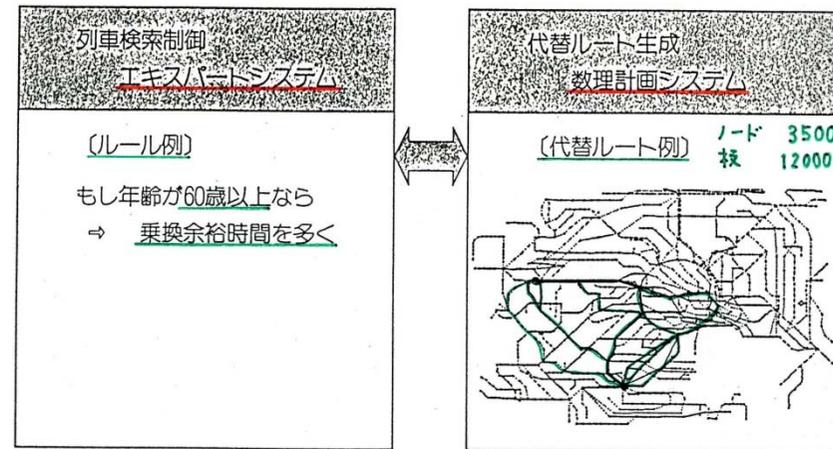
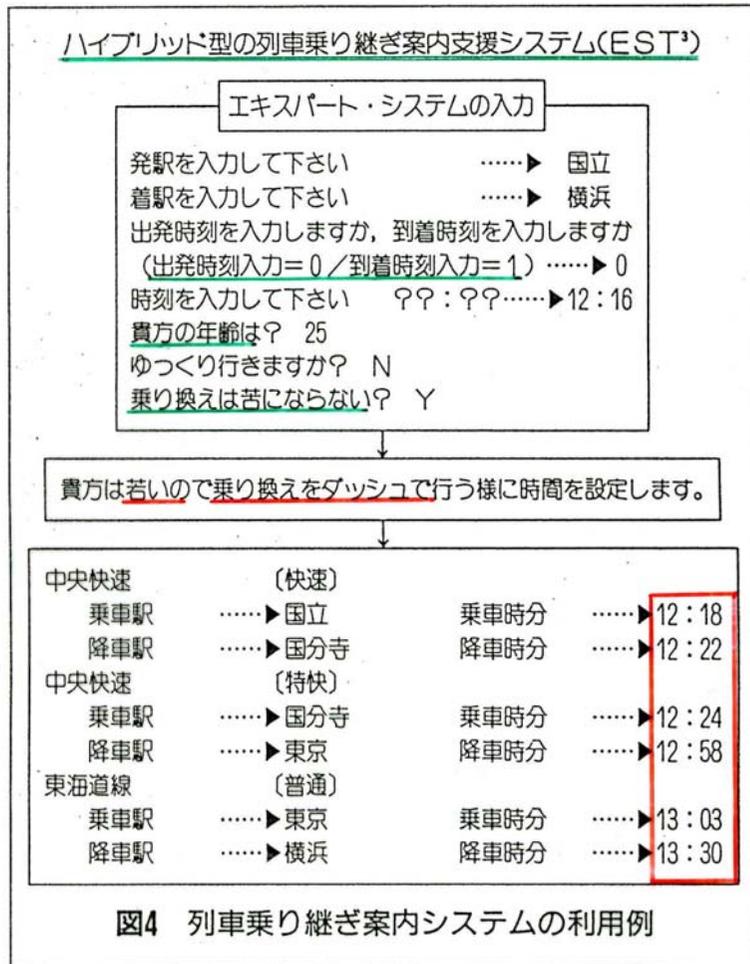


図1 ハイブリッド型のエキスパート・システム
(列車乗り継ぎ案内システムEST')

表1 複数ルートの例
(中央線国立駅～東海道線横浜駅)

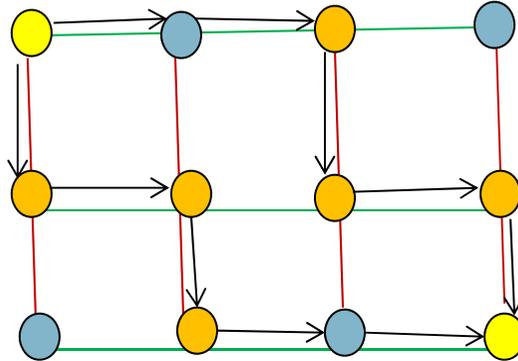
(1)	国立—(中央線)—東京—(東海道線)—横浜
(2)	国立—(中央線)—東京—(総武快速)—横浜
(3)	国立—(中央線)—新宿—(山手線)—品川—(東海道線)—横浜
(4)	国立—(中央線)—新宿—(山手線)—品川—(総武快速)—横浜
(5)	国立—(中央線)—新宿—(山手線)—品川—(京浜急行)—横浜
(6)	国立—(中央線)—新宿—(山手線)—渋谷—(東急)—横浜
(7)	国立—(中央線)—西国分寺—(武蔵野線)—府中本町—(南武線)—武蔵小杉—(東急)—横浜
(8)	国立—(中央線)—立川—(南武線)—武蔵小杉—(東急)—横浜
(9)	国立—(中央線)—八王子—(横浜線)—東神奈川—(根岸線)—横浜

鉄道乗車券・券面経路情報作成システム＋運賃計算方式の提案(1992)

発・着駅間の複数経路の生成と、その経路に対する乗車券の経路駅情報を生成するシステム

鉄道路線ネットワークをベースとした運賃計算方式の試案作成

運賃規則に適合した経路の生成



運賃割引規則の適用判断

- ・運賃規則のネットワーク表現
- ・ネットワークのトラバースによる適用可能規則の抽出
- ・割引間の排他制約の処理方式

$\text{Min} \{ \text{指定された中継ノードを経由するルート} \text{の距離} \}$
 \geq
発から中継ノードへの最短距離
+ 着から中継ノードへの最短距離

・運賃計算システム分野への進出が可能！！

- ・最短経路情報をベースとした実用可能性のある複数経路の生成技術
- ・非構造的な問題へのアプローチ法

Math-Model Research Inc.

メッシュデータと地理情報処理アーキテクチャ: TRAMPS

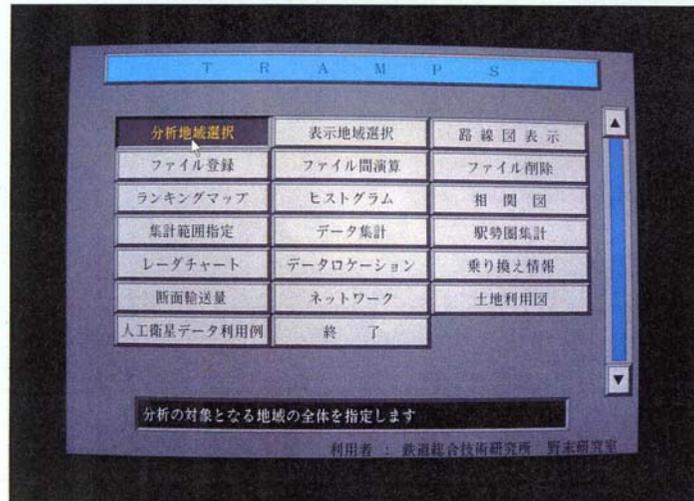
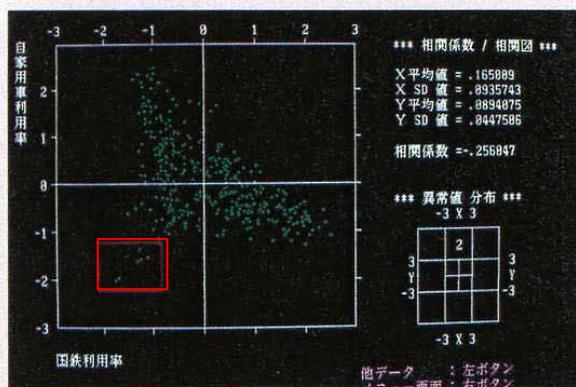
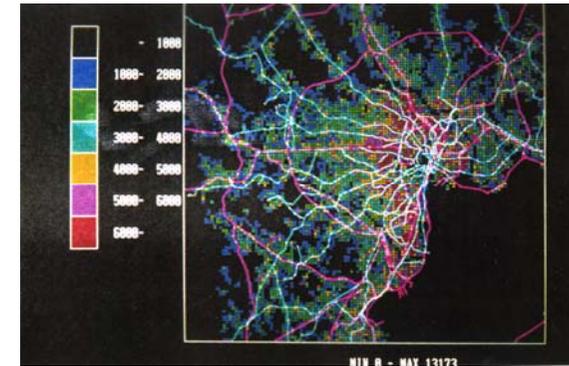


図8 分析データの階層

項目	階層	説明
国鉄人口	1階	国鉄人口
国鉄人口	2階	国鉄人口
国鉄人口	3階	国鉄人口
国鉄人口	4階	国鉄人口
国鉄人口	5階	国鉄人口
国鉄人口	6階	国鉄人口
国鉄人口	7階	国鉄人口
国鉄人口	8階	国鉄人口
国鉄人口	9階	国鉄人口
国鉄人口	10階	国鉄人口
国鉄人口	11階	国鉄人口
国鉄人口	12階	国鉄人口
国鉄人口	13階	国鉄人口
国鉄人口	14階	国鉄人口
国鉄人口	15階	国鉄人口
国鉄人口	16階	国鉄人口
国鉄人口	17階	国鉄人口
国鉄人口	18階	国鉄人口
国鉄人口	19階	国鉄人口
国鉄人口	20階	国鉄人口
国鉄人口	21階	国鉄人口
国鉄人口	22階	国鉄人口
国鉄人口	23階	国鉄人口
国鉄人口	24階	国鉄人口
国鉄人口	25階	国鉄人口
国鉄人口	26階	国鉄人口
国鉄人口	27階	国鉄人口
国鉄人口	28階	国鉄人口
国鉄人口	29階	国鉄人口
国鉄人口	30階	国鉄人口
国鉄人口	31階	国鉄人口
国鉄人口	32階	国鉄人口
国鉄人口	33階	国鉄人口
国鉄人口	34階	国鉄人口
国鉄人口	35階	国鉄人口
国鉄人口	36階	国鉄人口
国鉄人口	37階	国鉄人口
国鉄人口	38階	国鉄人口
国鉄人口	39階	国鉄人口
国鉄人口	40階	国鉄人口
国鉄人口	41階	国鉄人口
国鉄人口	42階	国鉄人口
国鉄人口	43階	国鉄人口
国鉄人口	44階	国鉄人口
国鉄人口	45階	国鉄人口
国鉄人口	46階	国鉄人口
国鉄人口	47階	国鉄人口
国鉄人口	48階	国鉄人口
国鉄人口	49階	国鉄人口
国鉄人口	50階	国鉄人口
国鉄人口	51階	国鉄人口
国鉄人口	52階	国鉄人口
国鉄人口	53階	国鉄人口
国鉄人口	54階	国鉄人口
国鉄人口	55階	国鉄人口
国鉄人口	56階	国鉄人口
国鉄人口	57階	国鉄人口
国鉄人口	58階	国鉄人口
国鉄人口	59階	国鉄人口
国鉄人口	60階	国鉄人口
国鉄人口	61階	国鉄人口
国鉄人口	62階	国鉄人口
国鉄人口	63階	国鉄人口
国鉄人口	64階	国鉄人口
国鉄人口	65階	国鉄人口
国鉄人口	66階	国鉄人口
国鉄人口	67階	国鉄人口
国鉄人口	68階	国鉄人口
国鉄人口	69階	国鉄人口
国鉄人口	70階	国鉄人口
国鉄人口	71階	国鉄人口
国鉄人口	72階	国鉄人口
国鉄人口	73階	国鉄人口
国鉄人口	74階	国鉄人口
国鉄人口	75階	国鉄人口
国鉄人口	76階	国鉄人口
国鉄人口	77階	国鉄人口
国鉄人口	78階	国鉄人口
国鉄人口	79階	国鉄人口
国鉄人口	80階	国鉄人口
国鉄人口	81階	国鉄人口
国鉄人口	82階	国鉄人口
国鉄人口	83階	国鉄人口
国鉄人口	84階	国鉄人口
国鉄人口	85階	国鉄人口
国鉄人口	86階	国鉄人口
国鉄人口	87階	国鉄人口
国鉄人口	88階	国鉄人口
国鉄人口	89階	国鉄人口
国鉄人口	90階	国鉄人口
国鉄人口	91階	国鉄人口
国鉄人口	92階	国鉄人口
国鉄人口	93階	国鉄人口
国鉄人口	94階	国鉄人口
国鉄人口	95階	国鉄人口
国鉄人口	96階	国鉄人口
国鉄人口	97階	国鉄人口
国鉄人口	98階	国鉄人口
国鉄人口	99階	国鉄人口
国鉄人口	100階	国鉄人口



G-8 国鉄利用率と
自家用車利用率の
相関図

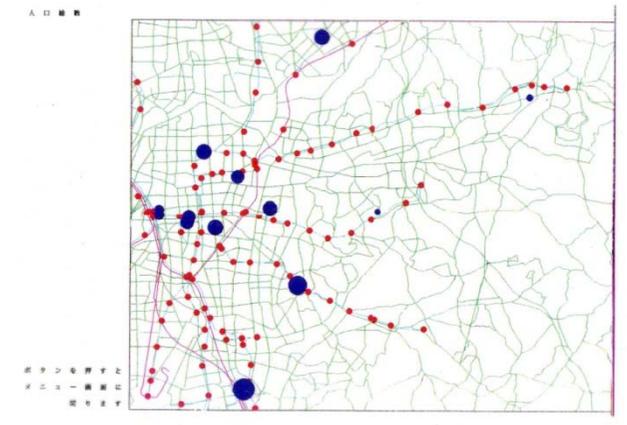
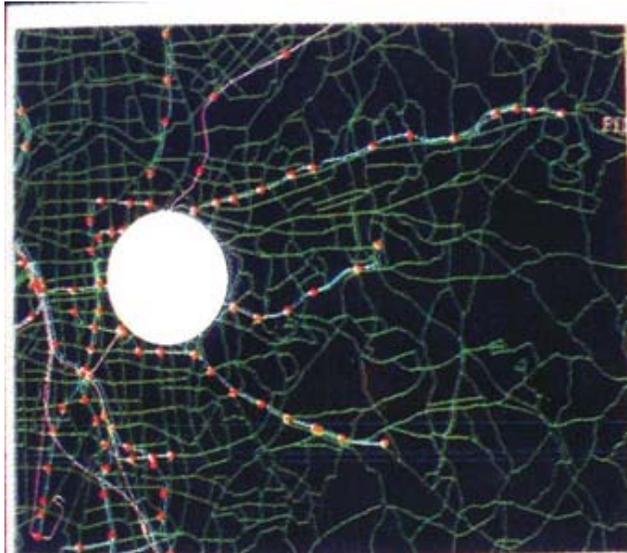
(データロケーション)



G-9
異常データの
地図上位置表示

商圈分析モデル

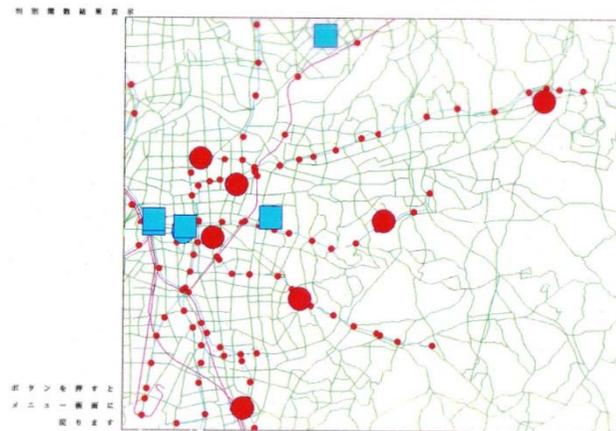
地理情報処理システム (TRAMPS) を用いて、店舗の営業成績を分析する



判別関数

判別テスト

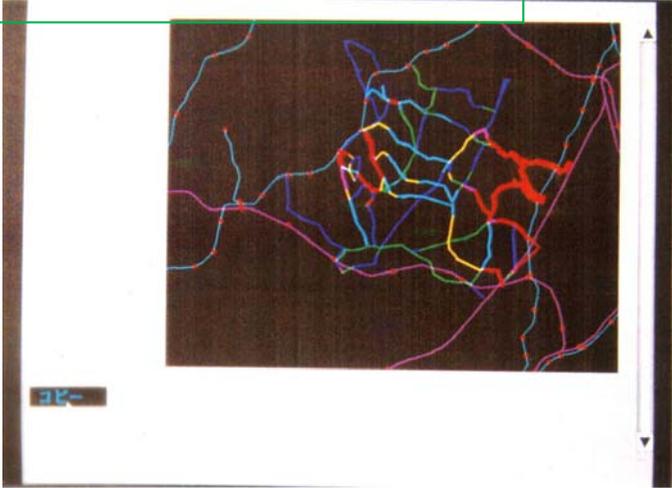
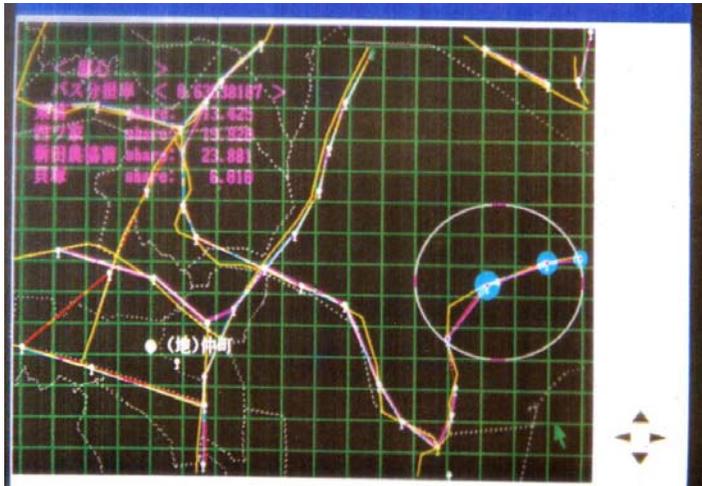
サンプル (1) の判別	< 1 >	-0.1159200	>>	1
サンプル (2) の判別	< 1 >	-0.1884510	>>	1
サンプル (3) の判別	< 1 >	-0.2147150	>>	1
サンプル (4) の判別	< 1 >	0.8427701	>>	2
サンプル (5) の判別	< 1 >	-0.2174260	>>	1
サンプル (6) の判別	< 1 >	-0.2049390	>>	1
サンプル (7) の判別	< 1 >	-0.0581654	>>	1
サンプル (8) の判別	< 1 >	-0.1535390	>>	1
サンプル (9) の判別	< 1 >	-0.1791640	>>	1
サンプル (10) の判別	< 2 >	0.0670372	>>	2
サンプル (11) の判別	< 2 >	0.2642040	>>	2
サンプル (12) の判別	< 2 >	0.2659370	>>	2
サンプル (13) の判別	< 2 >	0.4135690	>>	2
サンプル (14) の判別	< 2 >	0.2788000	>>	2



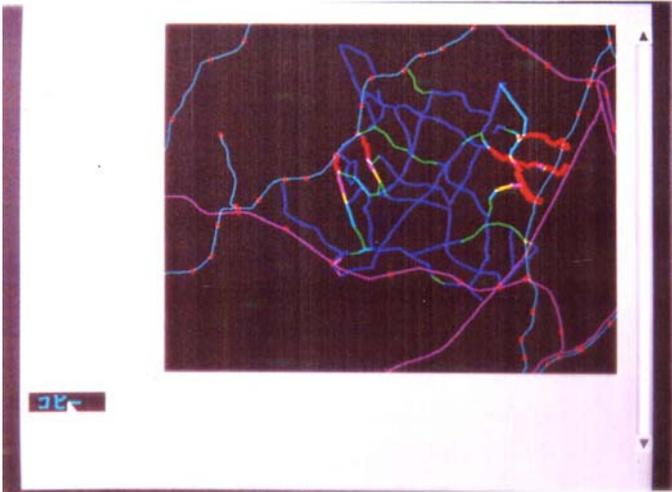
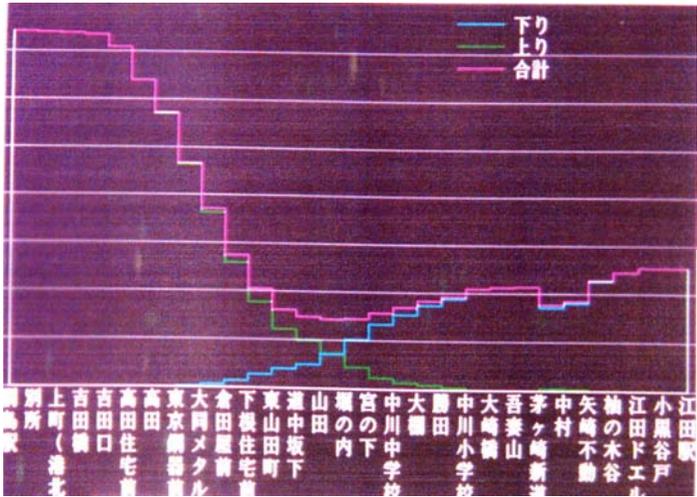
・商圈毎に集計した200種のデータに基づき判別分析を行った結果、非常に明快な結果が得られた。

Math-Model Research Inc.

地理情報処理 + 確率効用理論アーキテクチャ: バス輸送量推定



開 通 前



開 通 後

Key point: 確率効用モデル、地理情報処理、メッシュ・データ生成

リモート・センシング・データを用いた3D鉄道路線計画システム

人工衛星から観測された地表カラー情報、標高情報を利用した3Dの鉄道路線計画システム

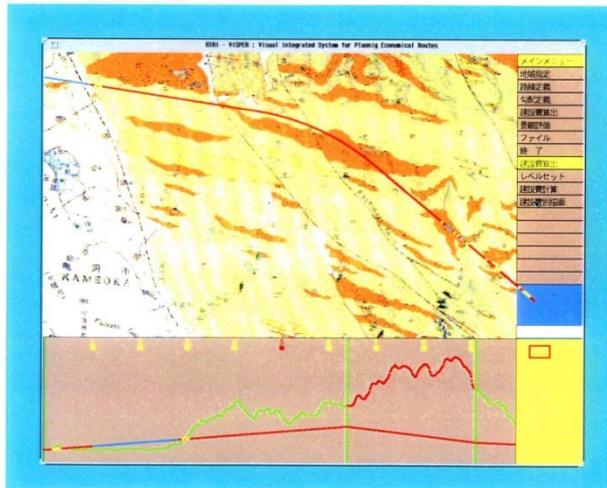
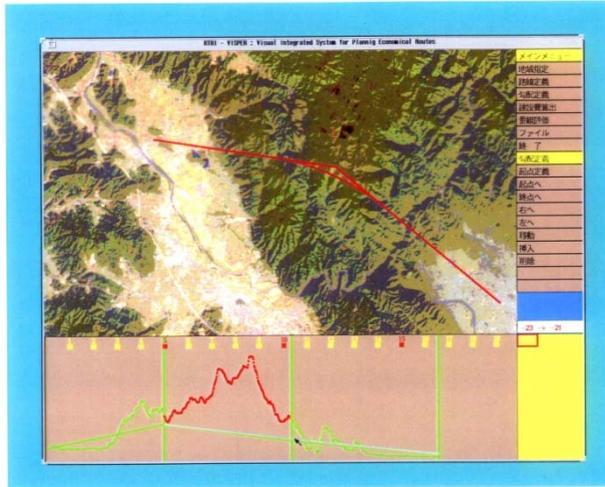


図 地質図表示

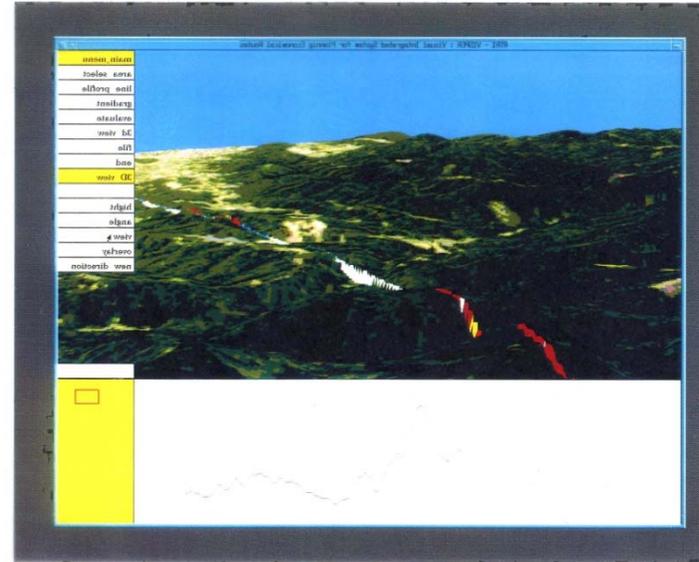


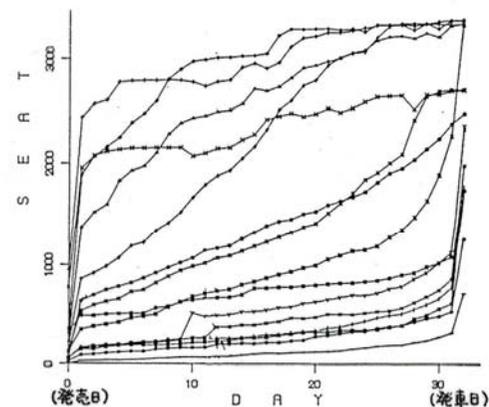
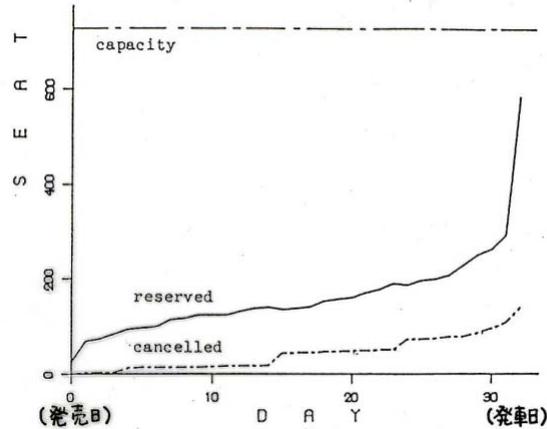
図 断面点を含むイメージ表示

- ・視点と対象メッシュを結ぶ線と眼前のスクリーンとの交点を計算し、遠方のメッシュから描画すれば良い。
- ・騒音、景観など対象物の視認性の問題に応用可能
- ・ゴルフ場等の障害物や断層等の状況を視認できる。

Math-Model Research Inc.

時系列処理:MARSの座席予約情報を利用した動的列車運行計画

現在5本運行されている「ひかり」の内、3本を事前に公表し、1週間の予約実績から、追加する列車を動的に設定する運行方式である。



時間帯別予約プロセス (混雑日)

予約情報を利用した動的列車設定

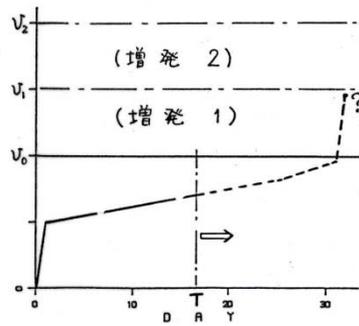


Table II

Summary of Train Schedule:light

T	P	Original Plan		New Plan	
		NT	LF	NT	LF
6	777	5	0.21	73	0.36
7	1085	5	0.30	73	0.50
8	2017	5	0.56	74	0.70
9	1708	4	0.59	4	0.59
10	1981	5	0.55	4	0.68
11	1426	4	0.49	73	0.66
12	1724	4	0.60	73	0.79
13	1943	5	0.54	74	0.67
14	1870	4	0.65	73	0.86
15	1963	4	0.68	73	0.91
16	1919	4	0.66	73	0.88
17	2636	5	0.73	5	0.73
18	2587	4	0.89	4	0.89
19	1362	3	0.63	3	0.63
20	781	2	0.54	2	0.54
Total	25779	63	0.57	51	0.70

-12

T Time
P Number of passengers actually passing through the middle point between Tokyo and Osaka
NT Number of trains operated in this time period
LF Load Factor; Capacity of each train is 723

Table I

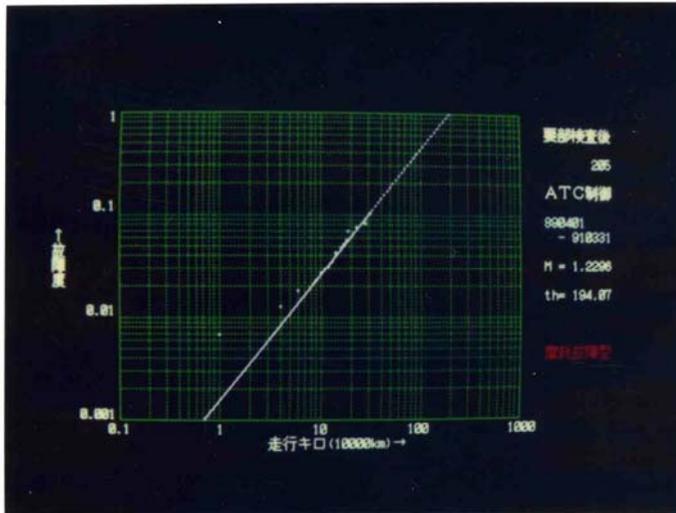
Summary of Train Schedule : heavy

T	P	Original Plan		New Plan	
		NT	LF	NT	LF
6	2474	5	0.68	5	0.68
7	3341	5	0.92	5	0.92
8	3344	5	0.93	5	0.93
9	2692	4	0.93	5	0.74
10	3373	5	0.93	5	0.93
11	3386	5	0.94	5	0.94
12	2700	4	0.93	5	0.75
13	3329	5	0.92	5	0.92
14	2355	4	0.81	4	0.81
15	1979	4	0.68	3	0.91
16	1733	4	0.60	3	0.80
17	2322	5	0.64	4	0.60
18	1769	4	0.61	3	0.82
19	1259	3	0.58	3	0.58
20	712	2	0.49	2	0.49
Total	36768	64	0.79	62	0.82

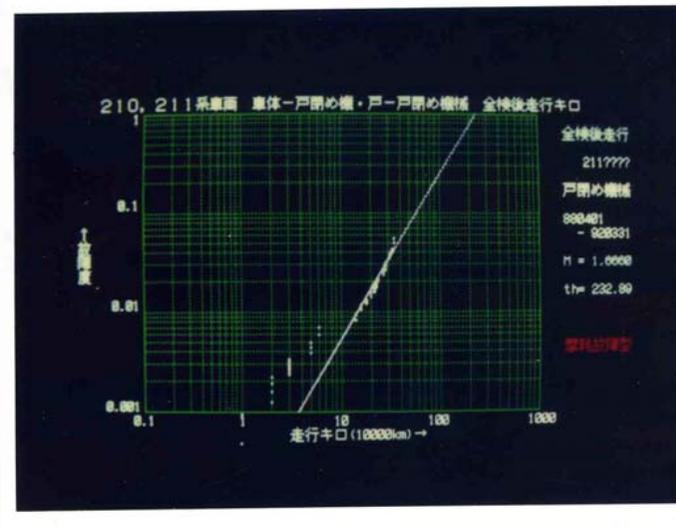
・MARSのジャーナル・テープの読み込みが最大の課題
・この時点での国鉄は、労使紛争が厳しく、動的な乗務員の運用が不可能であり、実現しなかったが、現在は、この方式を採用した臨時列車の運行が行われている。

信頼性管理システム

車両の故障履歴データから、故障のモードを推定し、メンテナンスに反映する。



・鉄道技研時代に、阿部俊一氏により開発された、中途打切りデータの信頼性解析法を採用
・この図より、機器が摩耗故障のモードであることが分かる。



・国鉄時代は、労使紛争が厳しく、信頼性という言葉が合理化と同列に扱われており、信頼性の教育を受けた職員は非常に少なく、あまり普及しなかった。

仮想質問: 札幌～千歳空港間にリニアを運行した場合の仮想質問による需用推定

札幌～千歳空港間にリニアを運行した場合の需用予測

仮想質問

本日のあなたの旅行条件（旅行目的、出発地、航空便、手荷物等）すべて同じ場合、今回利用した交通機関と比べて、あなたはリニアモーターカーを利用しますか。

空港ターミナル
到着ロビーから
徒歩 3分

リニアモーターカー

札幌ターミナル
JR札幌駅まで
乗換徒歩 3分

所要時間： 8分

運行時隔： 20分

利用料金： 1,200円

JR Railway Technical Research Institute

仮想質問の問題点

[アンケート結果の単純集計は大幅な偏りがある]

[仮想質問の誤差項の導入]

通常の確率効用 e_n の外にSP誤差を示す e_n' 、およびSPの偏りを表現する δ を導入した。

$V_i = U_i + e_i$ (既存交通機関)

$V_i = U_n + e_n + \theta e_n' + \delta$ (新設交通機関)

アンケート調査データからこの θ と δ を推定し、この項を取り除くことにより偏りを無くす。

[仮想質問と現在の機関選択データの結合]

[参照データの導入：特急との比較]

JR Railway Technical Research Institute

表2 誤差パラメータの推定結果

	特急	快速	普通	バス	自家用
μ	0.729 (3.36)	0.476 (1.83)	0.903 (2.88)	0.695 (2.87)	1.31 (2.02)
η'	0.662 (1.40)	0.234 (0.59)	0.584 (1.23)	1.177 (2.83)	1.59 (1.90)
参照構造	バス	特・バ	特・バ	特急	特・バ

- ・アンケートに記載するリニアのサービス水準を実験計画法で9種類設定
- ・主観的な指標を導入することで、モデルの適合度が向上した。
- ・実績データと仮想質問を組合わせた推定を行うことにより、仮想質問のバイアスを除くことが出来た。
- ・JRの千歳空港乗入れの効果が、このモデルで説明できた。

ORの理論によるモデル化は、なぜ上手く行かないのか？

1. 問題の把握

問題は徐々に分かる

2. 適用可能なORモデルの探索

モデルを見つけられない

3. モデル化

モデルが単純すぎる

4. 解の探索法の適用

データが無い
答えが出ない

5. 解の評価

解の質が悪い／評価が困難

6. 問題への解の適用

実用上の制約を満たさない

7. 状況の変化に対応

解の連続性が無い
モデルが改修できない

8. 問題解決？

ORは役に立たない？

ORのモデル・理論は、実践的か？

