

A dynamic discrete network design problem for maintenance planning in traffic networks

Pirmin & Fontaine Stefan Minner
Annals of Operations Research volume 253, pages757–772 (2017)

須賀理帆

東京大学交通・都市・国土学研究室 修士1年

論文情報

タイトル	A dynamic discrete network design problem for maintenance planning in traffic networks
出版年	25 March 2016
引用数	18
著者名	Pirmin & Fontaine Stefan Minner

研究のサマリー・良かった点・悪かった点

研究のサマリー

離散ネットワーク設計問題(DNDP)を拡張することにより、ネットワークの道路メンテナンス計画の動的モデル(DDNDP)をベンダース分解アルゴリズムによって提案した。

良かった点

- 他のアルゴリズムとの比較を行った結果、ベンダース分解の結果が最も良かった。(混雑度が低かった)

悪かった点

- 他のアルゴリズムは実装の際、予算制約により解が見つからなかった。
- ベンダース分解についての説明はあったが、他のアルゴリズムの詳しい説明がなかった。

新規性・有用性・信頼度

新規性

動的ネットワーク離散モデルに関する論文は少ない。一定の予算制限を加え、メンテナンス作業の長期計画を計算し、ネットワークの渋滞を最小限に抑え、DDNDP の2レベルモデルを導入しベンダース分解法で解いた。

有用性

2レベル最適化問題・ベンダース分解法による解き方

信頼度

ベンダース分解といくつかのアルゴリズムを比較。

自分の研究との関連性

自分の研究背景

人口減少と高齢化を背景に限られた予算と人員で維持管理していくために、舗装の早期劣化を防ぎ道路のライフサイクルコスト縮減を図る。



自分の研究内容

- ・大規模データを使った劣化の予測モデル。
- ・劣化状態と修繕工法の組み合わせ最適に加え、道路修繕工法や劣化状態、交通量で将来の劣化を予測する。(交通量、予算が制約条件になる。)

この論文との関連性

ベンダース分解を用いて最適な道路メンテナンス計画を提案している点

論文背景・問題意識

背景

世界中の多くの大都市は渋滞の問題に直面している。
道路の老朽化は渋滞を引き起こす要因の1つであり、ネットワーク全体の品質が低下させ、さらに混雑が発生させるため定期的なメンテナンス作業が必要。

課題

- 道路のメンテナンス中は、道路の収容力が低下したり、道路が閉鎖されたりするため、交通渋滞が発生しユーザーの移動時間が長くなる。(メンテナンス後は移動時間が短縮する)
- メンテナンスを先延ばしにすると、修理費用が増加する可能性がある。



メンテナンス作業によって発生するネットワーク内の渋滞を最小限に抑えるために、ネットワークの最適なメンテナンス計画を考える。

既往研究

DNDP(Discrete Network Design Problem)

都市全体の渋滞を最小限に抑えるために、都市の道路ネットワーク内のどのリンクを改善する必要があるかを決定する問題。非線形混合整数計画法モデルが開発され、分枝限定法で解かれた。

Bard, J. F. (1998). *Practical bilevel optimization: Algorithms and applications*. Berlin: Springer.

渋滞を最小限に抑えるという目標と旅行時間を最小限に抑えるという目標が必ずしも同じではない。

Braess, D., Nagurney, A., & Wakolbinger, T. (2005). On a paradox of traffic planning. *Transportation Science*, 39(4), 446–450.

ネットワーク内のすべての旅行者が出発地から目的地までの移動時間を最小限にするという前提の下で、最適なメンテナンス計画を見つけることによって、混雑が最小限に抑えられる。

LeBlanc, L. J. (1975). An algorithm for the discrete network design problem. *Transportation Science*, 9(3), 183–199.

目次

1. DDNDPとは

2. 手法：ベンダー分解アルゴリズムの提案

- 2レベル最適化問題
- 上位問題・下位問題の定式化
- 非線形問題を線形化
- 下位双対問題
- 最適性・実現可能性カットの生成

3. 他のアルゴリズムとの比較

- 貪欲法/分枝限定法/遺伝子アルゴリズムとの比較
- Sioux-Fall networkで実装

4. 考察と研究の発展性

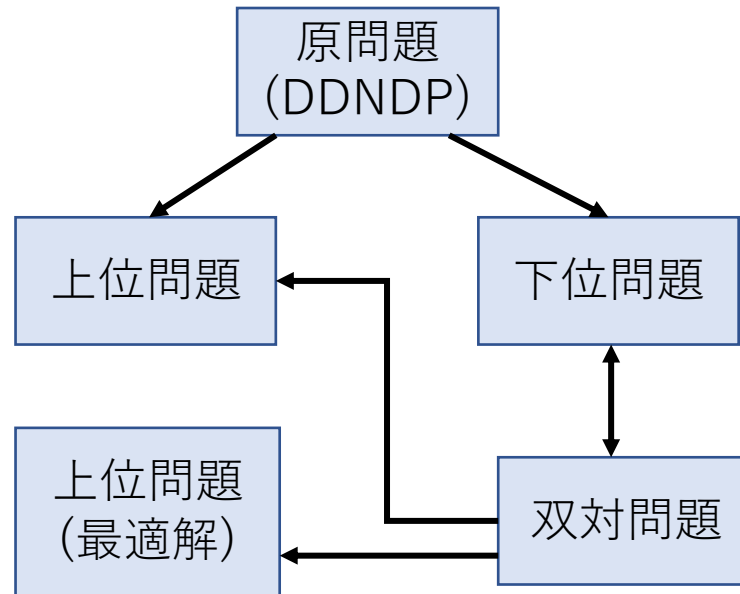
1 動的離散ネットワーク設計問題とは

DDNDP: Dynamic Discrete Network Design Problem

既往研究DNDP(Discrete Network Design Problem)を発展し、メンテナンス予算の制限があるネットワークの中で、渋滞を最小化するような動的モデルをベンダーズ分解で提案する。

2 ベンダーズ分解 Benders decompositionとは

最適化問題をそれより小さな規模の上位問題(線形部分)と下位問題(非線形部分)に分割し、両者を交互に繰り返し解くことにより、原問題の最適解を見つけるアルゴリズム。(大規模問題の解法)
上位問題において一部の変数を固定すると、残りの変数に関する最適化問題が線形計画問題となるような問題に対して有効なアルゴリズム。また、上位問題と双対問題のどちらか一方が両方の解法となる。

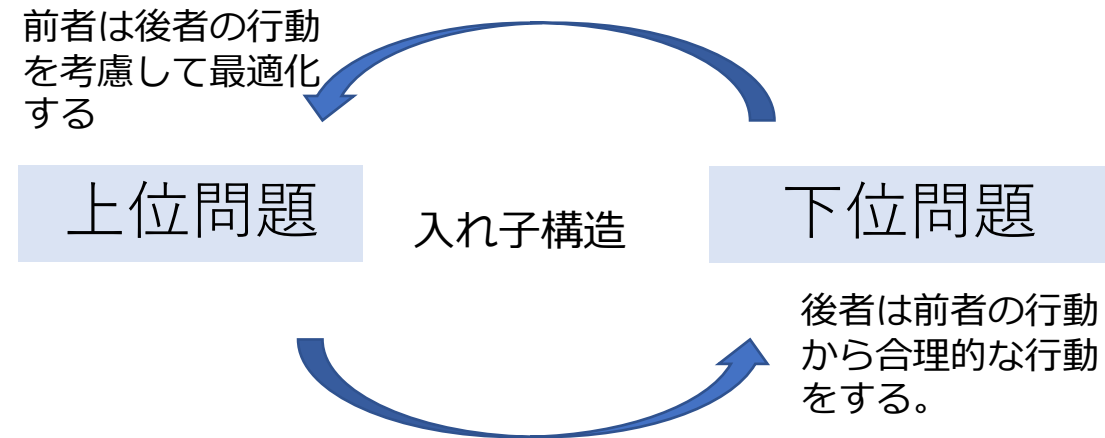


2レベル最適化問題とは

Bilevel optimization problem

2レベル最適化問題とは

上位問題（前者の問題）と下位問題（後者の問題）の2つの階層をもつ最適化問題。下位問題の目的関数や制約条件により決定される最適解を，上位の目的関数や制約条件で考慮して，下位問題の未定パラメータを決定する最適化問題。



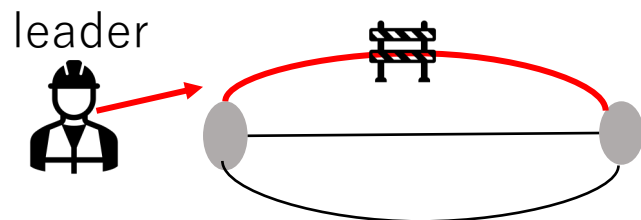
DDNDPに2レベル最適化問題を導入

原問題

メンテナンス予算の制限があるネットワークの中で、渋滞を最小化するメンテナンス計画を考える。

上位問題：Leader(前者)

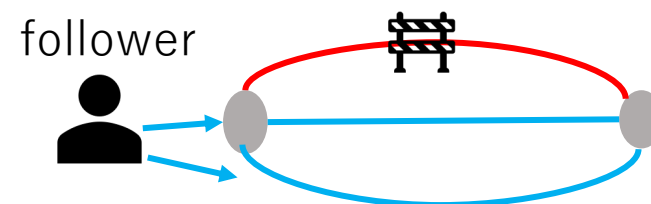
ネットワーク内のどの道路をどの期間にメンテナンスするか決定する。また、ネットワークの最適なメンテナンス計画を見つけることによって、渋滞を最小限に抑えようとする。



Leaderは制約とFollowerの行動が最適である条件の下で、目的を最適化する

下位問題：Follower(後者)

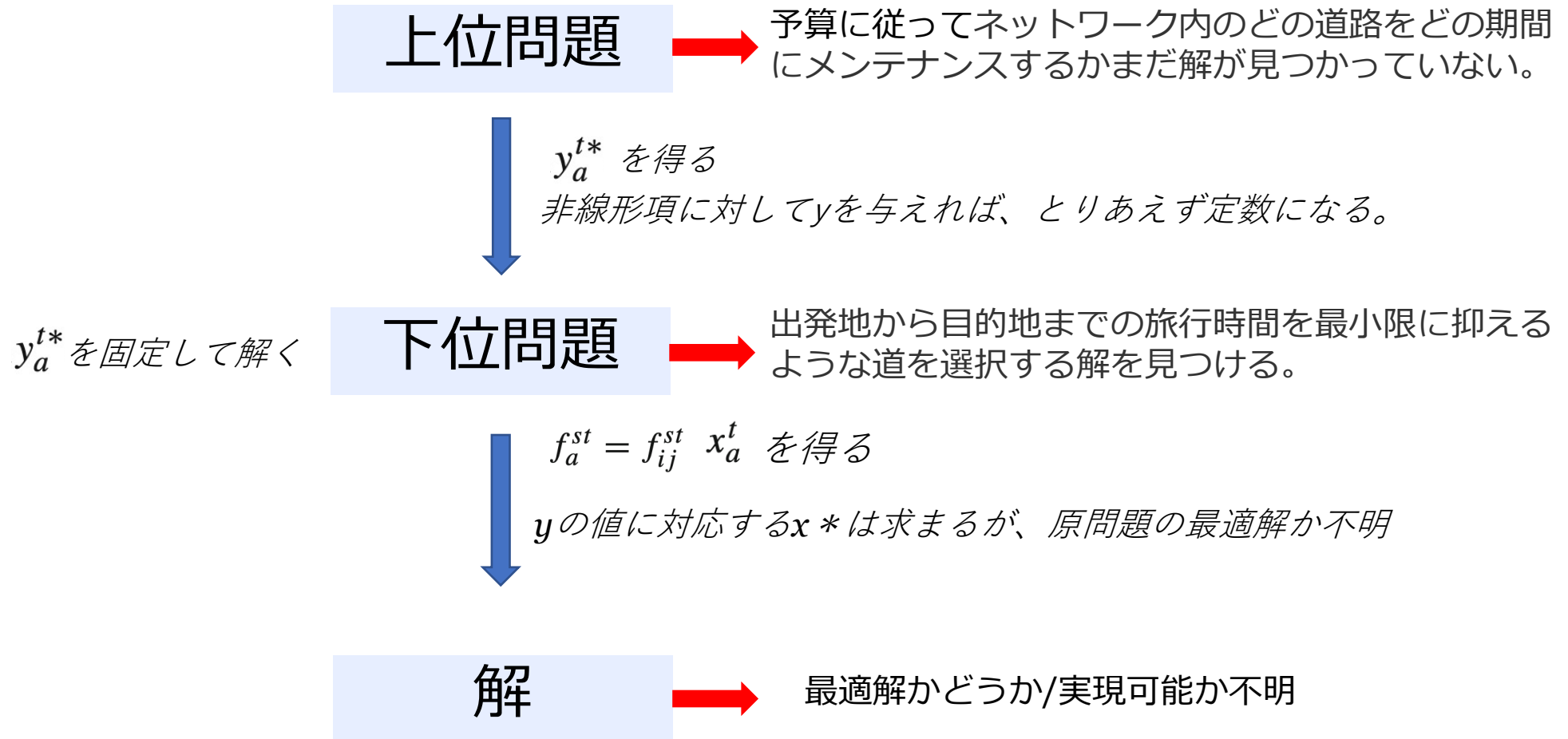
ネットワーク内のどの道を選択するかを決定する。そのとき、出発地から目的地までの旅行時間を最小限に抑えるような道を選択する。



FollowerはLeaderの決定に依存する制約に従って目的関数を制約化する

ベンダーズ分解法の流れ

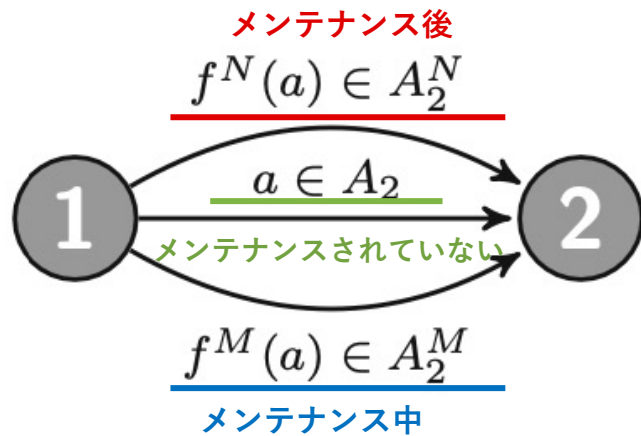
できるだけ線形計画問題をときたい。



最適性カット・実現可能カットの生成

情報

A2リンクの表現例



- A_2 : メンテナンスの必要がある道
- $f^N(a)$: メンテナンス後のリンク a
- $f^M(a)$: メンテナンス中のリンク a

旅行時間関数

$$h_a^t(x) := T_a \left(1 + K_a \left(\frac{x}{c_a r_a^t} \right)^4 \right)$$

自由流動旅行時間 渋滞影響
パラメータ 各 a の容量制限 信頼性指標

Bureau of Public Roads(1964)

信頼性指標 : 各期間の使用容量の割合。
この信頼性は旅行時間が長くなるにつれて年々減少し、
メンテナンス後に増加する。

上位問題(Leader) 定式化

目的関数

すべての期間でネットワーク内の総混雑度を最小化する

$$\min \sum_{a \in A, t \in T} \underline{x_a^t} \underline{h_a^t(x_a^t)}$$

累積流動変数(期間t内のa上において)

制約条件

$$s.t. \sum_{a \in A_2} \underline{b_a^t} \underline{y_a^t} \leq \underline{B} \quad \forall t \in T$$

費用(期間t内でaをメンテナンスするための) 予算制約

$$\sum_{t \in T, t \leq l_a} y_a^t = 1 \quad \forall a \in A_2^M$$

$$y_a^t + y_{a'}^t + y_{a''}^t = 1 \quad \forall a \in A_2, a' = f^M(a), a'' = f^N(a), t \in T$$

$$y_a^{t-1} \leq y_a^t + y_{a'}^t \quad \forall a \in A_2, a' = f^M(a), t \in T \setminus \{1\}$$

$$y_{a'}^{t-1} + y_{a''}^{t-1} \leq y_{a''}^t \quad \forall a \in A_2, a' = f^M(a), a'' = f^N(a), t \in T \setminus \{1\}$$

$$y_a^t \in \{0, 1\} \quad \forall a \in \bar{A}_2, t \in T$$

y_a^t : 決定変数

各期間の予算に従った各 $a \in A_2$ のメンテナンス期間の決定。

$y_a^t = 1$: 期間t内にaがまだメンテナンスされていない。

決定変数yを固定すると、残りの決定変数xに関する最適化問題は線形計画問題となり、容易に解ける

下位問題(follower) 定式化

上位問題で決定変数 $(y_a^t)^*$ を固定することによって, 下位問題が部分問題に分割できる

目的関数 各期間のすべてのODペアの旅行時間の合計を最小化する(ユーザー最適化)

$$\min \sum_{a \in A, t \in T} \int_0^{x_a^t} h_a^t(x) dx = \min \sum_{a \in A, t \in T} T_a x_a^t \left(1 + \frac{K_a}{5c_a^4 r_a^t} (x_a^t)^4 \right)$$

制約条件

$$\text{s.t. } \sum_{j \in N} f_{kj}^{st} - \sum_{i \in N} f_{ik}^{st} = \begin{cases} -\sum_{r \in R} q_{rs}, & k = s \\ q_{rs}, & k = r \\ 0, & \text{o.w.} \end{cases} \quad \forall s \in S, k \in N, t \in T$$

$$\underline{x}_a^t = \sum_{s \in S} \underline{f}_a^{st} \quad \forall a \in A, t \in T$$

$$x_a^t \leq M \underline{y}_a^t \quad \forall a \in \bar{A}_2, t \in T$$

$$\underline{f}_{ij}^{st} \geq 0 \quad \forall s \in S, (i, j) \in A, t \in T$$

未定パラメータの **連続変数** $f_a^{st} = f_{ij}^{st}$ と **累積流動変数** x_a^t を決定したい。



下位問題は各期間 $|T|$ の部分問題に分解して解いていく。

目的関数の区分線形近似

最適解 $f(x) = x^5$ (凸状非線形関数)



区分的線形関数を使って近似化する

$$\bar{f}(x) := \begin{cases} a_i x + b_i, & \text{for } x \in [\vartheta_{i-1}, \vartheta_i), i = 1, \dots, m-1 \\ a_i x + b_i, & \text{for } x \in [\vartheta_{i-1}, \infty), i = m \end{cases}$$

$$f_i := f(\vartheta_i)$$

m : 線形セグメントの数

$$\vartheta_m \geq \max_{a \in A, t \in T} x_a^t \quad a_i := \frac{f_i - f_{i-1}}{\vartheta_i - \vartheta_{i-1}} \quad b_i := -\vartheta_{i-1} a_i + f_{i-1}$$

$$(\vartheta_0, f_0), (\vartheta_1, f_1), \dots, (\vartheta_m, f_m)$$

$$\sum_{r \in R, s \in S} q_{rs}$$

目的関数関数の区分線形近似

最適解 $f(x)$

原問題

目的関数 $\min f_0 + a_1 x_1 + \sum_{i \in I} (a_i - a_{i-1}) x_i$

制約条件 $\text{s.t. } x_1 \leq x_i + \vartheta_{i-1} \quad i \in I$
 $x_i \geq 0 \quad i \in I$

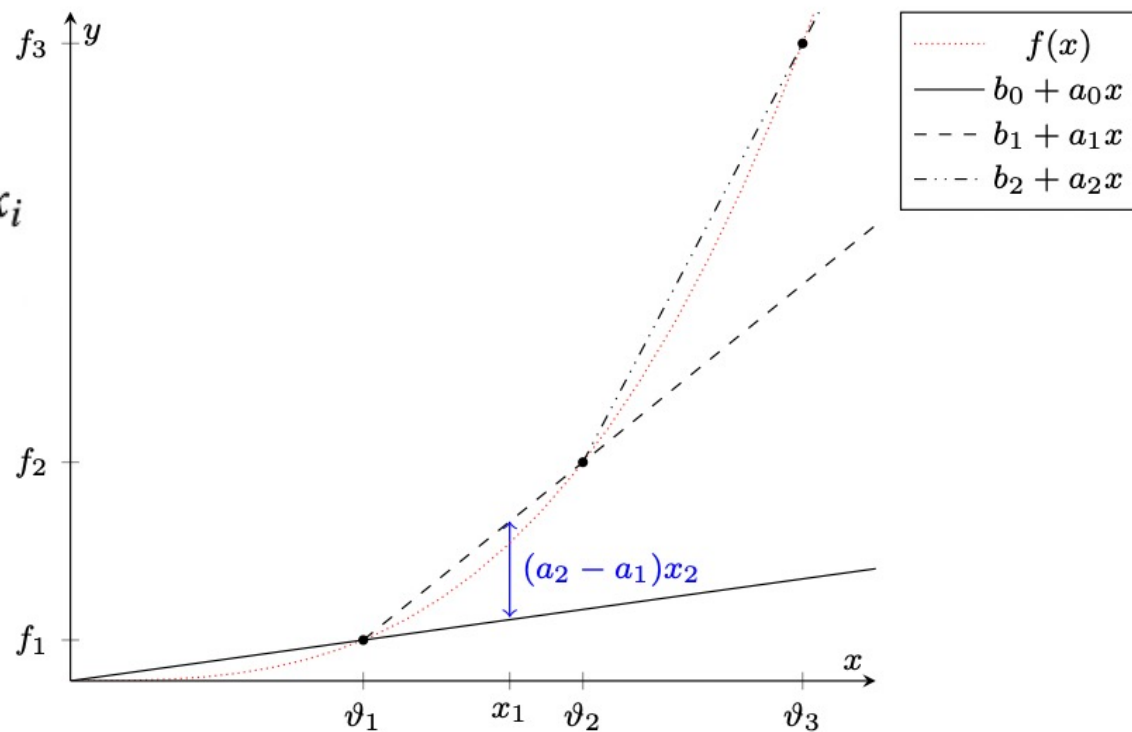


Fig.2 4つのデータポイントと $f(x)$ の近似の例

すべての期間 $t \in T$ で各リンク $a \in A$ の旅行時間が区分線形近似化された

ベンダーズ分解の流れ

ベンダーズ分解：主要構造

Step 0 : 上限 $UBD = \infty$ 。 y^* を y の任意の実行可能解に設定。

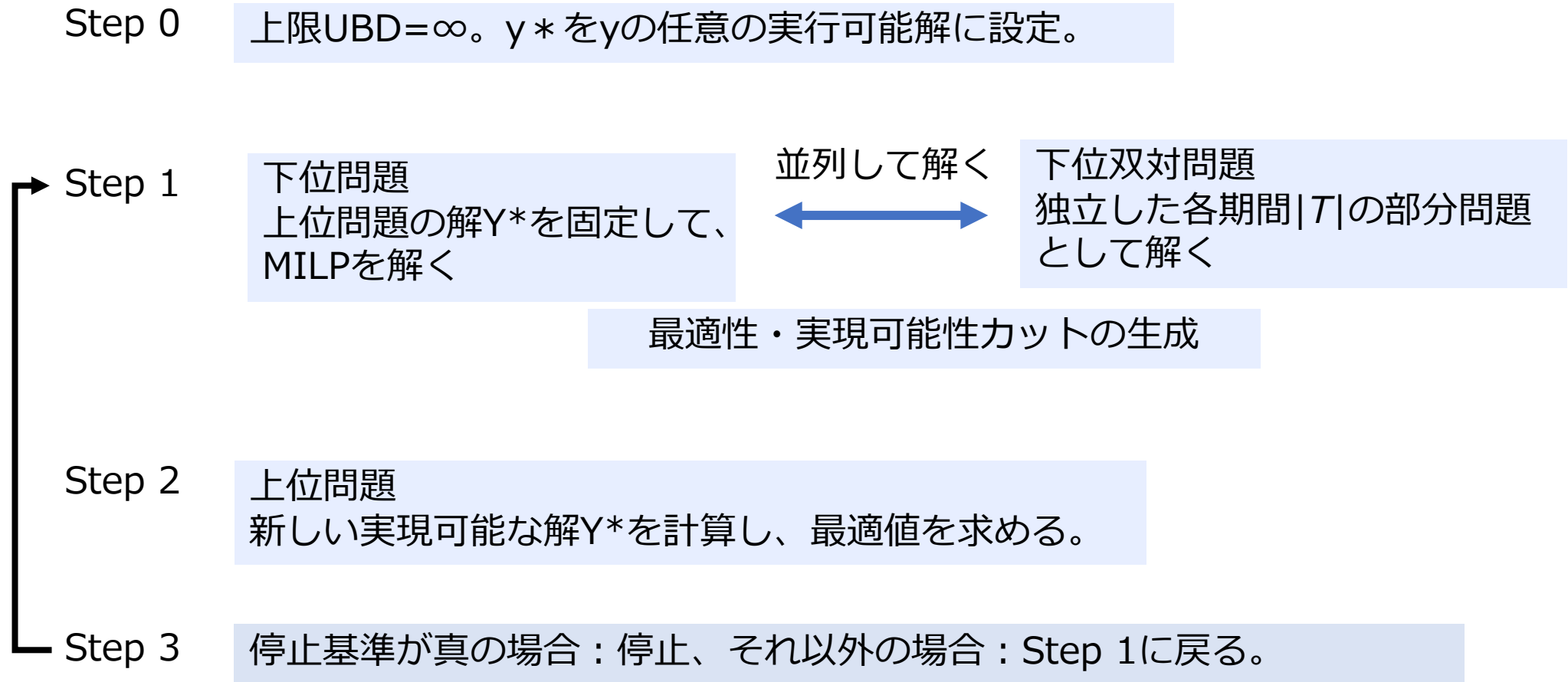
Step 1 :

- $y = y^*$ の下位問題の部分問題を並列に解きます。
- 下位問題の最適解の値を ZS とする。
- 上限値 $UBD = \min\{UBD, ZS\}$ を更新する。
- 下位問題の上限が決まっている場合は、上位問題に最適化カットを追加し、そうでない場合は上位問題に実行可能性カットを追加する。

Step 2 : 上位問題を解き、解 y^* を保存する。解いた上位問題の最適解の値を ZM とする。

Step 3 : 停止基準が真の場合 : 停止、それ以外の場合 : Step 1 に戻る。

反復の流れ



下位問題(Follower) KKT条件で最適性を保証

決定変数を固定し制約条件に従って最小化する

KKT条件では制約に双対変数 $u_{sk}^t, v_a^t, w_{ia}^t$ を導入する。

目的関数

$$\min \sum_{t \in T} \left(\sum_{\substack{a \in A \\ s \in S}} C_{1a}^t f_a^{st} + \sum_{\substack{i \in I \\ a \in A}} C_{ia}^t x_{ia}^t \right)$$

$$\begin{aligned} C_{1a}^t &:= T_a \left(1 + \frac{K_a}{c_a r_a^t} a_1 \right) & C_{ia}^t &:= \frac{T_a K_a}{c_a r_a^t} (a_i - a_{i-1}) \\ \hat{C}_{1a}^t &:= T_a \left(1 + \frac{K_a}{5c_a r_a^t} a_1 \right) & \hat{C}_{ia}^t &:= \frac{T_a K_a}{5c_a r_a^t} (a_i - a_{i-1}) \end{aligned}$$

制約条件

下位
問題
制約

$$\sum_{j \in N} f_{kj}^{st} - \sum_{i \in N} f_{ik}^{st} = q_{ks} \quad \forall s \in S, k \in N, t \in T$$

$$\sum_{s \in S} f_a^{st} \leq M y_a^{t*} \quad \forall a \in \bar{A}_2, t \in T$$

$$\sum_{s \in S} f_a^{st} \leq x_{ia}^t + \vartheta_{i-1} \quad \forall a \in A, i \in I, t \in T$$

KKT
条件

$$\sum_{\substack{a \in A \\ s \in S}} \hat{C}_{1a}^t f_a^{st} + \sum_{\substack{i \in I \\ a \in A}} \hat{C}_{ia}^t x_{ia}^t \leq \sum_{\substack{s \in S \\ k \in N}} q_{k,s} u_{ks}^t$$

双対変数

$$+ \sum_{a \in \bar{A}_2} M v_a^t y_a^t + \sum_{\substack{i \in I \\ a \in A}} w_{ia}^t \vartheta_{i-1} \quad \forall t \in T$$

双対
制約

$$u_{sk}^t - u_{sj}^t + \underbrace{\chi_a v_a^t}_{\text{指示関数}} + \sum_{i \in I} w_{ia}^t \leq \hat{C}_{1a}^t \quad \forall s \in S, a = (k, j) \in A, t \in T$$

$$- w_{ia}^t \leq \hat{C}_{ia}^t \quad \forall a \in A, i \in I, t \in T$$

$$f_a^{st} \geq 0 \quad \forall s \in S, a \in A, t \in T$$

$$x_{ia}^t \geq 0 \quad \forall i \in I, a \in A, t \in T$$

$$u_{sk}^t \in \mathbb{R} \quad \forall s \in S, k \in N, t \in T$$

$$v_a^t \leq 0 \quad \forall a \in \bar{A}_2, t \in T$$

$$w_{ia}^t \leq 0 \quad \forall i \in I, a \in A, t \in T$$

指示関数

$a \in \bar{A}_2$ のとき 1

$a \in A, i \in I, t \in T$ のとき 0

下位問題(Follower)

KKT条件の決定変数 y_a^t を媒介変数 μ_a^t に置き換えて線形化する

下位問題 制約条件

— 決定変数

$$y_a^t = 0 \text{ のとき } \mu_a^t = 0$$

$$y_a^t = 1 \text{ のとき } \mu_a^t = v_a^t$$

KKT
条件

$$\sum_{\substack{a \in A \\ s \in S}} \hat{C}_{1a}^t f_a^{st} + \sum_{\substack{i \in I \\ a \in A}} \hat{C}_{ia}^t x_{i,a}^t \leq \sum_{\substack{s \in S \\ k \in N}} q_{k,s} u_{ks}^t \\ + \sum_{a \in \bar{A}_2} M \underline{\mu}_a^t + \sum_{\substack{i \in I \\ a \in A}} w_{ia}^t \vartheta_{i-1} \quad \forall t \in T$$

$$\underline{\mu}_a^t \leq v_a^t + M_2(1 - y_a^t) \quad \forall a \in \bar{A}_2, t \in T$$

$$\underline{\mu}_a^t \geq v_a^t \quad \forall a \in \bar{A}_2, t \in T$$

$$\underline{\mu}_a^t \geq -M_2 y_a^t \quad \forall a \in \bar{A}_2, t \in T$$

$$\underline{\mu}_a^t \leq 0 \quad \forall a \in \bar{A}_2, t \in T$$

下位双対問題

下位双対問題の実現可能集合は決定変数に依存しない。

上位問題の最小値と下位問題の双対問題の最大値は一致するので、
双対問題の目的関数の値を最大化する双対変数の組み合わせを求める。

目的関数

$$\begin{aligned} \max \quad & \sum_{\substack{s \in S \\ k \in N}} q_{ks} \alpha_{sk}^t + \sum_{a \in \bar{A}_2} \beta_a^t + \sum_{\substack{a \in A \\ i \in I}} \vartheta_{i-1} \gamma_{ia}^t + \sum_{\substack{a \in A \\ s \in S}} \hat{C}_{ia}^t v_{sa}^t + \sum_{\substack{a \in A \\ i \in I}} \hat{C}_{ia}^t \tau_{ia}^t \\ & + \sum_{a \in \bar{A}_1} (M_2(1 - y_a^{t*}) \lambda_a^{1t} - M_2 y_a^{t*} \lambda_a^{3t}) \end{aligned}$$

双対変数

$$\alpha_{sk}^t, \beta_a^t, \gamma_{ia}^t, v_{sa}^t, \tau_{ia}^t, \delta^t, \lambda_a^{1t}, \lambda_a^{2t}, \lambda_a^{3t}$$

双対条件

$$f_a^{st}, x_{ia}^t, u_{sk}^t, v_a^t, \mu_a^t, w_{ia}^t$$

$$\begin{aligned} \text{s.t.} \quad & \alpha_{sk}^t - \alpha_{sj}^t + \chi_a \beta_a^t + \sum_{i \in I} \gamma_{ia}^t + \hat{C}_{1a}^t \delta^t \leq C_{1a}^t \quad \forall s \in S, a = (k, j) \in A \\ & -\gamma_{ia}^t + \hat{C}_{ia}^t \delta^t \leq C_{ia}^t \quad \forall a \in A, i \in I \\ & -q_{ks} \delta^t + \sum_{j \in N} v_{kj}^{st} - \sum_{i \in N} v_{ik}^{st} = 0 \quad \forall s \in S, k \in N \\ & -\lambda_a^{1t} - \lambda_a^{2t} + \sum_{s \in S} v_{sa}^t \geq 0 \quad \forall a \in A \\ & \lambda_a^{1t} + \lambda_a^{2t} + \lambda_a^{3t} - M \delta^t \geq 0 \quad \forall a \in A \\ & -\vartheta_{i-1} \delta^t + \sum_{s \in S} v_{sa}^t - \tau_{ia}^t \geq 0 \quad \forall i \in I, a \in A \end{aligned}$$

$$\begin{aligned} & \alpha_{sk}^t \in \mathbb{R} \quad \forall s \in S, k \in N \\ & \beta_a^t, \lambda_a^{1t} \leq 0 \quad \forall a \in \bar{A}_2 \\ & \lambda_a^{2t}, \lambda_a^{3t} \geq 0 \quad \forall a \in \bar{A}_2 \\ & \delta^t \leq 0 \\ & v_{sa}^t \leq 0 \quad \forall a \in A, s \in S \\ & \gamma_{ia}^t, \tau_{ia}^t \leq 0 \quad \forall i \in I, a \in A \end{aligned}$$

双対制約

下位双対問題で反復を制限

下位問題の期間 $t \in T$ が制限されていない場合、
下位双対問題の目的関数に上限解 M_3 をあたえることによって反復を制限する。

下位双対問題 目的関数

$$\sum_{\substack{s \in S \\ k \in N}} q_{ks} \alpha_{sk}^t + \sum_{a \in \bar{A}_2} \beta_a^t + \sum_{\substack{a \in A \\ i \in I}} \vartheta_{i-1} \gamma_{ia}^t + \sum_{\substack{a \in A \\ s \in S}} \hat{C}_{ia}^t \nu_{sa}^t + \sum_{\substack{a \in A \\ i \in I}} \hat{C}_{ia}^t \tau_{ia}^t \\ + \sum_{a \in \bar{A}_2} (M_2(1 - y_a^{t*}) \lambda_a^{1t} - M_2 y_a^{t*} \lambda_a^{3t}) \leq \underline{M_3}$$

極端な線を制限するために十分な大きい数

下位双対問題において上位問題最適性カットの生成

下位双対問題の解 $\alpha^*, \beta^*, \gamma^*, \delta^*, \nu^*, \tau^*, \lambda^*$ がすべてのメンテナンス期間 $t \in T$ で実現可能な場合、最適性カットを生成し上位問題に追加する。

上位問題 目的関数

$$\sum_{t \in T} \left(\sum_{\substack{s \in S \\ k \in N}} q_{ks} \alpha_{sk}^{t*} + \sum_{a \in \bar{A}_2} \beta_a^{t*} + \sum_{\substack{a \in A \\ i \in I}} \vartheta_{i-1} \gamma_{ia}^{t*} + \sum_{\substack{a \in A \\ s \in S}} \hat{C}_{ia}^t \nu_{sa}^{t*} + \sum_{\substack{a \in A \\ i \in I}} \hat{C}_{ia}^t \tau_{ia}^{t*} \right. \\ \left. + \sum_{a \in \bar{A}_2} (M_2(1 - y_a^t) \lambda_a^{1t*} - M_2 y_a^t \lambda_a^{3t*}) \right) \leq \underline{z}$$

最適性カット

上位問題の最適解に下限を与える

下位問題は解決され、実現可能カットが生成され、上位問題に加えられる。

上位問題 目的関数

$$\sum_{t \in T} \left(\sum_{\substack{s \in S \\ k \in N}} q_{ks} \alpha_{sk}^{t*} + \sum_{a \in \bar{A}_2} \beta_a^{t*} + \sum_{\substack{a \in A \\ i \in I}} \vartheta_{i-1} \gamma_{ia}^{t*} + \sum_{\substack{a \in A \\ s \in S}} \hat{C}_{ia}^t v_{sa}^{t*} + \sum_{\substack{a \in A \\ i \in I}} \hat{C}_{ia}^t \tau_{ia}^{t*} \right. \\ \left. + \sum_{a \in \bar{A}_2} (M_2(1 - y_a^t) \lambda_a^{1t*} - M_2 y_a^t \lambda_a^{3t*}) \right) \leq \underline{0}$$

上位問題へ最適性カットと実行可能カットの追加

カットを追加した上位問題

目的関数

$\min z$

制約条件

s.t. (2-7)

$c_o \quad \forall c_o \in \underline{C_O}$ 最適性カットのセット

$c_f \quad \forall c_f \in \underline{C_F}$ 実現可能カットのセット

Step3→最適性の証拠が得られていればこれを最適解として計算を終了する。
そうでなければ下位問題の最適化結果をフィードバックしつつ**Step1**へ戻る。

3 他のアルゴリズムとの比較

比較するアルゴリズム

- **BD(Benders Decomposition :ベンダース分解)**
- **LRF(Lowest reliability First)**
最も信頼性(使用容量の割合)の低い道を選ぶ貪欲法
- **HCF(Highest Congestion First)**
予算を超過するまで最も高い渋滞度の道を選ぶ貪欲法
- **B&B(Branch-and-Bound:分枝限定法)**
MIPを解くための並列計算手法を使う。
- **GA(Genetic Algorithm:遺伝子アルゴリズム)**
染色体をA2の長さの整数ベクトルとして定義。ベクトル内の各点は1からT(メンテナンス期間)をとり、予算制約の違反は目的関数の中でペナルティ化される。

Sioux- Fall networkの実装

Sioux- Fall network の実装を2回行った

1回目 少ないデータで実装(実行時間1時間)

- node: 24, arcs: 76(A2=38), OD-pairs:528, 総流量:3606, 計画期間(P):10年
- 予算条件(B=50,75,100)
- 最初の期間ですべての道路をメンテナンスする費用は420

2回目 1回目を拡張して実装(実行時間4時間)

- node: 100, arcs: 317, OD-pairs:817, 計画期間:8年
- 予算条件(B=300,400)
- 最初の期間のメンテナンスする費用は2062

* メンテナンス期限の制約はなくした。

BDは制約に違反していないにも関わらず混雑値が最も低かった

Table 3 Objective values for Sioux-Fall network (in 10^9) () 予算制約を超過した期間の数 <> 非線形目的関数で評価

B 予算	m データポイント	BD	LRF	HCF	B&B	GA		
						Best	Worst	Mean
50	20	8.07	8.49 ⁽⁵⁾	8.42 ⁽³⁾	–	8.28 ⁽³⁾	8.76 ⁽⁷⁾	8.53 ^(4.7)
		<7.60>	<7.88>	<7.90>	–	<7.78>	<8.32>	<8.02>
	40	7.39	7.63 ⁽⁵⁾	7.60 ⁽³⁾	–	7.52 ⁽³⁾	8.00 ⁽⁷⁾	7.73 ^(4.8)
		<7.29>	<7.55>	<7.54>	–	<7.45>	<7.93>	<7.65>
	60	7.18	7.51 ⁽⁵⁾	7.49 ⁽³⁾	–	7.35 ⁽³⁾	7.80 ⁽⁷⁾	7.59 ^(4.9)
		<7.14>	<7.49>	<7.46>	–	<7.31>	<7.77>	<7.55>
75	20	8.00	8.05	8.39	8.20	8.02	8.27	8.13
		<7.53>	<7.52>	<7.88>	<7.77>	<7.48>	<7.74>	<7.64>
	40	7.18	7.28	7.62	7.68	7.26	7.56	7.40
		<7.1>	<7.21>	<7.55>	<7.61>	<7.17>	<7.47>	<7.31>
	60	7.06	7.18	7.52	7.38	7.27	7.46	7.34
		<7.03>	<7.16>	<7.48>	<7.35>	<7.21>	<7.43>	<7.31>
100	20	7.80	8.02	8.34	8.19	7.74	8.00	7.91
		<7.30>	<7.45>	<7.72>	<7.72>	<7.24>	<7.58>	<7.44>
	40	7.10	7.19	7.53	7.45	7.18	7.36	7.25
		<7.01>	<7.17>	<7.46>	<7.32>	<7.04>	<7.26>	<7.15>
	60	6.97	7.12	7.43	7.69	7.08	7.28	7.18
		<6.93>	<7.10>	<7.41>	<7.66>	<7.06>	<7.24>	<7.15>

予算は必要最小限であり、LRF・HCF・B&B・GAは解を見つけられなかったため、前の期間から残った予算を使用できるように設定

混雑値

2回目の実装結果 Sioux- Fall networkの拡張

BDは他のアルゴリズムと比べて良い解を見つけた

Table 4 Objective values for the Extended Sioux-Fall network (in 10^8) () 予算制約を超過した期間の数 <> 非線形目的関数で評価

<i>B</i> 予算	<i>m</i> データポイント	BD	LRF	HCF	GA		
					Best	Worst	Mean
300	20	5.38	5.41 ⁽³⁾	5.37 ⁽⁴⁾	5.39 ⁽⁴⁾ , 5.43 ⁽²⁾	5.45 ⁽³⁾ , 5.40 ⁽⁵⁾	5.42 ^(3.6)
		<5.17>	<5.20>	<5.18>	<5.18>, <5.23>	<5.26>, <5.21>	<5.21>
	40	5.13	5.16 ⁽³⁾	5.13 ⁽⁴⁾	5.14 ⁽¹⁾	5.18 ⁽³⁾ , 5.16 ⁽⁶⁾	5.16 ^(3.6)
		<5.06>	<5.09>	<5.07>	<5.07>	<5.13>, <5.10>	<5.10>
400	20	5.38	5.42	5.38	5.38	5.42	5.40
		<5.18>	<5.22>	<5.18>	<5.17>	<5.21>	<5.19>
	40	5.13	5.18	5.13	5.13	5.17	5.14
		<5.07>	<5.12>	<5.07>	<5.07>	<5.11>	<5.08>

混雑値

3 課題と今後の発展

- BD の収束は実現できなかったため、反復回数を減らし、適切な下限を見つけることが今後の研究で必要である。
- 古典的な遺伝的アルゴリズムは、低予算で実現可能な解を見つけるのに問題があり、特定の演算子も開発する必要がある。

所感

- ベンダース分解法を通して、非線形問題や混合整数計画問題についても学ぶことができた。
- 問題を上位問題と下位問題に分解して解くという考え方は、自分の研究でも参考になると思った。
- モデルを理解するにあたって、数学的な知識が足りなく完璧には理解することができなかったの
でこれから勉強していきたい。

補足資料

N : ノード A : arc(ネットワーク中の道) $s \in S \subset N$: 目的地 $r \in R \subset N$: 起点

A_2 : メンテナンスされる必要がある道

$|T|$: 期間(メンテナンスが終了する必要がある計画範囲) q_{rs} : 需要

l_a : 期日(までにメンテナンスされる必要がある)

b_a^t : 費用(期間 t 中のリンク a をメンテナンスするための)

B : 各期間の予算

上位問題(Leader) 定式化

目的関数 全ての期間でネットワーク内の総混雑度を最小化する

$$\min \sum_{a \in A, t \in T} x_a^t h_a^t(x_a^t)$$

制約条件

(2)各期間ごとの予算制限

(3)全ての道がある期限までにメンテナンスされる

(4)道がメンテナンス前・中・後か

(5)期間(t-1)内にメンテナンスされていないならば、 a はメンテナンスされるか、されないか

(6)期間(t-1)内か前にメンテナンスされたら、新たらしくなった a は t に使われる必要がある。

(7)1: 期間 t 内にメンテナンスされていない。0:期間 t 前・内にメンテナンスされた。

$$s.t. \sum_{a \in A_2} b_a^t y_a^t \leq B \quad \forall t \in T \quad (2)$$

$$\sum_{t \in T, t \leq l_a} y_a^t = 1 \quad \forall a \in A_2^M \quad (3)$$

$$y_a^t + y_{a'}^t + y_{a''}^t = 1 \quad \forall a \in A_2, a' = f^M(a), a'' = f^N(a), t \in T \quad (4)$$

$$y_a^{t-1} \leq y_a^t + y_{a'}^t \quad \forall a \in A_2, a' = f^M(a), t \in T \setminus \{1\} \quad (5)$$

$$y_{a'}^{t-1} + y_{a''}^{t-1} \leq y_{a''}^t \quad \forall a \in A_2, a' = f^M(a), a'' = f^N(a), t \in T \setminus \{1\} \quad (6)$$

$$y_a^t \in \{0, 1\} \quad \forall a \in \bar{A}_2, t \in T \quad (7)$$

下位問題(follower) 定式化

目的関数 各期間のすべてのODペアの移動時間の合計を最小化する。(ユーザー最適化)

$$\min \sum_{a \in A, t \in T} \int_0^{x_a^t} h_a^t(x) dx = \min \sum_{a \in A, t \in T} T_a x_a^t \left(1 + \frac{K_a}{5c_a^4 r_a^t} (x_a^t)^4 \right)$$

制約条件

(9) 需要を満たすような流動保存制約

(10) リンク上に集約された流量

(11) M: 期間中に通れるリンクのみが使われる。リンク上の最大流量はネットワークの総需要量によって制限される。

(12)

$$\text{s.t. } \sum_{j \in N} f_{kj}^{st} - \sum_{i \in N} f_{ik}^{st} = \begin{cases} -\sum_{r \in R} q_{rs}, & k = s \\ q_{rs}, & k = r \\ 0, & \text{o.w.} \end{cases} \quad \forall s \in S, k \in N, t \in T \quad (9)$$

$$x_a^t = \sum_{s \in S} f_a^{st} \quad \forall a \in A, t \in T \quad (10)$$

$$x_a^t \leq M(y_a^t)^* \quad \forall a \in \bar{A}_2, t \in T \quad (11)$$

$$f_{ij}^{st} \geq 0 \quad \forall s \in S, (i, j) \in A, t \in T \quad (12)$$