二次元空間 Fujita and Ogawa (1982) モデルの数値解法の開発

秋本 克哉[1]・赤松 隆[2]

[1]学生会員　東北大学　情報科学研究科（〒 980-8579 仙台市青葉区荒巻字青葉 6-6-06-408）
E-mail: katsuya.akimoto.q2@dc.tohoku.ac.jp
[2]正会員　東北大学教授　大学院情報科学研究科人間社会情報科学専攻（〒 980-8579 仙台市青葉区荒巻字青葉 6-6-06-408）
E-mail: akamatsu@plan.civil.tohoku.ac.jp

Fujita and Ogawa(1982) モデルは，複数都心の創発を初めて示した代表的な集積経済モデルである．このモデルを二次元空間に拡張すると，均衡解を数値的に求めるために莫大な計算量や記憶容量が必要である．本研究では，二次元空間 Fujita and Ogawa モデルの均衡解を効率的に求めるためのアルゴリズムを提案する．そのためのアプローチとして，このモデルと等価な最適化問題を構築し， Benders Decomposition を適用する．系統的な数値実験によって，アルゴリズムの有効性が示された．

Key Words: 集積経済モデル，複数都心， *Benders Decomposition*

1. はじめに

Fujita and Ogawa [1)][FO] モデルは，複数都心の創発を初めて示した代表的な集積経済モデルである．Osawa and Akamatsu[2)] は，一次元空間では複数都心の創発メカニズムを明らかにしたが，二次元空間では明らかにしていない．既存研究では，二次元空間でどのような複数都心パターンが創発するのかすら明らかになっておらず，より現実的な二次元空間の集積メカニズムの解明が求められる．

二次元空間 FO モデルの集積メカニズムを明らかにするためには，数値計算によって系統的に均衡解を求める必要がある．ただし，二次元空間 FO モデルで均衡解として創発すると考えられる多様な複数都心パターンを計算するには，素朴な実装では必要な計算量や記憶容量が莫大である．なぜなら，これらのパターンを正確に表現するためには，問題の規模 K を大きくして空間解像度を上げる必要があり，さらに，FO モデルで用いられる未知変数のオーダーは，OD ペア数の K^2 であるからである．

本研究の目的は，大規模な二次元空間 FO モデルの均衡解を効率的に求めることができるアルゴリズムを構築することである．本研究では，下位問題を解く際に必要な演算および記憶させる要素数を大幅に削減することで，全体問題を効率的に解くアルゴリズムを構築する．そのために，FO モデルと等価なポテンシャル問題を構築する．さらに，Benders Decomposition を適用し，上位問題と下位問題に分解する．

2. Fujita and Ogawa モデル

(1) 空間設定と立地主体の行動

K 個の離散的な立地点集合 $\mathcal{K}$ から成る，二次元の格子状空間をもつ都市を考える．地点 $i, j \in \mathcal{K}$ 間のユークリッド距離を T_{ij} とする．この都市の総土地面積を S とし，各地点の土地面積を $\bar{S} \equiv S/K$ に固定する．都市には，消費者と企業が存在し，それぞれの総数を N, M とする．

消費者は効用（合成財の需要量）, $z_{ij} = W_j - tT_{ij} - S_h R_i$, を最大化するように居住地 $i \in \mathcal{K}$ および勤務地 $j \in \mathcal{K}$ を選択する．ここで，R_i は地点 i の地代，S_h は消費者の固定消費面積，W_j は地点 j での賃金，t は通勤費用パラメータを表す．この選択行動の結果，(i, j) を選択する消費者の総数を n_{ij} とする．

企業は，一定量の土地面積と労働量 L を用いて，財を生産する．企業同士には技術的外部性があり，その生産量は face-to-face のコミュニケーションの量に依存するものとする．地点 $i \in \mathcal{K}$ に位置する企業は利潤, $\Pi_i(\boldsymbol{m}) = A_i(\boldsymbol{m}) - S_f R_i - W_i L$, を最大化するように立地点を選択する．ここで，$m_j$ は地点 $j \in \mathcal{K}$ に立地する企業の総数，S_f は企業の固定消費面積，L は労働需要，$A_i(\boldsymbol{m}) \equiv \sum_j d_{ij} m_j$ はアクセシビリティ関数，$d_{ij} \equiv \exp(-\tau T_{ij})$ は空間割引関数，τ は交流費用パラメータである．

(2) 均衡条件

FO モデルの均衡条件は，短期均衡条件と長期均衡条件から構成される．長期均衡は，$\boldsymbol{R}, \boldsymbol{W}$ を与件としたときの空間均衡状態であり，その結果として，$\boldsymbol{m}, \boldsymbol{n}$ が決まる．ここで，空間均衡状態とは，各立地主体は自らの選択する立地点を変更するインセンティブをもたない状態のことである．

$$\begin{cases} z^* = W_j - tT_{ij} - S_h R_i & \text{if } n_{ij} > 0 \\ z^* \geq W_j - tT_{ij} - S_h R_i & \text{if } n_{ij} = 0 \end{cases} \quad \forall i, j$$

$$\begin{cases} \Pi^* = A_i(\boldsymbol{m}) - S_f R_i - W_i L & \text{if } m_i > 0 \\ \Pi^* \geq A_i(\boldsymbol{m}) - S_f R_i - W_i L & \text{if } m_i = 0 \end{cases} \quad \forall i$$

ここで，z^* は均衡効用，Π^* は均衡利潤である．次に，短期均衡は，数量変数 $\boldsymbol{m}, \boldsymbol{n}$ を与件としたときの土地市場および労働市場の需給均衡であり，その結果として，価格変数 $\boldsymbol{R}, \boldsymbol{W}$ が決まる：

$$\begin{cases} S_h \sum_j n_{ij} + S_f m_i = \bar{S} & \text{if } R_i > 0 \\ S_h \sum_j n_{ij} + S_f m_i \leq \bar{S} & \text{if } R_i = 0 \end{cases} \quad \forall i$$

$$\begin{cases} \sum_i n_{ij} = L m_j & \text{if } W_j > 0 \\ \sum_i n_{ij} \geq L m_j & \text{if } W_j = 0 \end{cases} \quad \forall j$$

(3) 等価な最適化問題

均衡解を効率的に求めるために，均衡条件と等価な最適化問題を構成する．その最適化問題は以下の通りである：

$$[\mathbf{P1}] \min_{\boldsymbol{m} \in \mathcal{M}, \boldsymbol{n} \in \mathcal{N}(\boldsymbol{m})} Z(\boldsymbol{m}, \boldsymbol{n}) \equiv -Z_1(\boldsymbol{m}) + Z_2(\boldsymbol{n})$$

ここで，

$$Z_1(\boldsymbol{m}) = \frac{1}{2} \sum_i \sum_j d_{ij} m_i m_j$$

$$Z_2(\boldsymbol{n}) = t \sum_i \sum_j T_{ij} n_{ij}$$

$$\mathcal{M} = \left\{ \boldsymbol{m} \geq \boldsymbol{0} \,\middle|\, \textstyle\sum_i m_i = M \right\}$$

$$\mathcal{N}(\boldsymbol{m}) = \left\{ \boldsymbol{n} \geq \boldsymbol{O} \,\middle|\, \begin{array}{ll} S_h \sum_j n_{ij} + S_f m_i \leq \bar{S} & \forall i \\ \sum_i n_{ij} \geq L m_j & \forall j \\ \sum_i \sum_j n_{ij} = N & \end{array} \right\}$$

3. アルゴリズム

(1) 均衡解を求めるための課題

二次元空間 FO モデルでは，多様な複数都心パターンが均衡解として創発すると考えられる．これらのパターンを正確に表現するためには，空間解像度を上げなくてはならない．また，問題 [P1] の $Z_2(\boldsymbol{n})$ の未知変数である $\boldsymbol{n}$ の要素数のオーダーは K^2 と等しい．例えば，正方形型の空間を想定する場合，最低でも一辺あたり百等分程度の精度が求められ，地点数を $K = (10^2)^2$ 以上にする必要がある．$\boldsymbol{n}$ の要素数は $K^2 = 10^8$ であり，膨大な計算量および記憶容量が必要になってしまう．

この大規模問題を精度よく解くためには，アルゴリズムを工夫する必要がある（$K = 10^4$ の場合，$\boldsymbol{n}$ を倍精度変数に設定すると，1GB の記憶容量が必要であり，実用的とはいいがたい）．素朴なアルゴリズムで問題 [P1] を解く場合，精度の高い均衡解を得ることは困難である．

(2) 提案アルゴリズムとその考え方

(1) で挙げられた課題を解決するためには，$\boldsymbol{n}$ の要素のうちすべてを演算および記憶させずに済むアルゴリズムが必要である．そのために，まず問題 [P1] に Frank-Wolfe 法を適用する．この方法では，勾配ベクトルを求めるために，問題 [P1] を線形近似して得られる以下の補助問題を解く：

$$[\mathbf{P2}] \min_{\boldsymbol{x}^{(k)}, \boldsymbol{y}^{(k)}} - \sum_i \Big(\sum_j d_{ij} m_j^{(k)} \Big) x_i^{(k)} + Z_2(\boldsymbol{y}^{(k)})$$

$$\text{s.t. } \boldsymbol{x}^{(k)} \in \mathcal{M},\ \boldsymbol{y}^{(k)} \in \mathcal{N}(\boldsymbol{x}^{(k)})$$

ここで，$\boldsymbol{m}^{(k)}, \boldsymbol{n}^{(k)}$ は問題 [P1] の k 回目の計算ステップ得られた解である．

問題 [P2] では要素数が大きい $\boldsymbol{y}^{(k)}$ を含んでいるため，膨大な記憶容量が必要である．この問題を解決するために，$\boldsymbol{y}$ に関するサブ問題と $\boldsymbol{x}$ に関するマスター問題に分解（Benders Decomposition）する（k 回目を表す $^{(k)}$ を省略する）：

$$[\mathbf{P3}] \min_{\boldsymbol{x} \in \mathcal{M}} \Big[- \sum_i \Big(\sum_j d_{ij} m_j \Big) x_i + \min_{\boldsymbol{y} \in \mathcal{N}(\boldsymbol{x})} Z_2(\boldsymbol{y}|\boldsymbol{x}) \Big]$$

サブ問題は Hitchcock 型最適輸送問題であり，問題 [P4] とする．本研究では，問題 [P4] を効率的に解くためのアルゴリズムを提案する．基本理論によると，最適解の非零の要素の個数は制約条件の本数（$2K - 1$）と同数である．この性質を活用して，本研究では，非零の要素の要素番号と値のみ記憶させる．汎用の数理計画パッケージを用いた場合，アルゴリズムで演算および記憶させる未知変数は K^2 であるので，格段に計算量および記憶容量を削減できる．

4. 数値実験

(1) 実験条件

問題の規模を変化させたときに，補助問題を解くために必要な CPU 処理時間がどのように変化するのかを明らかにする．本実験では，問題の規模を $K = 1^2, 2^2, ..., 8^2 \times 100$ の 8 パターンとする．アルゴリズム性能を確認するために，本アルゴリズムで作成したプロ

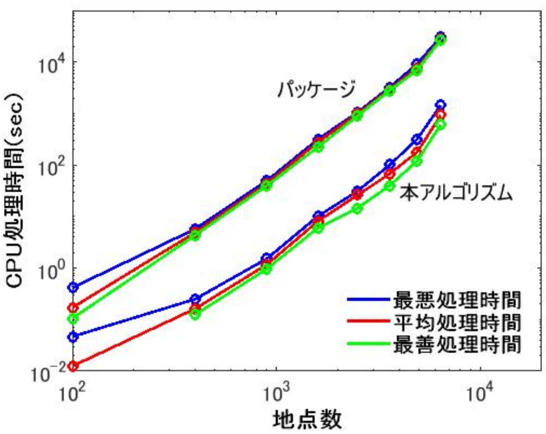
グラムと MATLAB の線形計画問題ソルバー「linprog」（アルゴリズムはレガシ内点法）を比較する．

上記の実験結果の頑健性を保証するために，パラメータ $(t, \tau, \boldsymbol{m}^{(1)})$ を変化させた実験を行う．パラメータ $\tau = 0.1$ に固定し，t を $t = 0.1, 0.3, 0.5, 0.7, 0.9$ の 5 パターン与える．また，初期値 $\boldsymbol{x}^{(1)}$ を 5 パターン与えることとする．

(2) 実験結果

図 1 は，問題の規模（横軸）と補助問題を解くために必要な CPU 処理時間（縦軸）の関係を示している．また，問題の規模ごとに，すべてのパラメータのケースでの最善処理時間，平均処理時間，最悪処理時間が示されている．この図から，次のことが読み取れる：

- 本アルゴリズムで作成したプログラムでは，実用的な時間で解くことができた．
- パッケージで解いた場合よりも CPU 処理時間が十分の一以下になった．



図–1 補助問題を解くために必要な CPU 処理時間

5. おわりに

本研究では，二次元空間 FO モデルと等価な最適化問題を効率的に解くためのアルゴリズムを提案し，補助問題を効率的に解けるのかどうか検証した．研究発表会で，全体問題を効率的に解けるのかどうか検証した結果と二次元空間 FO モデルのさまざまな均衡解の計算例を示す予定である．

参考文献

1) Fujita, M., Ogawa, H. : Multiple equilibria and structural transition of non-monocentric urban configurations, Regional science and urban economics, vol. 12, No.2 , pp. 161-196, 1982.
2) Osawa, M.,Akamatsu, T. Stochastically stable equilibria of the Fujita and Ogawa (1982) model. 第 30 回応用地域学会研究発表大会, 2016.

(2017. 7. 31 受付)

A DEVELOPMENT OF AN ALGORITHM FOR SOLVING THE FUJITA AND OGAWA (1982) MODEL IN A TWO-DIMENSIONAL SPACE

Katsuya AKIMOTO and Takashi AKAMATSU