




Su, C.J., and Judd, K.L.: Constrained optimization approaches to estimation of structural models, *Econometrica*, Vol. 80, pp. 2213-2230, 2012.

集中理論談話会 #15

2014/6/28(土)

D2 浦田 淳司



論文目次・概要

構造推定にあたって、NFXP法(Rust(1987))は、不動点の繰り返し計算を行うために計算負荷が高い。そこで、均衡制約条件付き数理計画問題(MPEC)を用いて等価な定式化を行い、計算負荷を低減させた手法を提案する。

1. Introduction
2. MPEC Approach to Estimation
3. Single-Agent Dynamic Discrete-Choice Models
 - 3.1 Single-Agent Dynamic Discrete-Choice Models
 - 3.2 Maximum-Likelihood Estimation
4. Monte Carlo Experiments
5. Conclusion

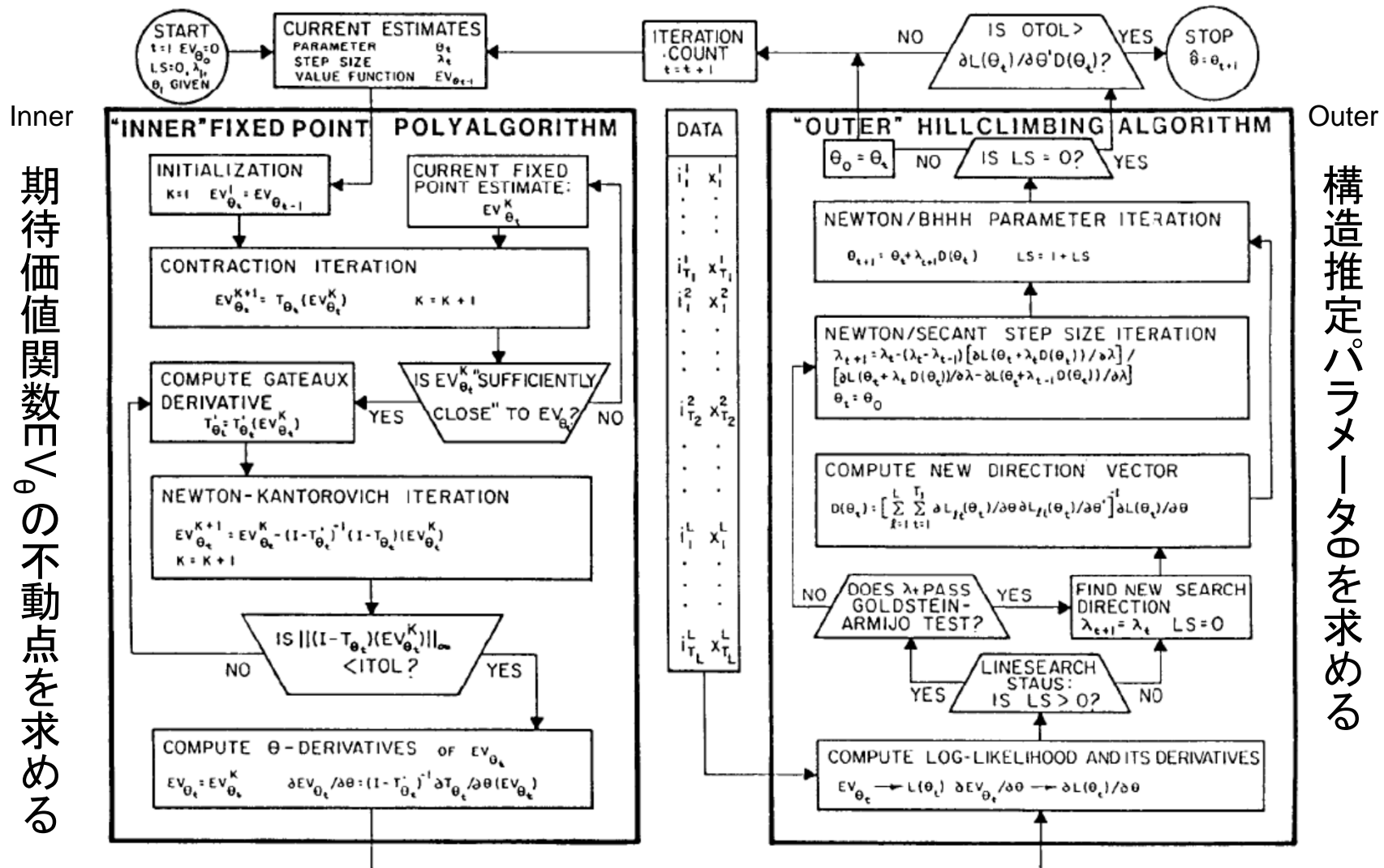
+ NPLとの比較

++ TRcの式(論文とは関係なし)

背景：NFXP法(構造推定アプローチ)

Rust, J.(1988) Statistical Models of Discrete Choice Processes, Transportation Research Part B, Vol. 22(2), pp. 125-158.

NESTED FIXED POINT MAXIMUM LIKELIHOOD ALGORITHM



θの繰り返し計算が負荷↓MPECの導入

期待値関数 EV_{θ} の不動点を求める

構造推定パラメータ θ を求める

均衡条件, 尤度の最大化

提案手法 (MPEC) を用いた推定法を説明 (2章)

観測可能な状態 x

選択結果 d

意思決定者 i ($1 \leq i \leq M$)

構造パラメータ θ : コスト, 利益などを示す

内生変数 σ : 意思決定者の政策関数. 構造パラメータ θ の関数となる.

σ は均衡条件を通じて, θ に拠る.

$$h(\theta, \sigma) = 0 \quad (1)$$

θ を所与としたときの, $\Sigma(\theta)$ を (1) 式を満たす σ の集合と定義する

$$\Sigma(\theta) := \{\sigma : h(\theta, \sigma) = 0\}$$

$\hat{\sigma}(\theta)$ は $\Sigma(\theta)$ の要素であり, 期待値関数とし, 一意に定まる.

また, $\Sigma(\theta)$ は θ と 1 対 1 対応する.

$L(\theta, \hat{\sigma}(\theta); X)$ を観測データ X の対数尤度とし, 尤度最大化パラメータは次と定義

$$\hat{\theta} = \operatorname{argmax}_{\theta} \frac{1}{M} \left\{ \max_{\hat{\sigma}(\theta) \in \Sigma(\theta)} L(\theta, \hat{\sigma}(\theta); X) \right\} \quad (3)$$

MPECによる定式化

NFXPアルゴリズムでは、次を繰り返す.

outer loop: 尤度関数を最大化する構造パラメータ θ を求める

inner loop: パラメータ θ を所与とし、すべての $\hat{\sigma}(\theta) \in \Sigma(\theta)$ の中から不動点解 $\hat{\sigma}(\theta)$ を求める

やりたいことは、

構造パラメータ θ を推定する際に、内生変数 σ が θ から算出されたものであればよい

⇒均衡制約条件(1)を用いた制約付き最適化問題として定義する

$$\max_{(\theta, \sigma)} \frac{1}{M} L(\theta, \sigma; X) \quad \text{subject to } h(\theta, \sigma) = 0 \quad (4)$$

定理1: (3)式と(4)式の等価性を示す

$\hat{\theta}$ は尤度最大化の(3)式で定義され、(4)式の解は $(\bar{\theta}, \bar{\sigma})$ とする.

$\hat{\sigma}^*(\theta) = \arg \max_{\hat{\sigma}(\theta)} L(\theta, \hat{\sigma}(\theta))$ と定義する.

また、 $L(\hat{\theta}, \hat{\sigma}^*(\hat{\theta})) = L(\bar{\theta}, \bar{\sigma})$ となる. このとき、 $\hat{\theta} = \bar{\theta}$ となる.

証明:

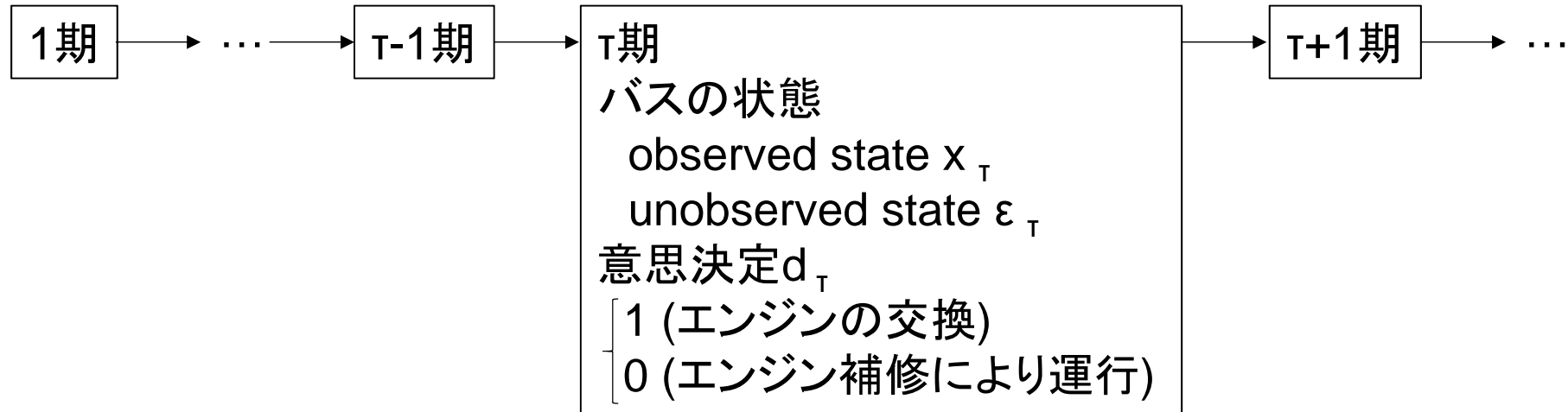
$(\bar{\theta}, \bar{\sigma})$ は(1)式を満足し、 $\bar{\sigma} \in \Sigma \bar{\theta}$ であるため、(3)式より、 $L(\hat{\theta}, \hat{\sigma}^*(\hat{\theta})) \geq L(\bar{\theta}, \bar{\sigma})$.

$\hat{\sigma}^*(\hat{\theta}) \in \Sigma(\hat{\theta})$ であるため、制約条件(1)を満たし、(4)式より、 $L(\hat{\theta}, \hat{\sigma}^*(\hat{\theta})) \leq L(\bar{\theta}, \bar{\sigma})$.

モデルが特定され、解が1つであるため、 $\hat{\theta} = \bar{\theta}$ となる.

NFXP法：Bus-Engine Replacement Problem

Rust, J.(1987) のバスエンジン補修/交換の問題を例にして, NFXP法を説明(3章)



ある期の効用関数 u

$$u(x, d, \varepsilon; \theta_1, RC) = v(x, d; \theta_1, RC) + \varepsilon(d)$$

$$v(x, d; \theta_1, RC) = \begin{cases} -c(x; \theta_1), & \text{if } d = 0 \\ -RC - c(0; \theta_1), & \text{if } d = 1 \end{cases}$$

c : 補修コスト

θ_1 : 補修パラメータ

RC : 交換コスト

θ_2, θ_3 : 状態推移パラメータ

状態 x の推移確率 (一次マルコフ性を仮定)

$$p(x_{t+1}, \varepsilon_{t+1} | x_t, \varepsilon_t, d_t; \theta_2, \theta_3)$$

NFXP法：将来期待効用の最大化

将来の価値関数は、時間割引を考慮した効用の最大化となる

$$V(x_t, \varepsilon_t) = \max_{\{d_t, d_{t+1}, d_{t+2}, \dots\}} \mathbb{E} \left[\sum_{\tau=t}^{\infty} \beta^{\tau-t} u(x_\tau, d_\tau, \varepsilon_\tau; \theta_1, \text{RC}) \right] \quad (6)$$

時間割引率 $\beta \in (0, 1)$

最適意思決定は無限期間先までを考慮しており、期によらず一定である

$$V(x, \varepsilon) = \max_d \left\{ v(x, d; \theta_1, \text{RC}) + \varepsilon(d) + \beta \int_{x'} \int_{\varepsilon'} V(x', \varepsilon') p(x', \varepsilon' | x, \varepsilon, d; \theta_2, \theta_3) dx' d\varepsilon' \right\} \quad (7)$$

次期状態 (x', ε')

↑ 今期と次期の状態により、価値関数を定義

NFXP法：期待価値関数・選択確率

仮定：推移確率の条件つき独立性(CI)

$$p(x', \varepsilon' | x, \varepsilon, d; \theta_2, \theta_3) = p_2(\varepsilon' | x'; \theta_2) p_3(x' | x, d; \theta_3)$$

CIの仮定を用いて，期待価値関数を次で定義する．

$$EV(x) = \int_{\varepsilon} V(x, \varepsilon) p_2(\varepsilon | x; \theta_2) d\varepsilon$$

選択肢の期待価値関数((7)式から ε を除いて)

$$EV(x, d) = v(x, d; \theta_1, RC) + \varepsilon(d) + \beta \int_{x'} EV(x') p_3(x' | x, d; \theta_3) dx'$$

これを期待価値関数の定義式に戻す

$$EV(x) = \int_{\varepsilon'} \max_d \{EV(x, d)\} p_2(\varepsilon' | x; \theta_2) d\varepsilon' \quad (8)$$

ε を極値分布とし，ロジット型の条件付き選択確率を導出($\theta = (RC, \theta_1, \theta_3)$ がパラメータ)

$$P(d | x; \theta) = \frac{\exp[v(x, d; \theta_1, RC) + \beta EV(x, d)]}{\sum_{d' \in \{0,1\}} \exp[v(x, d'; \theta_1, RC) + \beta EV(x, d')]} \quad (9)$$

NFXP法：Fixed-Point Solution

(7)式と極値分布により, Fixed-Point equationは次となる

$$\begin{aligned} EV(x, d) = & \int_{x'=0}^{\infty} \log \left\{ \sum_{d' \in \{0,1\}} \exp[\nu(x', d'; \theta_1, RC) + \beta EV(x', d')] \right\} \\ & \times p_3(dx'|x, d, \theta_3). \end{aligned} \quad (10)$$

今期の状態 x を K 分割, 次期の状態 x' を J 分割して, 表記.

$$p_3(x'|x_k, d, \theta_3) = \begin{cases} \Pr\{x' = \hat{x}_{k+j} | \theta_3\}, & \text{if } d = 0 \\ \Pr\{x' = \hat{x}_{1+j} | \theta_3\}, & \text{if } d = 1 \end{cases} \quad (11)$$

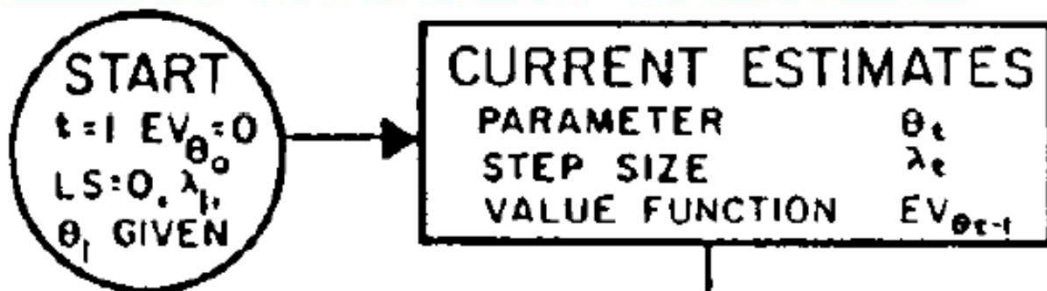
(11)式に代入

$$\begin{aligned} EV(\hat{x}_k, d) = & \sum_{j=0}^J \log \left\{ \sum_{d' \in \{0,1\}} \exp[\nu(x', d'; \theta_1, RC) + \beta EV(x', d')] \right\} \\ & \times p_3(x' | \hat{x}_k, d, \theta_3). \end{aligned} \quad (12)$$

※ この式が定義されている $[EV = [EV(\hat{x}_1, 0), \dots, EV(\hat{x}_K, 0), EV(\hat{x}_1, 1), \dots, EV(\hat{x}_K, 1)]]$

で, 結局この式に落ち着く. 不動点の求解.

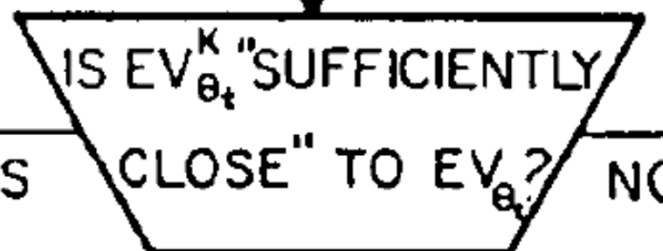
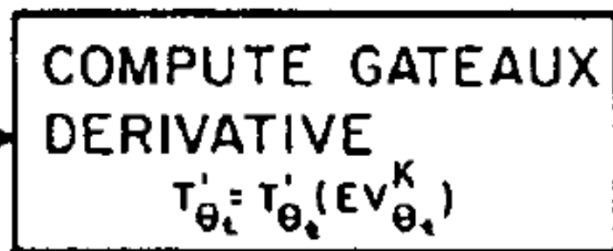
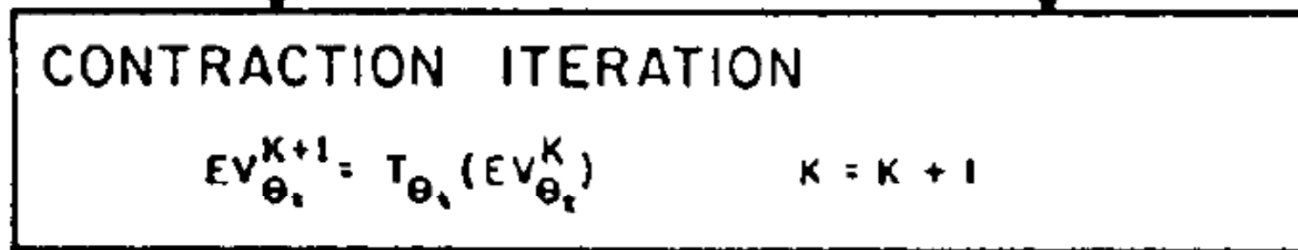
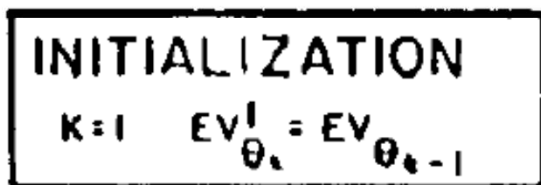
$$EV = T(EV, \theta) \quad (13)$$



INNER "FIXED POINT POLYALGORITHM"

Inne

Outer



θ_tの繰り返し計算が負荷↓MPECの導入

期待効用E_tの不動点を求める

構造推定パラメータθを定める

NFXP法：最尤法によるパラメータ推定

バス会社*i*の尤度 $\ell_i(X^i; \theta) = \prod_{t=2}^T P(d_t^i | x_t^i; \theta) p_3(x_t^i | x_{t-1}^i, d_{t-1}^i; \theta_3)$

M社の全体尤度 $\ell(\theta) = \prod_{i=1}^M \ell_i(X^i; \theta) = \prod_{i=1}^M \prod_{t=2}^T P(d_t^i | x_t^i; \theta) p_3(x_t^i | x_{t-1}^i, d_{t-1}^i; \theta_3)$ (14)

対数尤度 $L(\theta) = \log \ell(\theta)$

$$= \sum_{i=1}^M \sum_{t=2}^T \log[P(d_t^i | x_t^i; \theta)] + \sum_{i=1}^M \sum_{t=2}^T \log[p_3(x_t^i | x_{t-1}^i, d_{t-1}^i; \theta_3)] \quad (15)$$

対数尤度最大化により, $\theta = (RC, \theta_1, \theta_3)$ を求める

$$\max_{\theta} \frac{1}{M} L(\theta) \quad (16)$$

MPECによる定式化

(13)式, (16)式を等価な均衡制約条件付き最適化問題と定式化.
(等価であることの証明は定理1より)

$$\begin{aligned} \max_{(\theta, EV)} \quad & \frac{1}{M} \mathcal{L}(\theta, EV) \\ \text{subject to} \quad & EV = T(EV, \theta) \end{aligned} \quad (18)$$

NFXPが不動点の算出プロセスを毎回解いているのに比べて,
Bellmanの等式を一度評価すればよいので, 計算負荷は小さい.

計算精度の比較

β	Implementation	True values:	Parameters					MSE	
			RC	θ_{11}	θ_{30}	θ_{31}	θ_{32}		θ_{33}
0.975	MPEC/AMPL	Mean	12.212	2.607	0.0943	0.4473	0.4454	0.0127	3.111
		Std. dev.	(1.613)	(0.500)	(0.0036)	(0.0057)	(0.0060)	(0.0015)	-
	MPEC/MATLAB	Mean	12.212	2.607	0.0943	0.4473	0.4454	0.0127	3.111
		Std. dev.	(1.613)	(0.500)	(0.0036)	(0.0057)	(0.0060)	(0.0015)	-
	NFXP/MATLAB	Mean	12.213	2.606	0.0943	0.4473	0.4445	0.0127	3.123
		Std. dev.	(1.617)	(0.500)	(0.0036)	(0.0057)	(0.0060)	(0.0015)	-
0.980	MPEC/AMPL	Mean	12.134	2.578	0.0943	0.4473	0.4455	0.0127	2.857
		Std. dev.	(1.570)	(0.458)	(0.0037)	(0.0057)	(0.0060)	(0.0015)	-
	MPEC/MATLAB	Mean	12.134	2.578	0.0943	0.4473	0.4455	0.0127	2.857
		Std. dev.	(1.570)	(0.458)	(0.0037)	(0.0057)	(0.0060)	(0.0015)	-
	NFXP/MATLAB	Mean	12.139	2.579	0.0943	0.4473	0.4455	0.0127	2.866
		Std. dev.	(1.571)	(0.459)	(0.0037)	(0.0057)	(0.0060)	(0.0015)	-
0.985	MPEC/AMPL	Mean	12.013	2.541	0.0943	0.4473	0.4455	0.0127	2.140
		Std. dev.	(1.371)	(0.413)	(0.0037)	(0.0057)	(0.0060)	(0.0015)	-
	MPEC/MATLAB	Mean	12.013	2.541	0.0943	0.4473	0.4455	0.0127	2.140
		Std. dev.	(1.371)	(0.413)	(0.0037)	(0.0057)	(0.0060)	(0.0015)	-
	NFXP/MATLAB	Mean	12.021	2.544	0.0943	0.4473	0.4455	0.0127	2.136
		Std. dev.	(1.368)	(0.411)	(0.0037)	(0.0057)	(0.0060)	(0.0015)	-
0.990	MPEC/AMPL	Mean	11.830	2.486	0.0943	0.4473	0.4455	0.0127	1.880
		Std. dev.	(1.305)	(0.407)	(0.0036)	(0.0057)	(0.0060)	(0.0015)	-
	MPEC/MATLAB	Mean	11.830	2.486	0.0943	0.4473	0.4455	0.0127	1.880
		Std. dev.	(1.305)	(0.407)	(0.0036)	(0.0057)	(0.0060)	(0.0015)	-
	NFXP/MATLAB	Mean	11.830	2.486	0.0943	0.4473	0.4455	0.0127	1.880
		Std. dev.	(1.305)	(0.407)	(0.0036)	(0.0057)	(0.0060)	(0.0015)	-
0.995	MPEC/AMPL	Mean	11.819	2.492	0.0942	0.4473	0.4455	0.0127	1.892
		Std. dev.	(1.308)	(0.414)	(0.0036)	(0.0057)	(0.0060)	(0.0015)	-
	MPEC/MATLAB	Mean	11.819	2.492	0.0942	0.4473	0.4455	0.0127	1.892
		Std. dev.	(1.308)	(0.414)	(0.0036)	(0.0057)	(0.0060)	(0.0015)	-
	NFXP/MATLAB	Mean	11.819	2.492	0.0942	0.4473	0.4455	0.0127	1.892
		Std. dev.	(1.308)	(0.414)	(0.0036)	(0.0057)	(0.0060)	(0.0015)	-

^aFor each β , there are 250 replications. Standard deviations are reported in parentheses. MSE is calculated by summing over all structural parameters.

- ・250回計算の平均をとり、パラメータの平均、分散を比較
- ・ β が大きい場合は、NFXPと同じ結果であるなど、両者の精度の差はほぼない

計算時間の比較

- ・MPECのほうがNFXPよりも計算時間は短い(AMPLで180倍以上, MATLABで3倍以上)

β	Implementation	Runs Converged (out of 1250 runs)	CPU Time (in sec.)	# of Major Iter.	# of Func. Eval.	# of Contraction Mapping Iter.
0.975	MPEC/AMPL	1240	0.13	12.8	17.6	–
	MPEC/MATLAB	1247	7.90	53.0	62.0	–
	NFXP	998	24.60	55.9	189.4	134,748
0.980	MPEC/AMPL	1236	0.15	14.5	21.8	–
	MPEC/MATLAB	1241	8.10	57.4	70.6	–
	NFXP	1000	27.90	55.0	183.8	162,505
0.985	MPEC/AMPL	1235	0.13	13.2	19.7	–
	MPEC/MATLAB	1250	7.50	55.0	62.3	–
	NFXP	952	43.20	61.7	227.3	265,827
0.990	MPEC/AMPL	1161	0.19	18.3	42.2	–
	MPEC/MATLAB	1248	7.50	56.5	65.8	–
	NFXP	935	70.10	66.9	253.8	452,347
0.995	MPEC/AMPL	965	0.14	13.4	21.3	–
	MPEC/MATLAB	1246	7.90	59.6	70.7	–
	NFXP	950	111.60	58.8	214.7	748,487

^aFor each β , we use five starting points for each of the 250 replications. CPU time, number of major iterations, number of function evaluations and number of contraction mapping iterations are the averages for each run.

+ NPL(擬似最尤推定法)との比較

Aguirregabiria and Mira(2002)

TABLE I
MONTE CARLO EXPERIMENT

Experiment design	
Model:	Bus engine replacement model (Rust)
Parameters:	$\theta_0 = 10.47; \theta_1 = 0.58; \beta = 0.9999$
State space:	201 cells
Number observations:	1000
Number replications:	1000
Initial probabilities:	Kernel estimates

Monte Carlo distribution of MLE
(In parenthesis, percentages over the true value of the parameter)

	θ_0	θ_1
Mean Absolute Error:	2.08 (19.9%)	0.17 (29.0%)
Median Absolute Error:	1.56 (14.9%)	0.13 (22.7%)
Std. dev. estimator:	2.24 (21.4%)	0.16 (26.9%)
Policy iterations (avg.):	6.2	

Monte Carlo distribution of PI estimators (relative to MLE)
(All entries are 100* (K-PI statistic-MLE statistic)/MLE statistic)

Parameter	Statistics	Estimators		
		1-PI	2-PI	3-PI
θ_0	Mean AE	4.7%	1.6%	0.3%
	Median AE	14.2%	0.2%	-0.3%
	Std. dev.	6.8%	1.2%	0.3%
θ_1	Mean AE	18.7%	1.5%	0.2%
	Median AE	25.1%	0.7%	0.6%
	Std. dev.	11.0%	1.3%	0.2%

- ・計算速度はパラメータが4つの場合は、NPLは9倍速い
- ・MPEC型はNFPXと等価であるが、NPLは漸近等価.

Nested Pseudo Likelihood Algorithm (NPL):

Let $\hat{\theta}_f$ be an estimate of θ_f . Start with an initial guess for the conditional choice probabilities, $P^0 \in [0, 1]^{MJ}$. At iteration $K \geq 1$, apply the following steps:

Step 1: Obtain a new pseudo-likelihood estimate of α , α^K , as

$$(11) \quad \alpha^K = \arg \max_{\alpha \in \Theta} \sum_{i=1}^n \ln \Psi_{(\alpha, \hat{\theta}_f)}(P^{K-1})(a_i | x_i)$$

where $\Psi_{\theta}(P)(a|x)$ is the element (a, x) of $\Psi_{\theta}(P)$.

Step 2: Update P using the 'arg max' from step 2, i.e.

$$(12) \quad P^K = \Psi_{(\alpha^K, \hat{\theta}_f)}(P^{K-1}).$$

Iterate in K until convergence in P (and α) is reached.

まとめ

まとめ

- ・構造モデルの推定にMPECを用いる手法を提案
- ・シングルエージェントの動的離散選択モデルで実証
- ・標準的なソフトウェアで計算可能であり, NFXPよりも早い

今後の課題

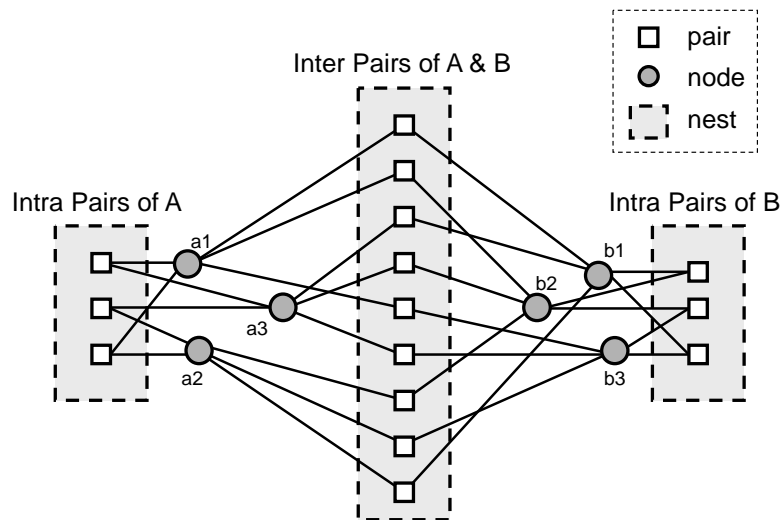
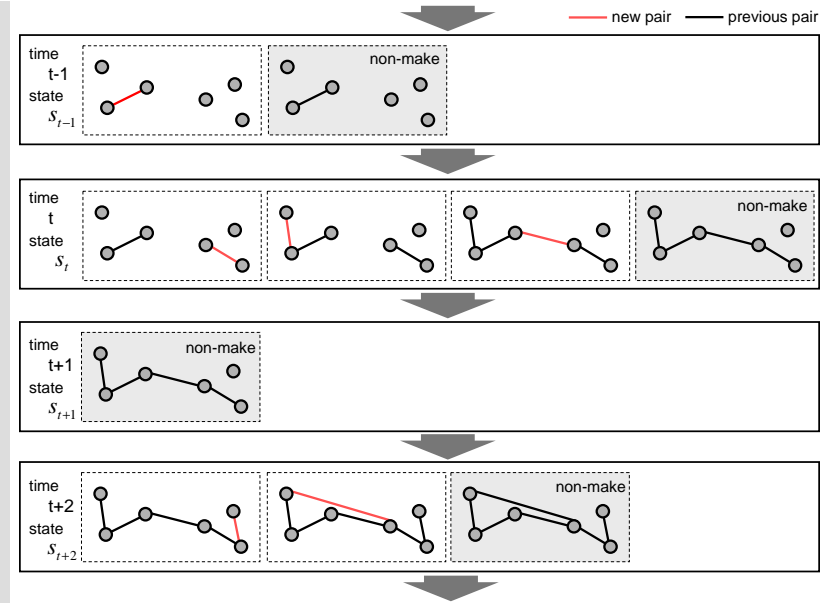
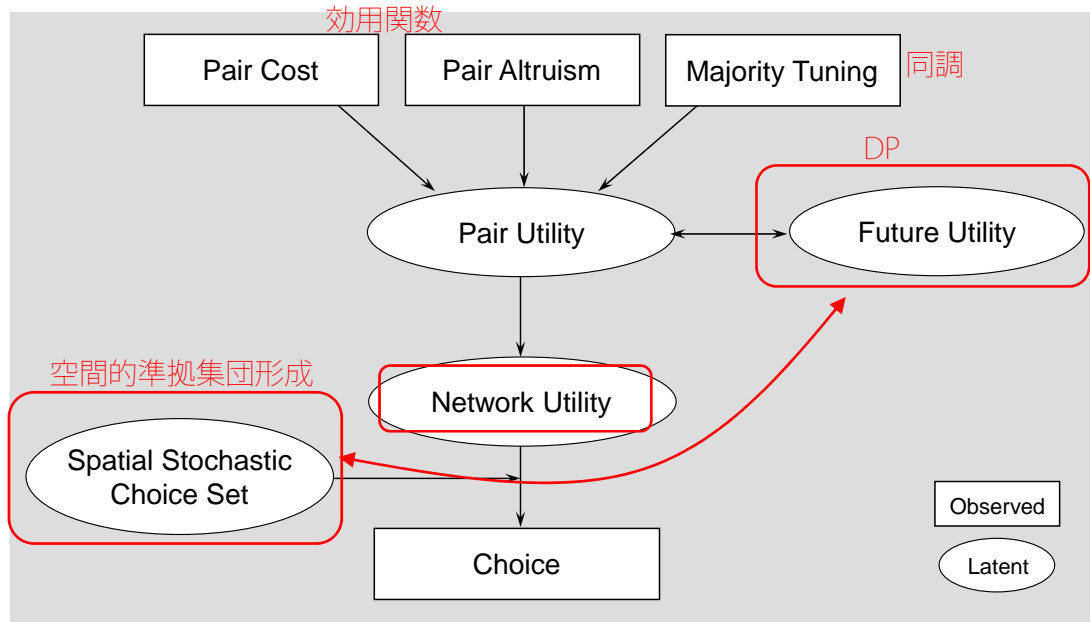
- ・動的離散選択ゲームにも適用可能かを検証する

所感

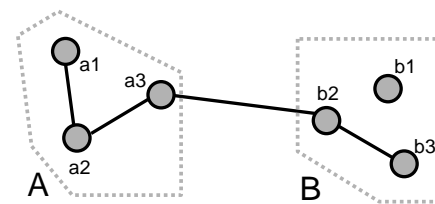
- ・コロンブスの卵的？
- ・等価性と計算性の両方をカバー

つづく？

TRc の式



b) Dividing Intra and Inter Pairs by basic groups



a) Dividing Basic Group

TR c の式2 (Tuning Effect)

選択確率 (時刻t, Pair Set g)

$$P(d_t | S_{g,t}; \theta) = P(d_t | L, S_{g,t}; \theta) P(L, S_{g,t}; \theta) = \frac{\exp((u(S_{g,t}) + \beta v(S_{g,t}))/\sigma)}{R_{L,t}} \frac{\exp(\sigma \ln R_{L,t})}{\sum_{L'} \exp(\sigma \ln R_{L',t})}$$

$$\begin{aligned} \text{Network Utility } S_{g,t}(d_t) &= m^{l_{g,t}} \left(\frac{1}{N_g} \sum_{ij \in g} \theta_1 |x_{i,t}^{\text{dam}} - x_{j,t}^{\text{dam}}| \right) + \delta_g^{\text{intra}} \theta_3 \ln k_{g,t}^{\text{intra}} + \delta_g^{\text{inter}} \theta_4 \ln k_{g,t}^{\text{inter}} \\ &= O_1 + \theta' \ln k_{g,t} \end{aligned}$$

$$\begin{aligned} \frac{\exp((u(S_{g,t}) + \beta v(S_{g,t}))/\sigma)}{R_{L,t}} &= \exp(u(S_{g,t})/\sigma) \frac{\exp(\beta v(S_{g,t})/\sigma)}{R_{L,t}} \\ &= [\exp(\theta' \ln k_{g,t} + O_1)]^{1/\sigma} O_2 \\ &= [\exp(\theta' \ln k_{g,t}) \exp(O_1)]^{1/\sigma} O_2 \\ &= [\exp(\ln(k_{g,t}^{\theta'}))]^{1/\sigma} O_3 O_2 \\ &= (k_{g,t}^{\theta'})^{1/\sigma} O_3 O_2 = (k_{g,t})^{\theta'/\sigma} O_3 O_2 \end{aligned}$$

適応度モデル
ノード v_i のリンク形成確率

$$P(v_i) = \frac{\eta_i k_i}{\sum_j \eta_j k_j}$$

ご清聴ありがとうございました.