^{応用一般均衡分析入門} 第6章:一般均衡モデルのチェック 方法*

武田 史郎†

Date: 2024/01/28, Version 4.0

目次

1	導入・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	2
2	マクロの説明・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	2
3	均衡価格の0次同次性(ニュメレール)のチェック ・・・・・・・・・・・・・・・・・・・・・・・・	3
3.1	プログラムの説明 ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	3
3.2	結果・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	10
3.3	まとめ(価格についての 0 次同次性のチェック) ・・・・・・・・・・・・・・・・・・・・・・	13
4	ワルラス法則のチェック ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	13
4.1	プログラムとシナリオ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	13
4.2	結果・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	15
4.3	ワルラス法則が満たされていないケース・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	15
4.4	まとめ(ワルラス法則のチェック) ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	17
5	外生変数の比例変化によるチェック ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	17
5.1	シナリオ ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	17
5.2	結果・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	18
5.3	まとめ ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	18
6	簡単な政策シミュレーション ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	19
6.1	シナリオ ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	19
6.2	結果・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	20
6	.2.1 シミュレーション結果・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	20
7	終わりに・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	21
参考了	文献 ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	22

*このファイルの配布場所: https://shirotakeda.github.io/ja/research-ja/cge-howto.html

[†]所属:京都産業大学経済学部. Website: https://shirotakeda.github.io/ja/

1 導入

今回の内容について。

- 第5章で一般均衡モデルを GAMS で表現して解いた。この第6章では第5章のモデルを チェックするという作業、及び簡単な政策のシミュレーションをおこなう。
- CGE 分析で利用するモデルは規模が大きく、変数も式も多くなる。そのようなモデルを表す プログラムは複雑になる。そのため自分の意図通りのモデルをスムーズにプログラムで表現で きるということは少なく、プログラムを記述する段階で多くの間違いが入ってしまうことが 多い。
- そこで自分の意図したモデルを作成できたかどうかをチェックする作業が必要になる。これに は様々なアプローチがあるが、第6章では次の3つのチェックをおこなう。
 - 。 均衡価格の0次同次性(ニュメレール)に関するチェック
 - 。 ワルラス法則に関するチェック
 - 。 外生変数の比例変化によるチェック

3 つとも我々のモデルにおいて成り立っていなければならない性質が実際に成り立っているか どうかを確認することで、モデル、データ、プログラムの正しさをチェックするという方法で ある。

- また、政策分析の簡単な例として、次の2つのシミュレーションをおこなう。
 - 。 一部の部門に対して 10 %の消費税を導入するシミュレーション
 - 。 3 つの部門の労動投入に 20 %の課税を導入するシミュレーション
- モデル自体の説明は第5章でおこなっているので、そちらを参照して欲しい。

2 マクロの説明

この第 6 章で利用するプログラムでは、GAMS の「マクロ」という機能を利用している。モデ ルのチェックの前にまずマクロとはどのようなものかを簡単に説明しておく。マクロとは、ある一 定の表現(手順)を何度も利用するための機能である。

マクロの基本的な書式は次の通りである。

\$macro macro_name macro_body

マクロは \$macro 命令によって定義する。macro_name の部分がマクロの名前になり、 macro_body のところに記述されたプログラムがマクロの中身になる。例えば、以下のプログラム では ratio という名前のマクロを定義している。

```
parameter a, b, c;
a = 1;
b = 2;
$macro ratio(x,y) x / y
c = ratio(a,b);
display c;
```

x と y はマクロの引数である。マクロはプログラムを実行する際には中身が展開される。よって、c に ratio(a,b) の値を代入する部分は次のように展開されて実行される。

c = a / b;

つまり、ratio(a,b) と書けば、a / b と書いたときと同じになるということである。プログラ ムを書く際、煩雑で複雑な表現を何度も利用するというときがよくある。そのような場合には、そ の表現をマクロに定義することでプログラムを簡潔にすることができる。非常に便利な機能である ので、利用法を覚えておくのがよい。マクロの使い方について詳しくは "Macros in GAMS" を見て 欲しい。

3 均衡価格の0次同次性(ニュメレール)のチェック

第2章において説明したように、我々が考える一般均衡モデルでは、均衡価格の0次同次性という性質が成り立った。また、その性質を基に、ある財をニュメレールとしてモデルを解くという方法が用いられた。この均衡価格の0次同次性が実際に成り立っているかをチェックする。

3.1 プログラムの説明

ge_zero_h_check.gms はニュメレールに関するチェックをおこなうプログラムである。以下、 内容を説明していく。

```
* ge_sample_dual.gms を読み込む。
```

\$include ge_sample_dual.gms

ge_zero_h_check.gms ではモデルには第5章の ge_sample_dual.gms のモデルをそのまま利用 する。よって、まずここで ge_sample_dual.gms をそのまま読み込んでいる。\$include はファイ ルを読み込むという命令である。実行の際にここで「282 Unable to open include file」というよう なエラーが生じたら、読み込むファイルが見つからないということである¹⁾。

parameter
 results 結果を表示するためのパラメータ
 results_pc 変化率を表示するためのパラメータ (%)

;

ここではまず results と results_pc というパラメータを宣言している。これは計算結果を入 れておくためのパラメータである。

```
$macro out_resutls(sc) \
   results("u",sc) = u.1; \setminus
   results("y_agr",sc) = y.l("agr"); \
   results("y_man",sc) = y.l("man"); \
   results("y_ser",sc) = y.l("ser"); \
   results("c_agr",sc) = d.l("agr"); \
   results("c_man",sc) = d.l("man"); \
   results("c_ser",sc) = d.l("ser"); \
   results("p_man",sc) = p.1("man"); \
   results("pf_lab",sc) = p_f.l("lab"); \
   results("m",sc) = m.l; \
   results("p_man-p_agr",sc) = p.1("man") / p.1("agr"); \
   results("pf_lab-p_agr",sc) = p_f.l("lab") / p.l("agr"); \
   results("m-p_agr",sc) = m.l / p.l("agr"); \
   results("p_man-pf_lab",sc) = p.l("man") / p_f.l("lab"); \
   results("pf_lab-pf_lab",sc) = p_f.l("lab") / p_f.l("lab"); \
    results("m-pf_lab",sc) = m.l / p_f.l("lab");
```

ここで out_results というマクロを定義している²⁾。これは計算結果の様々な値を results と いうパラメータに代入する動作を一つのマクロとしてまとめたものである。この out_results と いうマクロは sc という引数を持っているが、sc には後にシナリオを表す集合を入れる。

行末の「バックスラッシュ(\)」は記述が次の行に継続しているということを表す記号である。 上のプログラムは合計 10 数行あるが、それをまとめて一つの out_results というマクロとして定 義するためにバックシュラッシュを利用している。このように一度 out_results を定義すれば、 後は out_results とプログラムに書くことで、上の 10 数行のプログラムを実行するのと同じ動作 をすることになる。

以下、マクロを使って results に代入している値を説明しておく。

results("u",sc) = u.1; \

¹⁾ その場合、GAMS IDE を利用しているのなら、タイトルバーに表示されているプロジェクトフォルダにファイルが あるかどうかを確認する。他の場合は、2 つの GMS ファイルが同じフォルダにあるかを確認する。

²⁾ バックスラッシュは場合によっては¥マークに表示される。

これは効用の水準を表している。変数のレベル値を代入するので、どの変数にも.1 を付けて いる。

```
results("y_agr",sc) = y.l("agr"); \
results("y_man",sc) = y.l("man"); \
results("y_ser",sc) = y.l("ser"); \
```

一番上は AGR 部門の生産量である。他の2行は MAN、SER の生産量である。

```
results("c_agr",sc) = d.l("agr"); \
results("c_man",sc) = d.l("man"); \
results("c_ser",sc) = d.l("ser"); \
```

一行目は AGR 財の消費量である。このモデルでは中間投入もあるので、生産量 ≠ 消費量である。他の 2 行は MAN、SER の消費量である。

results("p_man",sc) = p.l("man"); \

これは MAN の価格である。p("man") のそのままの値であるので、以下ではこの価格を便宜 上「絶対価格」と呼ぶことにする。ただし、この価格は AGR をニュメレールとし、その価格を 1 と設定したときの p("man") の価格であるので、厳密には AGR の価格を 1 に固定した上での p("man")/p("agr") という相対価格でもある。

```
results("pf_lab",sc) = p_f.l("lab"); \
results("m",sc) = m.l; \
```

これは労働の価格と所得水準である。これも以下では絶対価格・所得と呼ぶ。

results("p_man-p_agr",sc) = p.l("man") / p.l("agr"); \

これは AGR で測った MAN の相対価格であり、MAN 一つが AGR 何個分の価値を持つかを表 している。元々、AGR をニュメレールにとりその価格を 1 としているので、この相対価格は単 なる p.1("man") と同じ値をとる。しかし、ニュメレールを変更した場合には、p.1("man") と p.1("man")/p.1("agr") の値は変わってくる。実際、以下ではニュメレールを変更するシミュ レーションをおこない、どう変わるかをチェックする。

```
results("pf_lab-p_agr",sc) = p_f.l("lab") / p.l("agr"); \
results("m-p_agr",sc) = m.l / p.l("agr"); \
results("p_man-pf_lab",sc) = p.l("man") / p_f.l("lab"); \
results("pf_lab-pf_lab",sc) = p_f.l("lab") / p_f.l("lab"); \
results("m-pf_lab",sc) = m.l / p_f.l("lab");
```

以上は全て相対価格である。p("agr") で割っているのは AGR で測った相対価格、p_f("lab") で割っているのは労働で測った相対価格である。最初の二つは AGR で測った相対価格、下の2つ は労働で測った相対価格である。最後は労働で測った相対所得(実質所得)である。

それではプログラムの続きを説明していこう。

```
set var 結果表示用の集合 /
u, y_agr, y_man, y_ser, c_agr, c_man, c_ser, p_man, pf_lab, m,
p_man-p_agr, pf_lab-p_agr, m-p_agr, p_man-pf_lab,
pf_lab-pf_lab, m-pf_lab /;
```

ここでは計算結果をまとめるために利用する集合 var を定義している。

set	sce	計算する	るシナリオ /
		sce1	デフォールトの均衡
		sce2	資本の賦存量を2倍にした均衡
		sce3	AGR の価格を 2 倍にした均衡
		sce4	ニュメレールを労動に変更したときの均衡
		sce5	SCE2+ニュメレールを労動に変更したときの均衡
		/;	
display	sce;		

ここではシナリオを表すための集合 sce を定義している。sce の要素は sce1~sce5 の 5 つであ り、以下ではシナリオとして 5 つを考える。

[シナリオ sce1]

これはデフォールトの均衡である。ge_sample_dual.gms のプログラムと同じ均衡である。これを比較の基準とする。

[シナリオ sce2]

これは資本の賦存量を2倍にしたときの均衡である。このシナリオでは各財の生産量、価格がど

う変わるかをチェックする。

[シナリオ sce3]

これは AGR の価格を2倍にした均衡である。元々、AGR をニュメレールとして、その価格を 1に固定していた。ここではその価格を2に変更してみる。価格について0次同次性が成り立つモ デルであるので、AGR の価格を(1から)2に変更すれば、数量変数や相対価格は全く変わらず、 単に絶対価格が全て2倍になるだけのはずである。このシナリオではこれをチェックする。

[シナリオ sce4]

ニュメレールを労働に変更する。つまり、労働の価格を1に固定して計算する。ニュメレールの 変更によって、均衡がどう変化するかをチェックする。価格について 0 次同次性が満たされるモ デルではニュメレールの選択は数量変数の値、相対価格の値には全く影響を与えないはずである。 それを確認する。

[シナリオ sce5]

sce4 の設定で、資本の賦存量を 2 倍にするというシミュレーションをおこなう。ニュメレール を変更することで、sce2 の場合と結果がどう変わるかをチェックする。0 次同次性が満たされる なら、sce2 のケースとは絶対価格の値は変わってくるが、相対価格、数量変数の値は同じになる はずである。

以下は各シナリオを解くプログラムである。

ここではデフォールトのシナリオを解いている。特に何の変更もせず、solve 命令によりモデル を解いているだけである。out_results("sce1") という命令によって、マクロ our_results の引 数 sc の部分に "sce1" が入る形で、マクロが実行されることになる。

```
* SCE2: 資本の賦存量を2倍にした均衡
* 2倍に増加
v_bar("cap") = v_bar0("cap") * 2;
solve ge sample_dual using mcp;
```

```
    * マクロの実行
    out_resutls("sce2");
    * 元の値に戻す。
    v_bar("cap") = v_bar0("cap");
```

ここは sce2 を解く部分である。資本の賦存量を2倍にして、モデルを解いている。解いた後は 先ほどと同様マクロを実行し、さらにここでは資本の賦存量を元々の値に戻している。

```
* ------
* SCE3: AGR の価格を2倍にした均衡
p.fx("agr") = 2;
solve ge_sample_dual using mcp;
out_resutls("sce3");
p.fx("agr") = 1;
```

AGR の価格は元々 1 に固定されているが、この部分ではそれを 2 に変更してモデルを解いている。後は同様である。

```
* ------* SCE4: ニュメレールを労動に変更したときの均衡
p.up("agr") = inf;
p.lo("agr") = 0;
p_f.fx("lab") = 1;
solve ge_sample_dual using mcp;
out_resutls("sce4");
```

ニュメレールを AGR から労働に変更している。具体的には、これまで 1 に固定していた p("agr") を自由に動くように変更し(ただし、下限は 0 としている)、代わりに p_f("lab") を 1 に固定している。

```
* -----* SCE5: sce4+資本の賦存量を2倍にした均衡
p.up("agr") = inf;
p.lo("agr") = 0;
```

```
p_f.fx("lab") = 1;
v_bar("cap") = v_bar0("cap") * 2;
solve ge_sample_dual using mcp;
out_resutls("sce5");
v_bar("cap") = v_bar0("cap");
```

ここでは、ニュメレールを労働にした上で、sce2 と同様のシミュレーションをおこなっている。

ここは計算結果をまとめている部分である。results に変数の値を代入したが、それから変数の 変化率(sce1 の値から何%変化したか)を求め、results_pc に代入している。

execute_unload "ge_zero_h_check.gdx", results, results_pc;

ここはパラメータ results と results_pc の値を ge_zero_h_check.gdx という GDX ファイル にエクスポートしている。execute_unload という命令は以下のような書式を持つ。

execute_unload "GDX ファイル名", 出力するもの1, 出力するもの2;

"GDX ファイル名"の部分に出力する GDX ファイルの名前を指定する。そして、その後に、出 力するものを列挙する(複数あるときはカンマで区切る)。出力可能なのは、パラメータ、変数、

```
集合などである。
```

execute_unload 命令については詳しくは "GAMS Data eXchange (GDX)" を参照して欲しい。

```
* $onecho ~ $offecho の間の部分を temp.txt というファイルの中に出力
$onecho > temp.txt
i=ge_zero_h_check.gdx o=ge_zero_h_check.xlsx epsout=0
par=results rng=results!A1 rdim=1 cdim=1
par=results_pc rng=results_pc!A1 rdim=1 cdim=1
$offecho
* temp.txt の中身を引数として gdxxrw を実行する。
execute '=gdxxrw @temp.txt';
* temp.txt を削除
execute 'del temp.txt';
```

ここでは GDX ファイル内の results と results_pc というパラメータの値を Excel ファイル ge_zero_h_check.xlsx にエクスポートしている。まず、GDX ファイルから Excel ファイルに出 力する設定を temp.txt というファイルに書き込み、その temp.txt というファイルの中身を引数 として gdxxrw を実行するという手順になっている。

- 「\$enecho > temp.txt」という命令で \$onecho~\$offecho の間の記述を temp.txt というファ イルに出力することになる。
- 「gdxxrw @temp.txt」というように gdxxrw の後に@ マークを付けたファイル名を書くと、 temp.txt の中身を引数として gdxxrw を実行せよという意味になる。
- execute は外部プログラムを実行する命令であるが、「execute '=gdxxrw'」のように外部プ ログラム名の前に「=」を付けると、そのプログラムの実行が終了するまでその後のコードを 実行せずに待てという意味になる。
- execute 'del temp.txt'; は temp.txt ファイルを削除するという意味である。「del」は Windows に標準的に付属する、ファイルを削除するためのプログラム名である。execute 命 令、外部プログラム gdxxrw については詳しくは "GDXXRW" を参照して欲しい。

3.2 結果

それでは結果を見てみよう。表1と表2は上のプログラムを実行した結果を ge_zero_h_check.xlsx に出力したものである。表1は各シナリオにおける変数の値(水 準)を表しており、表2は sce1の値からの変化率(%)を表している。

[sce1]

均衡において全ての絶対価格は 1 になる。これは ge_sample_dual.gms で既に見た。こうなる

表	1:	変数の値	

		sce1	sce2	sce3	sce4	sce5
効用	u	360.00	480.00	360.00	360.00	480.00
	y_agr	140.00	174.80	140.00	140.00	174.80
生産量	y_man	300.00	417.06	300.00	300.00	417.06
	y_ser	150.00	188.56	150.00	150.00	188.56
	c_agr	70.00	87.40	70.00	70.00	87.40
消費量	c_man	220.00	305.85	220.00	220.00	305.85
	c_ser	70.00	88.00	70.00	70.00	88.00
	p_man	1.00	0.81	2.00	1.00	0.52
絶対価格	pf_lab	1.00	1.56	2.00	1.00	1.00
	m	360.00	420.89	720.00	360.00	270.00
	p_man-p_agr	1.00	0.81	1.00	1.00	0.81
AGRで測った価格	pf_lab-p_agr	1.00	1.56	1.00	1.00	1.56
	m-p_agr	360.00	420.89	360.00	360.00	420.89
	p_man-pf_lab	1.00	0.52	1.00	1.00	0.52
労働で測った価格	pf_lab-pf_lab	1.00	1.00	1.00	1.00	1.00
	m-pf_lab	360.00	270.00	360.00	360.00	270.00

表 2: sce1 における値からの変化率(%)

		sce1	sce2	sce3	sce4	sce5
効用	u	0.00	33.33	0.00	0.00	33.33
	y_agr	0.00	24.85	0.00	0.00	24.85
生産量	y_man	0.00	39.02	0.00	0.00	39.02
	y_ser	0.00	25.71	0.00	0.00	25.71
	c_agr	0.00	24.85	0.00	0.00	24.85
消費量	c_man	0.00	39.02	0.00	0.00	39.02
	c_ser	0.00	25.71	0.00	0.00	25.71
	p_man	0.00	-19.34	100.00	0.00	-48.26
絶対価格	pf_lab	0.00	55.88	100.00	0.00	0.00
	m	0.00	16.91	100.00	0.00	-25.00
	p_man-p_agr	0.00	-19.34	0.00	0.00	-19.34
AGRで測った価格	pf_lab-p_agr	0.00	55.88	0.00	0.00	55.88
	m-p_agr	0.00	16.91	0.00	0.00	16.91
	p_man-pf_lab	0.00	-48.26	0.00	0.00	-48.26
労働で測った価格	pf_lab-pf_lab	0.00	0.00	0.00	0.00	0.00
	m-pf_lab	0.00	-25.00	0.00	0.00	-25.00

のは、元々、均衡価格が1になるようにモデルのパラメータを設定しているためである。

[sce2]

資本の賦存量を2倍にした結果、生産量は全部門で増加している。特に MAN の生産の増加率が 高い。これは MAN が資本集約的な部門であるためである。MAN の絶対価格は低下している。こ れは MAN の生産の伸び率が大きいことが原因である。一方、労働の絶対価格はかなり上昇してい る。これは労働が相対的に希少になったことが原因と考えられる。

効用も上昇している。これは要素賦存量が増加しているので当然の結果である。絶対的な所得、 及び AGR で測った相対所得は上昇しているのに対し、労働で測った相対所得は低下している。労 働で測った相対所得が低下しているのは労働の価格の上昇が高いためである。このように、何を基 準にして測るかによって価格、所得の変化の大きさ、方向は変わってくる。これは、計算結果を見 るときに、何をニュメレールにとっているのか、あるいは何を基準にしているのか(何をデフレー タにしているのか)が重要であるということである。

[sce3]

生産量、消費量、効用など数量を表す変数については sce1 のときと全く変わらない。また、 AGR で測った相対価格、労働で測った相対価格も全く変わらない。変わるのは絶対価格だけであ り、理論モデルから導けるように全て 2 倍になっている。このモデルでは実際に価格について 0 次同次性が成り立っていることが確認できる。

[sce4]

このケースは sce1 と全く変わらない。つまり、ニュメレールを労働としても(労働の価格を 1 と固定して解いても)、AGR がニュメレールのときと全く変わらないということである。

数量変数や相対価格が何をニュメレールにしようと変わらないのは当然の結果である。しかし、 ここでは絶対価格、p("man")やm などの値も変わっていない。本来、これらの値は何をニュメ レールにするかによって変化するはずである。今回、p("man")やm の値も変わっていないのは、 労働の価格 p_f("lab")が sce1 において元々1 という値をとっていたためである。労働にニュメ レールを変更しているが、それが元々均衡において1という値をとっていたものを1 に固定したと いうだけであるので、全く何も変わらないという結果となっている。

[sce5]

sce4 ではニュメレールを労働に変更したが、労働は元々 1 という均衡価格を持っていたので何 も変わらないという結果となった。これではニュメレールを変更することの効果がわかりにくいの で、sce2 と同じシミュレーションをニュメレールを変更した上でおこなったのが sce5 である。 ニュメレールの変更によって sce2 とどのような違いが生じるかを確認する。

まず、効用、生産量、消費量などの数量変数の値は何も変わっていない。つまり、ニュメレール の変更は資本賦存量が数量変数に与える効果に全く影響を与えないということである。また、相対 価格も全く変わっていない。ニュメレールの選択は相対価格にも全く影響しないことが確認でき る。一方、絶対価格は変わっている。何をニュメレールにするかで絶対価格は全く違う値になるこ とが確認できる。

3.3 まとめ(価格についての0次同次性のチェック)

以上、価格についての 0 次同次性が成り立っているかどうかを確認した。このプログラムでは 実際に成り立っていることがわかったが、モデル、あるいはプログラムのどこかがおかしい場合に は 0 次同次性が成り立たないということが生じてくる。例えば、ge_zero_h_check_fail.gms を実 行して、結果を見て欲しい。このプログラムの実行結果では、ニュメレールの変更によって数量変 数や相対価格の値が変わってしまう。これは均衡価格についての 0 次同次性が満たされないモデ ルになっているということであり、実際このプログラムには 2 カ所間違いが含まれている(自分で 探してみて欲しい)。

このように、ニュメレールを変更して解いてみて結果が変わらないかをチェックすることでモデ ル、プログラムに間違いがないかを確認することができる。簡単にできるチェックであるので、自 分でモデルを作成したときには、必ずニュメレールを用いたチェックをおこなうのがよい³⁾。

4 ワルラス法則のチェック

価格についての 0 次同次性とともに、我々が考える一般均衡モデルで成り立つ性質としてワル ラス法則があった。以下では、ワルラス法則が実際に成り立っているかどうかをチェックする。

4.1 プログラムとシナリオ

*

モデルには第 3 節と同じモデルを利用する。ge_walras_check.gms がそのプログラムである。 計算するシナリオは前節の sce1~sce5 から sce3 を除いた 4 つのシナリオである。以下、主に ge_zero_h_check.gms との相違点を説明する。

```
ワルラス法則が成立しているかのチェック。0 になれば OK
results("p_agr_m",sc) = p.m("agr"); \
results("pf_lab_m",sc) = p_f.m("lab"); \
```

ここはマクロ out_resutls(sc)の定義の部分である。ここでは p("agr") と p_f("lab")の marginal 値を代入している。第4章で説明したように、MCP タイプのモデルにおいて変数の marginal 値はその変数が対応する式の乖離幅を表している。モデルを定義している ge_sample_dual.gms を見て欲しい。例えば、変数 p は次のような式 eq_p に対応している。

```
    * 財の市場均衡
    e_p(i) ... y(i) =e= sum(j, a_x(i,j)*y(j)) + d(i);
```

³⁾ 当たり前の話のように聞こえるかもしれないが、CGE 分析の学習者には 0 次同次性が満たされないモデルを作成し てシミュレーションをしようとし、正常に解けないという問題に直面していることがよく見られる。

この e_p(i) の定義より、変数 p(i) の marginal 値は次の値を表すことになる。

y(i) - sum(j, a_x(i,j)*y(j)) - d(i)

これは財*i*の超過供給量であるので、変数 p(i)の marginal 値は結局財*i*の超過供給量を表すことになる。よって、p.m(i)がゼロかどうかを見ることで、財*i*の市場が均衡しているかどうかがわかる。同様に変数 p_f(f)は生産要素 fの市場均衡条件 e_p_f(f)に対応しており、p_f.m(f)の値は生産要素 fの超過供給量を表すことになる。

```
    * これもワルラス法則が成立しているかのチェック(全ての市場の超過
    * 需要額の和を計算)。0になれば OK
    results("v_ed",sc) = \
    sum(i, p.l(i)*(sum(j, a_x.l(i,j)*y.l(j)) + d.l(i) - y.l(i))) \
    + sum(f, p_f.l(f)*(sum(i, a_f.l(f,i)*v_a.l(i)) - v_bar(f))) \
    + sum(i, p_va.l(i)*(a_v.l(i)*y.l(i) - v_a.l(i))); \
```

これは全ての市場の超過需要額の和の値を計算し、パラメータに代入している部分である。ワル ラス法則とは全ての市場の超過需要額の和が0になるという性質であった。よって、この値が0に なるならば、ワルラス法則が満たされていることになる。

```
四捨五入
results("p_agr_m",sc) = round(results("p_agr_m",sc), 5) + eps; \
results("pf_lab_m",sc) = round(results("pf_lab_m",sc), 5) + eps; \
results("v_ed",sc) = round(results("v_ed",sc), 5) + eps;
```

ここは小数点以下第6位で四捨五入して桁数を減らしている。eps は 0 に近いが 0 ではない数 値を表す特殊な記号である。eps を加えているのは後で Excel ファイルに結果を出力するためで ある。

それでは以下でシナリオの説明をおこなう。

[シナリオ sce1]

これは何も変化させない元々の均衡である。この均衡で全ての市場の超過需要額の和が 0 に なっているかをチェックする。また、このシナリオでは AGR をニュメレールにとって、p("agr") を 1 に固定している。このように変数を固定すると、その変数が対応している式はモデルから除 外される。変数 p("agr") は e_p("agr") に対応しているので、この場合は AGR の市場均衡条件 がモデルから除外されることになる。しかし、もしワルラス法則が成立するならば、AGR 以外の 市場が均衡していれば、AGR の市場も自動的に均衡するはずである。ワルラス法則が成り立って いるかどうかのチェックのもう一つの方法として、モデルから除外された AGR の市場が均衡して いるかをチェックする。これには p.m("agr") の値が 0 になっているかどうかを調べればよい。

[シナリオ sce2]

これは資本の賦存量を2倍にするシナリオである。チェックする項目は sce1 と同じである。

[シナリオ sce4]

このシナリオではニュメレールを労働に変更している。全ての市場の超過需要額の和が 0 に なっているかどうかをチェックするのは sce1 と同じ。ニュメレールは労働に変更したのでこの 場合は労働市場の均衡条件がモデルから除外されている。そこで労働市場の均衡条件が実際に成 り立っているかをチェックする。これには p_f.m("lab") の値が 0 になっているかを確認すれば よい。

[シナリオ sce5]

「sce4 +資本の賦存量を2倍」というシナリオである。この場合のチェック項目は sce4 と同じ。

4.2 結果

表 3 は各シナリオにおける変数の値を掲載している。モデルもシナリオも同じであるので、基本的には第 3 節のシミュレーションの結果と同じである。

[sce1]

まず p("agr")の marginal 値はゼロになっている。よって、AGR の市場は均衡していることが 確認できる。これはワルラス法則が満たされていることを意味する。また、全ての市場の超過需要 額の和はゼロになっている。こちらでもワルラス法則が満たされていることがわかる。

[sce2]

資本を増加させても sce1 で成立した二つの結果が同様に成り立っていることがわかる。

[sce4 & sce5]

ニュメレールを変更したが、ニュメレールの市場は均衡しているし、超過需要額の和はゼロであ ることがわかる。やはりワルラス法則は成り立っていることがわかる。

4.3 ワルラス法則が満たされていないケース

前節のプログラムではワルラス法則は満たされた。ここでは参考のためワルラス法則が満 たされないケースを見てみよう。以下は ge_walras_check_fail.gms を実行した結果である。 このプログラムは基本的に ge_walras_check.gms と同じであるが、モデルのファイルとして ge_sample_dual_alt.gms を利用しているところだけが異なっている。

[結果]

結果は表 4 にある。この場合、sce1 では p("agr") の marginal 値が –29.91 になっている。こ

		sce1	sce2	sce4	sce5
効用	u	360.00	480.00	360.00	480.00
	y_agr	140.00	174.80	140.00	174.80
生産量	y_man	300.00	417.06	300.00	417.06
	y_ser	150.00	188.56	150.00	188.56
	c_agr	70.00	87.40	70.00	87.40
消費量	c_man	220.00	305.85	220.00	305.85
	c_ser	70.00	88.00	70.00	88.00
	p_man	1.00	0.81	1.00	0.52
絶対価格	pf_lab	1.00	1.56	1.00	1.00
	m	360.00	420.89	360.00	270.00
Marginal/di	p_agr_m	0.00	0.00	0.00	0.00
waigiiai庫	pf_lab_m	0.00	0.00	0.00	0.00
超過需要額	v_ed	0.00	0.00	0.00	0.00

表3:ワルラス法則のチェック

表4:ワルラス法則のチェック(上手くいかないケース)

		sce1	sce2	sce4	sce5
効用	u	359.88	486.61	356.16	508.13
	y_agr	117.87	148.59	158.33	204.47
生産量	y_man	311.13	435.07	310.36	455.25
	y_ser	159.88	198.73	167.37	218.16
	c_agr	70.13	88.16	70.28	93.42
消費量	c_man	219.64	311.10	215.75	321.69
	c_ser	70.11	88.81	70.17	93.97
	p_man	1.01	0.79	1.18	0.56
絶対価格	pf_lab	0.98	1.54	1.00	1.00
	m	361.44	422.43	413.27	307.23
Marginal/di	p_agr_m	-29.91	-33.78	0.00	0.00
iviaigiliai@	pf_lab_m	0.00	0.00	-27.80	-35.71
超過需要額	v_ed	-0.16	6.00	-6.57	16.63

れは AGR の市場に 29.91 の超過需要が生じていることを表している。その結果、全ての市場の超 過需要額の和もゼロにはなっていない。また、sce4 では労働市場に 27.80 の超過需要が発生して いる。sce2、sce5 についてもニュメレールの市場が均衡していないのと同時に、超過需要の総額 が 0 になっていない。

以上の結果は、このモデル(ge_sample_dual_alt.gms のモデル)においてワルラス法則が満た されていないことを示している。これはモデル、あるいはデータのどこかに間違いが含まれている ことを意味している。

4.4 まとめ(ワルラス法則のチェック)

ワルラス法則のチェックも、価格の0次同次性のチェック同様に、モデル、データ、プログラム の誤りをチェックするために利用できる。価格の0次同次性のチェック同様に、自分でモデルを作 成したときには、このチェックをおこなうのがよい。

5 外生変数の比例変化によるチェック

ここまで我々が考えてきたモデル(この後のモデルでも同じだが)は生産関数や効用関数に一次 同次の関数を利用している。このようなモデルにおいては、モデルの外生変数(主に生産要素の賦 存量)を同率、例えば X% だけ比例的に変化させたときに

- 全ての内生的な数量変数(生産量、消費量等)、金額変数(生産額、消費額、所得額、支出額
 等)も同率で(X%)変化する
- 価格変数のみ変わらない

という結果になるはずである。逆に言えば、このような結果が成り立たないならばモデル、デー タ、プログラムのどこかがおかしいということになる。「外生変数の比例変化によるチェック」と はこの性質を利用しておこなうチェックのことである。

[注意]:効用関数や生産関数に一次同次関数を利用していないなら元々上の性質は成り立たないの で、このチェックは使えない。費用に固定費用がかかるモデルや、従量税を導入しているモデル、 数量規制を導入しているモデルでも同じである。

5.1 シナリオ

ge_proportional_check.gms というファイルがこのチェックをおこなうプログラムである。中 身は価格の 0 次同次性をチェックした ge_zero_h_check.gms とほぼ同じである。ただし、解くシ ナリオだけ違っている。シナリオは次の 4 つである。

- sce1:これは何も変化させないで解くシナリオ。
- sce2:これは資本の賦存量を2倍にするシナリオ
- sce3:これは資本と労働の賦存量を2倍にするシナリオ
- sce4:これは資本と労働の賦存量を 0.5 倍にするシナリオ

チェックしたいのは sce3 において数量変数が 2 倍になると同時に、価格が変化しないかどう か、sce4 において数量変数が 0.5 倍になると同時に、価格が変化していないかどうかということ である。

5.2 結果

		scel	sce2	sce3	sce4
効用	u	0.00	33.33	100.00	-50.00
	y_agr	0.00	24.85	100.00	-50.00
生産量	y_man	0.00	39.02	100.00	-50.00
	y_ser	0.00	25.71	100.00	-50.00
	c_agr	0.00	24.85	100.00	-50.00
消費量	c_man	0.00	39.02	100.00	-50.00
	c_ser	0.00	25.71	100.00	-50.00
	p_man	0.00	-19.34	0.00	0.00
絶対価格	pf_lab	0.00	55.88	0.00	0.00
	m	0.00	16.91	100.00	-50.00
	p_man-p_agr	0.00	-19.34	0.00	0.00
AGRで測った価格	pf_lab-p_agr	0.00	55.88	0.00	0.00
	m-p_agr	0.00	16.91	100.00	-50.00
	p_man-pf_lab	0.00	-48.26	0.00	0.00
労働で測った価格	pf_lab-pf_lab	0.00	0.00	0.00	0.00
	m-pf_lab	0.00	-25.00	100.00	-50.00

表5:外生変数の比例変化によるチェック(sce1からの変化率、%)

表5の値は sce1 からの変数の変化率(%)を表している。まず、sce2 では生産量、消費量等の 数量変数も価格もどちらも変化しており、その変化率は変数によって異なっている。これに対し、 資本・労働をどちらも2 倍ずつに増加させたシナリオでは、全ての数量変数、金額変数が 100% 増 加(つまり、2 倍に増加)し、一方、価格変数は変化していない。期待通りの計算結果になってい ることがわかる。sce4 についても同様に資本・労働の減少率に等しいだけ全ての数量変数、金額 変数が減少し、価格変数は変化していない。

同率の変化をもたらさないケース

ge_proportional_check.gms では外生変数の比例的変化に対し、内生的な数量変数、金額 変数は同率で変化し、価格変数は不変であった。ge_proportional_check_fail.gms はその ような結果が成り立たないケースである。このプログラムではモデルのファイルとして ge_sample_dual_alt.gms を用いている。実際、計算して見てチェックして欲しい。

5.3 まとめ

以上の例が示すように、ある種のモデルにおいては、外生変数を比例的に変化させるというシ ミュレーションをおこなうことでモデルの間違いをチェックすることが可能である。特に、この方 法を使うことで、ニュメレールの変更によるチェック、ワルラス法則によるチェックでは見つけら れない誤りを見つけることもできるので、ニュメレールの変更によるチェック、ワルラス法則によ るチェックとともにこの方法によるチェックをおこなうようにするとよい。

6 簡単な政策シミュレーション

この節では、例として次の二つの政策の分析をおこなう。

- MAN と SER に対して 10% の消費税を導入するシミュレーション
- 3 つの部門の労動投入に 20% の課税を導入するシミュレーション

6.1 シナリオ

ge_simulation.gms というプログラムがこの政策のシミュレーションをおこなっているファイ ルである。利用するモデルのファイルは、ここまでと同様に ge_sample_dual.gms である。以下で は、各シナリオの設定をしている部分のプログラムを説明する。

set	sce	計算するシナリオ /
		sce1 デフォールトの均衡
		sce2 man と ser に対して 10%の消費税を導入するシナリオ
		sce3 労動投入に 20%の課税を導入するシナリオ
		/;
displa	y sce;	

ここではシナリオを表す集合を定義している。シミュレーションではデフォールトの均衡から政 策によってどれだけ変数が変化するかを分析する。

```
* ------
* SCE2: man と ser に対して 10%の消費税を導入するシナリオ
t_c("man") = 0.1;
t_c("ser") = 0.1;
solve ge_sample_dual using mcp;
out_resutls("sce2");
t_c(i) = 0;
```

ここでは sce2 を解いている。t_c(i) が財 i の消費に対する従価税率を表すパラメータである。 MAN と SER についてこの t_c(i) に 0.1 をセットすると、両財の消費に 10% の従価税が課され ることになる。元々 t_c(i) はゼロに設定されている。つまり元々は消費税はないと仮定されてい る。解いた後に初期値のゼロを入れ直している。

```
* ------
* SCE3: 労動投入に 20%の課税を導入するシナリオ
t_f("lab",i) = 0.2;
solve ge_sample_dual using mcp;
out_resutls("sce3");
t_f(f,i) = 0;
```

ここでは sce3 を解いている。t_f(f,i) は部門 i の生産要素 f の投入に対する税率を表している。t_f("lab",i) に 0.2 を設定すると、全ての部門の労働投入に 20% の課税を導入することを意味する。ここでも解いた後に、初期値である 0 に戻している。

6.2 結果

シミュレーション結果を見る前に、各シナリオでのチェック項目を確認する。

[sce2]

これは MAN と SER に消費税を導入するシナリオである。MAN と SER は税金がかかるので、 その消費、生産が影響を受けると考えられる。消費、生産がどう変化するかをチェックする。同時 に、税金がかからない AGR の消費、生産がどう変化するかもチェックする。また、税金の導入に よって効用の水準がどう変化するかもチェックする。

[sce3]

これは労働投入への税金という政策であるので、労働投入がどう変化するかをチェックする。また、その結果、効用水準がどう変化するかをチェックする。

それでは以下で結果を見てみよう。表 6 が結果を示している。

6.2.1 シミュレーション結果

表 6 の数値は sce1 における変数の数値からの変化率(%)を表している。

[sce2]

消費税が導入された MAN と SER の消費量は減少し、それに伴い生産量を減少している。一 方、消費税がかからない AGR の消費量、生産量は増加している。これは MAN と SER で利用さ れていた生産要素が AGR の生産に利用されるようになるからである。AGR で測った MAN の相 対価格、労働で測った MAN の相対価格はともに低下している。これは MAN への消費税によって MAN への需要が減少したためである。効用水準は低下している。このモデルは完全競争モデル

		sce1	sce2	sce3
効用	u	0.00	-0.04	0.00
	y_agr	0.00	2.26	0.00
生産量	y_man	0.00	-0.74	0.00
	y_ser	0.00	-0.42	0.00
	c_agr	0.00	3.85	0.00
消費量	c_man	0.00	-0.92	0.00
	c_ser	0.00	-0.97	0.00
	p_man	0.00	-0.12	0.00
絶対価格	pf_lab	0.00	0.31	-16.67
	m	0.00	7.90	0.00
	p_man-p_agr	0.00	-0.12	0.00
AGRで測った価格	pf_lab-p_agr	0.00	0.31	-16.67
	m-p_agr	0.00	7.90	0.00
	p_man-pf_lab	0.00	-0.43	20.00
労働で測った価格	pf_lab-pf_lab	0.00	0.00	0.00
	m-pf_lab	0.00	7.57	20.00

表 6: 政策シミュレーション (sce1 からの変化率、%)

で、他に何の歪みもないモデルである。よって、自由放任がパレート効率的であり、税金の導入は 代表的家計の効用を低下させる方向に働くはずである。計算結果は実際にそのような結果が成り 立っていることを示している。

[sce3]

表の結果からわかるように価格が変化するだけで生産量も消費量も変わらない。生産量が変わら ないので労働投入量も変わらない。数量が全く変化しないので効用水準も全く変わらない。

なぜ税金を導入したにもかかわらずその効果がないのか?このモデルでは労働の供給量は外生的 に固定されている。従って、どう税金をかけようが労働供給量全体が変化することはない。一方、 労働課税は全ての部門で同じ率で導入されているので、部門間での労働の配分が変わる効果も生じ ない。以上のように、労働供給全体の量が変わらないことに加え、部門間での配分も変化しないの で、全く何の効果も持たないという結果になる。

投入側で労働への課税を支払い、それを家計が受け取るという効果が生じているのであるが、労 働課税の分、企業が支払う賃金、家計が受け取る賃金が減少する。この賃金の減少が家計に税の収 入が生じるという効果を完全に相殺してしまうので、所得の変化の効果も生じない。

7 終わりに

この章では一般均衡モデルを数値的に解く際の、モデルのチェック方法を説明した。一般均衡モ デルにおいて、成り立つべき性質が実際に成り立っているかを見ることで、モデルに誤りが含まれ ているかどうか判断するという方法であった。後に、第 9 章ではこの章で紹介した方法も含め、 CGE モデルをチェックする方法について見る。

参考文献

Simon, C. P. and L. Blume (1994). Mathematics for Economists. New York: W. W. Norton Company.

8 問題

問題1

ここまでチェックのためのシミュレーションをおこなうにあたって、モデルとしては ge_sample_dual.gms のモデル、つまり双対アプローチを用いて記述したモデルを用いてきた。同 じシミュレーションを伝統的アプローチによって記述したモデルを用いておこなってみよ。

問題 2

全ての財に同率(例えば、10%)の消費税を課すという政策の効果を計算し、その結果を説明し なさい。

9 履歴

- 2023-04-28: 誤植の修正。プログラムのファイル名の変更(walrus → walras)
- 2022-01-13: 説明の追加・修正。
- 2018-07-20: 説明の追加・修正。
- 2017-03-15: 説明の修正。