Hitotsubashi Journal of Economics 38 (1997) 139-148. © The Hitotsubashi Academy

A PROOF OF ARROW'S IMPOSSIBILITY THEOREM BY MATHEMATICA

SHIN-ICHI TAKEKUMA

Abstract

Arrow's impossibility theorem, the non-existence of social welfare function, is proved by using a program of Mathematica. A society consisting of two individuals and three alternatives is considered, and the social welfare function which satisfies the conditions of unanimity and independence of irrelevant alternatives is constructed by programming and is shown to be dictatorial.

I. *Introduction*

The purpose of this paper is to prove the impossibility theorem of Arrow (1963), the non-existence of social welfare function, by using a program of Mathematica. There have been many papers on Arrow's Impossibility Theorem. Among them, the proofs by Fishburn (1970) and Suzumura (1988) suggest that the theorem can be proved by finitely many steps, and therefore by a computer's program. In this note, we consider a simple society consisting of two individuals and three alternatives, and show that the theorem can be proved by a program of Mathematica.

The step of our proof follows the original arguments by Arrow (1950). First, we start with a social welfare function that satisfies the unanimity condition. Second, we assume that one of the two individuals is not the dictator. Third, we extend the social welfare function to another one by using the condition of independence of irrelevant alternatives. We repeat the same procedure sequentially. Finally, we show that, in the social welfare function which we get in the end, the other individual is the dictator.

In general cases where many individuals and alternatives exist, the theorem could be proved by Mathematica in a similar way, but it might neither be easy nor obvious.

II. *Notation*

Let us consider a simple society which consists of two individuals and three alternatives to choose. Two individuals are called "individual 1" and "individual 2". Three alternatives are named as "a", "b", and "c". The set of all alternatives is denoted by X, and the product of X and itself is denoted by $X \times X$, that is,

$X=\{a, b, c\}$, and $X\times X=\{(x, y) \mid x\in X, y\in X\}$.

The sets X and $X\times X$ are defined and denoted respectively by **sX** and **psX** in our programming of Mathematica as follows:

```
In[1]:=
    sX={a,b,c}
Out[1]:=
    {a, b, c}
In[2]:=
    psX=Flatten[Table[ {sX[[i]],sX[[j]]},{i,3},{j,3} ],1]
Out[2]:=
    {{a, a}, {a, b}, {a, c}, {b, a}, {b, b}, {b, c},
    {c, a}, {c, b}, {c, c}}
```

Let $\succ$ be a relation, which is a subset of $X\times X$. When $(x, y)\in\succ$, which is written as $x\succ y$, it means that x is preferred to y in $\succ$. We consider relations which are asymmetric and negatively transitive, namely, relations which have the following properties:

(AS) $x\succ y$ implies not $y\succ x$.

(NT) not $x\succ y$ and not $y\succ z$ imply not $x\succ z$.

An asymmetric and negatively transitive relation is called a "weak order". We write $x\sim y$ if and only if not $x\succ y$ and not $y\succ x$, If relation $\succ$ is asymmetric and negatively transitive, then relation $\succ$ is transitive, and moreover exactly one of $x\succ y$, $y\succ x$, $x\sim y$ holds for each x, $y\in X$ (see Fishburn (1970, Thm.2.1, p.13).

By $\mathscr{P}$, we denote the set of all weak orders on set X. In our programming, set $\mathscr{P}$ is denoted by **sP**, and the number of elements in $\mathscr{P}$ is 13, as shown in the following:

```
In[3]:=
    sP3=Table[Delete[Permutations[sX][[i]],{4-j} ],
              {i,6},{j,3} ];
    y={};
    Do[AppendTo[y,Sort[Delete[sP3[[i]],5-2j]]],{i,6},{j,2} ];
    sP2=Union[y];
    sP=Join[ {{}},sP2,sP3]
Out[7]:=
    {{}, {{a, b}, {a, c}}, {{a, b}, {c, b}},
    {{a, c}, {b, c}}, {{b, a}, {b, c}},
    {{b, a}, {c, a}}, {{c, a}, {c, b}},
    {{a, b}, {a, c}, {b, c}}, {{a, c}, {a, b}, {c, b}},
    {{b, a}, {b, c}, {a, c}}, {{b, c}, {b, a}, {c, a}},
    {{c, a}, {c, b}, {a, b}}, {{c, b}, {c, a}, {b, a}}}
In[8]:=
    nsP=Length[sP]
Out[8]:=
    13
```

III. *Social Welfare Function*

A social welfare function is a mapping from $\mathscr{P}\times\mathscr{P}$ to $\mathscr{P}$. Let f: $\mathscr{P}\times\mathscr{P}\rightarrow\mathscr{P}$ be a social welfare function. When $(\succ_1, \succ_2)\in\mathscr{P}\times\mathscr{P}$ and $f(\succ_1, \succ_2)=\succ_S$, relations $\succ_1$ and $\succ_2$ denote respectively the preferences of individual 1 and individual 2, and relation $\succ_S$ denotes the social preference derived from social welfare function f.

Let us state Arrow's conditions on the social welfare function. The social welfare function follows the unanimity condition. If everybody prefers x to y, then x is socially referred to y:

(U) If $x\succ_1 y$ and $x\succ_2 y$, then $x\succ_S y$.

An individual such that whenever he prefers x to y, then x is socially preferred to y, is the "dictator". The social welfare function admits no dictator:

(ND) There is no i such that, for all x, y and $(\succ_1, \succ_2)\in\mathscr{P}\times\mathscr{P}$, $x\succ_i y$ implies $x\succ_S y$,

As far as any individual does not change his preference between x and y, the social preference between x and y does not change. The social preference is independent of irrelevant alternatives:

(IIA) Let $x, y\in X$, $(\succ_1, \succ_2)$, $(\succ_1', \succ_2')\in\mathscr{P}\times\mathscr{P}$, and for each $i=1, 2$, $x\succ_i y$ if and only if $x\succ_i' y$. Then, $x\succ_S y$ if and only if $x\succ_S' y$, where $f(\succ_1', \succ_2')=\succ_S'$.

IV. *Proof by Mathematica*

If social welfare function f satisfies condition (U), then $f(\succ_1, \succ_2)\supset\succ_1\cap\succ_2$. Therefore, the "minimum" social welfare function that satisfies condition (U) is defined by $\succ_1\cap\succ_2$, which is denoted by **paretoSWF** in our programming and is defined as follows:

In[9]:=

```
paretoSWF[p1__,p2__]:=Intersection[p1,p2]
```

Let us assume that individual 1 is not the dictator, that is, there is a pair (x, y) of alternatives such that $x\succ_1 y$ and not $x\succ_S y$. Without loss of generality, we assume that that $a\succ_1 c$ and not $a\succ_S c$. In this case, apparently, by (U), not $a\succ_2 c$. Also, when $a\succ_S b$, not $a\succ_S c$ implies $c\succ_S b$. Moreover, when $b\succ_S c$, not $a\succ_S c$ implies $b\succ_S a$. Thus, function **paretoSWF** can be extended to a function, **fd1**, in the following way:

In[10]:=

```
fd1[p1__,p2__]:=Which[Count[p1,{a,c}]==0,paretoSWF[p1,p2],
    Count[p2,{a,c}]>0,paretoSWF[p1,p2],
    Count[paretoSWF[p1,p2],{a,b}]>0,
        Union[paretoSWF[p1,p2],{{c,b}}],
    Count[paretoSWF[p1,p2],{b,c}]>0,
        Union[paretoSWF[p1,p2],{{b,a}}],
    True,paretoSWF[p1,p2]]
```

The difference of **paretoSWF** and **fd1** is shown in the following list:

```
In[11]:=
    Do[If[paretoSWF[sP[[i]],sP[[j]]]=!=fd1[sP[[i]],sP[[j]]]],
            Print["case(",i,",",j,")"];
            Print["      ",sP[[i]],",",sP[[j]]];
            Print[" => ",fd1[sP[[i]],sP[[j]]]]],
    {i,nsP},{j,nsP}]
    case(2,3)
        {{a, b}, {a, c}}, {{a, b}, {c, b}}
    => {{a, b}, {c, b}}
    case(2,12)
        {{a, b}, {a, c}}, {{c, a}, {c, b}, {a, b}}
    => {{a, b}, {c, b}}
    case(4,5)
        {{a, c}, {b, c}}, {{b, a}, {b, c}}
    => {{b, a}, {b, c}}
    case(4,11)
        {{a, c}, {b, c}}, {{b, c}, {b, a}, {c, a}}
    => {{b, a}, {b, c}}
    case(8,3)
        {{a, b}, {a, c}, {b, c}}, {{a, b}, {c, b}}
    => {{a, b}, {c, b}}
    case(8,5)
        {{a, b}, {a, c}, {b, c}}, {{b, a}, {b, c}}
    => {{b, a}, {b, c}}
    case(8,11)
        {{a, b}, {a, c}, {b, c}}, {{b, c}, {b, a}, {c, a}}
    => {{b, a}, {b, c}}
    case(8,12)
        {{a, b}, {a, c}, {b, c}}, {{c, a}, {c, b}, {a, b}}
    => {{a, b}, {c, b}}
```

In the above list, cases (4,5), (4,11), (8,5), and (8,11) show that, when not $b \succ_1 a$ and $b \succ_2 a$, $b \succ_S a$ holds. Therefore, by (U) and (IIA), $b \succ_2 a$ implies $b \succ_S a$. Starting with this fact, we extend **paretoSWF** to **f1** by virtue of condition (IIA) in the following way:

```
In[12]:=
    swf0[p1__,p2__]:=paretoSWF[p1,p2];
    f1[p1__,p2__]:=Which[Count[p1,{b,a}]>0,swf0[p1,p2],
            Count[p2,{b,a}]>0,Union[swf0[p1,p2],{{b,a}}],
            True,swf0[p1,p2]]
```

Furthermore, since the social ordering is transitive, **f1** can be extended to **swf1** in the following way:

```
In[14]:=
    tran[p__]:=Which[Length[p] !=2,p,
              p[[1,2]]===p[[2,1]],
                  Union[p,{{p[[1,1]],p[[2,2]]}}],
              p[[1,1]]===p[[2,2]],
                  Union[p,{{p[[2,1]],p[[1,2]]}}],
              True,p]
In[15]:=
        swf1[p1__,p2__]:=tran[f1[p1,p2]]
```

There are 40 cases in the difference of **paretoSWF** and **f1**. The difference of **f1** and **swf1** is shown in the following list:

```
In[16]:=
        r=0;
        Do[If[f1[sP[[i]],sP[[j]]] =!=
                swf0[sP[[i]],sP[[j]]],r=r+1],
        {i,nsP},{j,nsP}];r
Out[17]:=
    40
In[18]:=
    Do[If[swf1[sP[[i]],sP[[j]]] =!=
              f1[sP[[i]],sP[[j]]],Print["case(",i,",",j,")"];
              Print["      ",sP[[i]],",",sP[[j]]];
              Print[" => ",swf1[sP[[i]],sP[[j]]]]],
    {i,nsP},{j,nsP}]
    case(2,10)
        {{a, b}, {a, c}}, {{b, a}, {b, c}, {a, c}}
    => {{a, c}, {b, a}, {b, c}}
    case(3,13)
        {{a, b}, {c, b}}, {{c, b}, {c, a}, {b, a}}
    => {{b, a}, {c, a}, {c, b}}
    case(9,10)
        {{a, c}, {a, b}, {c, b}}, {{b, a}, {b, c}, {a, c}}
    => {{a, c}, {b, a}, {b, c}}
    case(9,13)
        {{a, c}, {a, b}, {c, b}}, {{c, b}, {c, a}, {b, a}}
    => {{b, a}, {c, a}, {c, b}}
```

In the above list, cases (2.10) and (9.10) show that, when not $b \succ_1 c$ and $b \succ_2 c$, $b \succ_S c$ holds. Therefore, by (U) and (IIA), $b \succ_2 c$ implies $b \succ_S c$. Hence, in the exactly same way we can extend **swf1** to **swf2** in the following way:

```
In[19]:=
```

```
f2[p1__,p2__]:=Which[Count[p1,{b,c}]>0,swf1[p1,p2],
        Count[p2,{b,c}]>0,Union[swf1[p1,p2],{{b,c}}],
        True,swf1[p1,p2]];
swf2[p1__,p2__]:=tran[f2[p1,p2]]
```

There are 38 cases in the difference of **f2** and **swf1**. The difference of **f2** and **swf2** is shown in the following list:

```
In[21]:=
    r=0;
    Do[If[f2[sP[[i]],sP[[j]]]=!=swf1[sP[[i]],sP[[j]]]],
        r=r+1],
    {i,nsP},{j,nsP}];r
Out[22]:=
    38
In[23]:=
    Do[If[swf2[sP[[i]],sP[[j]]]=!=f2[sP[[i]],sP[[j]]]],
        Print["case(",i,",",j,")"];
        Print["      ",sP[[i]],",",sP[[j]]];
        Print[" => ",swf2[sP[[i]],sP[[j]]]]]],
    {i,nsP},{j,nsP}]
    case(3,8)
        {{a, b}, {c, b}}, {{a, b}, {a, c}, {b, c}}
    => {{a, b}, {a, c}, {b, c}}
    case(12,8)
        {{c, a}, {c, b}, {a, b}}, {{a, b}, {a, c}, {b, c}}
    => {{a, b}, {a, c}, {b, c}}
```

The above list shows that, when not $a \succ_1 c$ and $a \succ_2 c$, $a \succ_S c$ holds. Therefore, by (U) and (IIA), $a \succ_2 c$ implies $a \succ_S c$. Again, in the exactly same way we can extend **swf2** to **swf3**. There are 38 cases in the difference of **f3** and **swf2**. The difference of **f3** and **swf3** is shown as follows:

```
In[24]:=
    f3[p1__,p2__]:=Which[Count[p1,{a,c}]>0,swf2[p1,p2],
        Count[p2,{a,c}]>0,Union[swf2[p1,p2],{{a,c}}],
        True,swf2[p1,p2]];
    swf3[p1__,p2__]:=tran[f3[p1,p2]]
In[26]:=
    r=0;
    Do[If[f3[sP[[i]],sP[[j]]]=!=swf2[sP[[i]],sP[[j]]]],
        r=r+1],
    {i,nsP},{j,nsP}];r
Out[27]:=
    38
```

```
In[28]:=
    Do[If[swf3[sP[[i]],sP[[j]]]=!=f3[sP[[i]],sP[[j]]],
            Print["case(",i,",",j,")"];
            Print["      ",sP[[i]],",",sP[[j]]];
            Print[" => ",swf3[sP[[i]],sP[[j]]]]],
    {i,nsP},{j,nsP}]
    case(7,9)
        {{c, a}, {c, b}}, {{a, c}, {a, b}, {c, b}}
    => {{a, b}, {a, c}, {c, b}}
    case(13,9)
        {{c, b}, {c, a}, {b, a}}, {{a, c}, {a, b}, {c, b}}
    => {{a, b}, {a, c}, {c, b}}
```

The above list shows that, when not $a \succ_1 b$ and $a \succ_2 b$, $a \succ_S b$ holds, and therefore, by (U) and (IIA), that $a \succ_2 b$ implies $a \succ_S b$. Again, we can extend **swf3** to **swf4**. There are 38 cases in the difference of **f4** and **swf3**. The difference of **f4** and **swf4** is shown as follows:

```
In[29]:=
    f4[p1__,p2__]:=Which[Count[p1,{a,b}]>0,swf3[p1,p2],
            Count[p2,{a,b}]>0,Union[swf3[p1,p2],{{a,b}}],
            True,swf3[p1,p2]];
    swf4[p1__,p2__]:=tran[f4[p1,p2]]
In[31]:=
    r=0;
    Do[If[f4[sP[[i]],sP[[j]]]=!=swf3[sP[[i]],sP[[j]]],
            r=r+1],
    {i,nsP},{j,nsP}];r
Out[32]:=
    38
In[33]:=
    Do[If[swf4[sP[[i]],sP[[j]]]=!=f4[sP[[i]],sP[[j]]],
            Print["case(",i,",",j,")"];
            Print["      ",sP[[i]],",",sP[[j]]];
            Print[" => ",swf4[sP[[i]],sP[[j]]]]],
    {i,nsP},{j,nsP}]
    case(6,12)
        {{b, a}, {c, a}}, {{c, a}, {c, b}, {a, b}}
    => {{a, b}, {c, a}, {c, b}}
    case(11,12)
        {{b, c}, {b, a}, {c, a}}, {{c, a}, {c, b}, {a, b}}
    => {{a, b}, {c, a}, {c, b}}
```

The above list shows that, when not $c \succ_1 b$ and $c \succ_2 b$, $c \succ_S b$ holds, and therefore, by (U) and (IIA), that $c \succ_2 b$ implies $c \succ_S b$. Again, we can extend **swf4** to **swf5**. There are 38 cases in the difference of **f5** and **swf4**. The difference of **f5** and **swf5** is shown as follows:

```
In[34]:=
    f5[p1__,p2__]:=Which[Count[p1,{c,b}]>0,swf4[p1,p2],
            Count[p2,{c,b}]>0,Union[swf4[p1,p2],{{c,b}}],
            True,swf4[p1,p2]];
    swf5[p1__,p2__]:=tran[f5[p1,p2]]
In[36]:=
    r=0;
    Do[If[f5[sP[[i]],sP[[j]]]=!=swf4[sP[[i]],sP[[j]]]],
            r=r+1],
    {i,nsP},{j,nsP}];r
Out[37]:=
    38
In[38]:=
    o[If[swf5[sP[[i]],sP[[j]]]=!=f5[sP[[i]],sP[[j]]]],
            Print["case(",i,",",j,")"];
            Print["      ",sP[[i]],",",sP[[j]]];
            Print[" => ",swf5[sP[[i]],sP[[j]]]]]],
    {i,nsP},{j,nsP}]
    case(1,13)
        {}, {{c, b}, {c, a}, {b, a}}
    => {{b, a}, {c, a}, {c, b}}
    case(2,13)
        {{a, b}, {a, c}}, {{c, b}, {c, a}, {b, a}}
    => {{b, a}, {c, a}, {c, b}}
    case(4,13)
        {{a, c}, {b, c}}, {{c, b}, {c, a}, {b, a}}
    => {{b, a}, {c, a}, {c, b}}
    case(5,13)
        {{b, a}, {b, c}}, {{c, b}, {c, a}, {b, a}}
    => {{b, a}, {c, a}, {c, b}}
    case(8,13)
        {{a, b}, {a, c}, {b, c}}, {{c, b}, {c, a}, {b, a}}
    => {{b, a}, {c, a}, {c, b}}
    case(10,13)
        {{b, a}, {b, c}, {a, c}}, {{c, b}, {c, a}, {b, a}}
    => {{b, a}, {c, a}, {c, b}}
```

The above list shows that not $c \succ_1 a$ and $c \succ_2 a$ imply $c \succ_S a$, and therefore, by (U), that $c \succ_2 a$ implies $c \succ_S a$. Again, we can extend **swf5** to **swf6**. There are 32 cases in the difference of **f6** and **swf5**. As the following result shows, there is no difference between **f6** and **swf6**.

```
In[39]:=
    f6[p1__,p2__]:=Which[Count[p1,{c,a}]>0,swf5[p1,p2],
            Count[p2,{c,a}]>0,Union[swf5[p1,p2],{{c,a}}],
```

```
            True,swf5[p1,p2]];
    swf6[p1__,p2__]:=tran[f6[p1,p2]]
In[41]:=
    r=0;
    Do[If[f6[sP[[i]],sP[[j]]]=!=swf5[sP[[i]],sP[[j]]]],
            r=r+1],
    {i,nsP},{j,nsP}];r
Out[42]:=
    32
In[43]:=
    Do[If[swf6[sP[[i]],sP[[j]]]=!=f6[sP[[i]],sP[[j]]]],
            Print["case(",i,",",j,")"];
            Print["        ",sP[[i]],",",sP[[j]]];
            Print[" => ",swf6[sP[[i]],sP[[j]]]]],
    {i,nsP},{j,nsP}]
```

In the following final procedure, we check up if individual 2 is the dictator in social welfare function **swf6**, or not.

```
In[44]:=
    r=0;
    Do[If[Union[swf6[sP[[i]],sP[[j]]]]=!=Union[sP[[j]]],
        r=r+1;
            Print["        ",sP[[i]],",",sP[[j]]];
                Print[" => ",swf6[sP[[i]],sP[[j]]]]],
    {i,nsP},{j,nsP}];r
Out[45]:=
    0
```

The above result shows that individual 2 is the dictator. Namely, social welfare function **swf6**, in which individual 1 is not the dictator, and which satisfies conditions (U) and (IIA), makes individual 2 the dictator. Hence, we can conclude that there is no social welfare function that satisfies all the conditions (U), (IIA), and (ND).

HITOTSUBASHI UNIVERSITY

REFERENCES

Arrow, K. J.(1963), "*Social Choice and Individual Values*," 2nd edition, Wiley, Now York.

Arrow, K. J.(1950), A Difficulty in the Concept of Social Welfare, *Journal of Political Economy* 58, pp.328–346.

Fishburn, P. C.(1970), Arrow's Impossibility Theorem: Concise Proof and Infinite Voters, *Journal of Economic Theory* 2, pp.103–106

Fishburn, P. C.(1970), *Utility Theory for Decision Making*, John Wiley & Sons, Inc., New

York.

Suzumura, K.(1988), Reduction of Social Choice Problems: A Simple Proof of Arrow's General Possibility Theorem, *Hitotsubashi Journal of Economics* 29, pp.219–221.