Abstract

We describe two-person simultaneous play games. First, we use a zero sum game to illustrate minimax, dominant and best response strategies. We illustrate Nash Equilbria in the Prisoner's Dilemma and the Battle of the Sexes Game, and distinguish three types of Nash Equilibria: a pure strategy, a mixed strategy, and a continuum (partially) mixed strategy. Then we introduce the program, Nash.m and use it to solve the games. We display the full code of Nash.m, and finally we discuss the performance characteristics of Nash.m

A Program for Finding Nash Equilibria

John Dickhaut[1]
University of Minnesota
Carlson School of Management

Todd Kaplan [2]
University of Minnesota
Economics Department

June 16, 1991

[1]671 Management and Economics Building, 271 19th Ave. S., Minneapolis, MN. 55455

[2]E-Mail: todd@atlas.socsci.umn.edu

Introduction:

Simple parlour games such as tic-tac-toe and scissors, paper and rock can serve as references for elements of two person games. Such games have specific players, distinct choices for each player, and some notion of implicit payoffs for each player. In these cases, it is convenient to think of one player's gain (or win) as another's loss. In Chess or Go such games can become quite complicated, but it is still the case that moves are well-defined; furthermore, we expect the players to use some intelligent play or strategy. Other examples of strategic behavior include pricing in the market place, taking a particular political position, and deciding on the level of nuclear capacity. In such examples the strategy set is not only large but may not be well defined; also, one person's gain need not be the other's loss.

Game Theory is used to analyze environments in which there are distinct players, each with a set of well-defined actions, and each with a payoff function which associates that player's payoff with each possible combination of actions. By means of examples, the paper provides a brief introduction to the fundamentals of Game Theory, and focuses on those elements of Game Theory that relate to the program, Nash.m, our main contribution. The roots of the Game Theory used here are in [Von Neumann and Morgenstern, 1953] and [Nash, 1951], but the literature is extensive and employed in many diverse disciplines such as Economics, Statistics, Political Science, Finance, and Accounting. A very readable introduction is provided by [Rapoport, 1970]. More advanced applications are in [Kreps, 1990] and [Friedman, 1986].

We will represent the two person games as tensors which we will also call the Normal Form of the game (See for example Figure 1). This representation reflects the idea that for each player the moves are simultaneous. The entries in the cells reflect payoffs to both players. The outcome of the Game Theory analysis is to identify the equilibria of the game, that is what each intelligent player would do given each player is aware of the opposing players choices. Informally, the idea of an equilibrium is that it is a set of plans of action for each player such that neither player would change plans given the plan of the other player. In our examples we will be demonstrating several games with equilibria based on several different types of strategies: minimax, dominant strategy, and best response (Nash). We begin with a simple zero-sum game in which we demonstrate intuitively all three of the concepts minimax, dominant strategy, and best response; we then clarify the assumptions we are making

		Player B 1	Player B 2
Player A	1	(-.5,.5)	(-1,1)
Player A	2	(1,-1)	(0,0)

Figure 1: Simple Parlour Game

about payoffs and what is meant by mixed strategies. Then we examine non-zero sum games and illustrate the different types of Nash equilibria: pure strategy Nash equilibria, mixed strategy Nash equilibria, and partially mixed strategy Nash equilibria. We then present the program Nash.m which is applied to the example games, followed by our study of the robustness of Nash.m.

Definitions:

Consider the simple zero-sum game in Figure 1 . Each player has two choices, 1 or 2, which could be represented by each player hiding the choice of one finger or two behind the back. Player A's payoffs are the left entry in the cells while Player B's are the right entry. When each player's choices are simultaneously revealed, payoffs are made. In a few paragraphs we shall describe the nature of the numbers representing payoffs but for the time being assume that these numbers are dollars and that the entries in the cells reflect the amount paid by one player to the other. Positive numbers represent money received.

What might an equilibrium for this game be? One approach, the minimax approach, determines a player's choice based on minimizing the maximum damage that might be done by the opponent. Thus Player A will wish to avoid a loss of -1 and will choose action 2, while Player B will likewise attempt to avoid -1 and choose action 2. An alternative approach is to base the choice of an action on dominant strategies, that is an action that is best given any action of the opponent. Note that Player A is better off choosing strategy 2 regardless of the action choice of Player B, and similarly for Player B. A Nash equilibrium is one in which the play of each player is a best response to the play of the other player. Each player choosing action 1 could not be a Nash Equilibrium since Player A's best response to Player B's play would not be action 1. Player B choosing action 2 is a best response to Player B's choosing strategy 2 and vice-versa. In this example, the three different criteria governing strategy choice all lead to the same equilibrium outcome (each player choosing action 2), although this need not always be the case. Before illustrating this point we will elaborate on the fundamental properties of payoffs as well as introduce the concept of a mixed strategy

We will construe the numbers representing payoffs to have several properties. For any player a higher payoff is assumed to be preferred to a lower payoff. Furthermore, we assume that some transformations of the payoffs will not result in distortions. In particular positive linear transformations of the form, $f[x] = ax + b$, $a > 0$, will be assumed to not distort the inherent relationships between the payoff numbers. It should be clear that if in figure 1, one or both of the players payoffs were multiplied by 4 and 3 were subtracted, the same equilbrium would emerge regardless of whether the minimax, dominant strategy, or best response approach was used to find equilibria. Measuring payoffs is a topic in its own right and beyond the scope of the present paper. We will follow Von Neuman and Morgenstern and refer to these payoffs as utilities.

Figure 2 will be the basis for introducing the idea of mixed strategies. Figure 2 represents a two-player game, each player having two pure strategies. Player A has strategies $\{a_1, a_2\}$ and Player B has strategies $\{b_1, b_2\}$. The cells in the matrix represent utilities of the players that result from simultaneous choices. When A chooses a_1 and B chooses b_1, A gets x_{11} and B gets y_{11}. We assume that each player not only has the opportunity to a play single strategy, but can mix his strategies by assigning probability weights. We denote the sets of such mixtures for players A and B by

$$\begin{aligned} M_A &= \{(p_1, p_2)|p_1 + p_2 = 1, p_1, p_2 \geq 0\} \text{ and} \\ M_B &= \{(q_1, q_2)|q_1 + q_2 = 1, q_1, q_2 \geq 0\}. \end{aligned}$$

Pure strategies are mixed strategies with the entire probability mass on one strategy. The value of the game to player i, $V_i : M_A \times M_B \mapsto \Re$ (where $i \in$

A \ B	b_1	b_2
a_1	(x_{11}, y_{11})	(x_{12}, y_{12})
a_2	(x_{21}, y_{21})	(x_{22}, y_{22})

Figure 2: A Two-Player, Two-Strategy Game

$\{A, B\}$), is that player's expected utility given the strategies of both players. So for example, given $\{(\hat{p}_1, \hat{p}_2), (\hat{q}_1, \hat{q}_2)\}$, the value of the game to player A is $V_A((\hat{p}_1, \hat{p}_2), (\hat{q}_1, \hat{q}_2)) = \hat{p}_1\, \hat{q}_1\, x_{11} + \hat{p}_1\, \hat{q}_2\, x_{12} + \hat{p}_2\, \hat{q}_1\, x_{21} + \hat{p}_2\, \hat{q}_2\, x_{22}$. In a **Nash Equilibrium** , each player chooses his probability mixture to maximize his value conditional on the other player's selected probability mixture; in other words, his probability mixture is a best response to the other player's probability mixture. Thus, $\{(p_1^*, p_2^*), (q_1^*, q_2^*)\}$ is a Nash Equilibrium if and only if it satisfies

$$
\begin{aligned}
V_A((p_1^*, p_2^*), (q_1^*, q_2^*)) &\geq V_A((p_1, p_2), (q_1^*, q_2^*)) \quad \forall\, (p_1, p_2) \in M_A \text{ and} \\
V_B((p_1^*, p_2^*), (q_1^*, q_2^*)) &\geq V_B((p_1^*, p_2^*), (q_1, q_2)) \quad \forall\, (q_1, q_2) \in M_B
\end{aligned}
$$

A Nash equilibrium, $\{(p_1^*, p_2^*), (q_1^*, q_2^*)\}$, is totally mixed if and only if $\{(p_1^*, p_2^*), (q_1^*, q_2^*)\} \in \mathring{M}_A \times \mathring{M}_B$, where

$$
\begin{aligned}
\mathring{M}_A &= \{(p_1, p_2) | p_1 + p_2 = 1, p_1, p_2 > 0\} \text{ and} \\
\mathring{M}_B &= \{(q_1, q_2) | q_1 + q_2 = 1, q_1, q_2 > 0\}.
\end{aligned}
$$

In our notation, a strict dominant strategy, $\{(p_1^*, p_2^*), (q_1^*, q_2^*)\}$, is represented by

$$
\begin{aligned}
V_A((p_1^*, p_2^*), (q_1, q_2)) &> V_A((p_1, p_2), (q_1, q_2)) \\
&\quad \forall\, (p_1, p_2) \in M_A\,, (p_1, p_2) \neq (p_1^*, p_2^*)\ , \forall\, (q_1, q_2) \in M_B \text{ and} \\
V_B((p_1, p_2), (q_1^*, q_2^*)) &> V_A((p_1, p_2), (q_1, q_2)) \\
&\quad \forall\, (q_1, q_2) \in M_B\,, (q_1, q_2) \neq (q_1^*, q_2^*)\ \forall\, (p_1, p_2) \in M_A
\end{aligned} \tag{1}
$$

The minimax strategy is represented by

$$(p_1^*, p_2^*) = \arg \max_{(p_1,p_2)} \min_{(q_1,q_2)} V_A((p_1, p_2), (q_1, q_2))$$

$$(q_1^*, q_2^*) = \arg \max_{(q_1,q_2)} \min_{(p_1,p_2)} V_B((p_1, p_2), (q_1, q_2))$$

In the remainder of the paper the main focus will be on Nash Equilibria. This focus is motivated by the generality of the Nash concept. For example in zero-sum games, the case in which a pessimistic approach such as minimax seems most appropriate, the Nash concept and minimax concept lead to the same equilibria. In any game in which there is an equilibrium in dominant strategies, that equilibrium will also be a Nash equilibrium.

Examples:

Prisoners' Dilemma:

To illustrate the Nash Equilibrium concept, we present three games whose solutions increase in complexity.

Consider the Prisoners' Dilemma in Figure 3. Two software pirates are held in separate interrogation cells. They are suspected of having deleted a large amount of disk space prior to capture; however, enough evidence was recovered to convict each pirate for 6 months. Interrogators offer to cut the following deal with the prisoners: If each prisoner rats on the other, they both get 5 years. On the other hand, if one prisoner rats and the other

A \ B	not rat	rat
not rat	(.5,.5)	(10,0)
rat	(0,10)	(5,5)

Figure 3: Prisoners' Dilemma in Years.

A \ B	b_1	b_2
a_1	(-.5,-.5)	(-10,0)
a_2	(0,-10)	(-5,-5)

Figure 4: Prisoners' Dilemma in Utility.

doesn't, then the squealer gets off on probation while the other spends 10 years hard time.

Figure 4 represents the outcome in utilities rather than in years. Utilities represent an individual's preference for outcomes. A less preferred outcome receives a lower utility. In this game, it can be seen that the only Nash equilibrium is $\{a_2, b_2\}(\equiv \{(p_1 = 0, p_2 = 1), (q_1 = 0, q_2 = 1)\})$, since Player A would choose a_2 regardless of the choice of Player B; similarly, Player B would choose b_2. The choice of rat is a dominant strategy as well as minimax strategy for both players; however this is not a zero-sum setting.

		Jan: movie	Jan: football
Ed	movie	(2,1)	(0,0)
Ed	football	(0,0)	(1,2)

Figure 5: Battle of the Sexes.

Battle of the Sexes:

Now consider the "Battle of the Sexes" depicted in Figure 5. Ed, Player A, prefers movies to football, while Jan, Player B, says she prefers football to movies. They prefer being together to being separate. There are two obvious Nash Equilibria, $(movie, movie)$ and $(football, football)$, but there is a third, a mixed strategy equilibrium, $\{(\frac{2}{3}, \frac{1}{3}), (\frac{1}{3}, \frac{2}{3})\}$. To demonstrate that this is an equilibrium, assume that Ed chooses movie with probability $\frac{2}{3}$ and football with probability $\frac{1}{3}$. Then Jan's expected utility will be

$$V_B((\frac{2}{3}, \frac{1}{3}), (q_1, q_2)) = (2)\frac{1}{3}q_1 + \frac{2}{3}q_2 = \frac{2}{3}(q_1 + q_2) = \frac{2}{3}$$

Jan cannot affect her value of the game, so $(\frac{1}{3}, \frac{2}{3})$ is an optimal strategy for Jan. A similar argument leads Ed to choose $(\frac{2}{3}, \frac{1}{3})$, when Jan chooses $(\frac{1}{3}, \frac{2}{3})$.

To derive mixed strategy equilibria, we use the following fact: if $\{(p_1^*, p_2^*), (q_1^*, q_2^*)\}$ is a totally mixed strategy Nash equilibrium, (p_1^*, p_2^*) must make Player B indifferent across all mixtures of strategies in M_B and (q_1^*, q_2^*) must make Player A indifferent. Thus, $(p_1^*, p_2^*), (q_1^*, q_2^*)$ must solve

$$\begin{aligned} V_B((p_1, p_2), (1, 0)) &= V_B((p_1, p_2), (0, 1)) \text{ and } (p_1, p_2) \in \mathring{M}_A \text{ ; and} \\ V_A((1, 0), (q_1, q_2)) &= V_A((0, 1), (q_1, q_2)) \text{ and } (q_1, q_2) \in \mathring{M}_B \text{ .} \end{aligned}$$

$$\Longrightarrow (1)p_1 + (0)p_2 = (0)p_1 + (2)p_2 \text{ and } (p_1, p_2) \in \mathring{M}_A$$

Figure 6: Best Response Function for Jan and Ed

$$(2)q_1 + (0)q_2 \quad = \quad (0)q_1 + (1)q_2 \text{ and } (q_1, q_2) \in \mathring{M}_B$$

The only solution to these equations is $\{(p_1^* = \frac{2}{3}, p_2^* = \frac{1}{3}), (q_1^* = \frac{1}{3}, q_2^* = \frac{2}{3})\}$.

When there are only two players, each with 2 strategies, it is possible to graph (Figure 6) the best response functions of each of the players, and the intersection of the two functions represents the equilibria. None of the Nash Equilibria contain either dominant strategies or minimax strategies.

Modified Battle of the Sexes:

Now we modify the original "Battle of the Sexes" game, Figure 5., by changing the payoff to Ed from 0 to 2, when Ed chooses $football$ and Jan chooses $movie$.

The pure strategy equilibria of the game are the same as before, $(movie, movie)$ and $(football, football)$. The new twist this game illustrates is the existence of a continuum of (partially) mixed equilibria, where one player chooses a mixed strategy and the other player chooses a pure strategy. If Jan chooses $movie$, then Ed is indifferent among any mixture of strategies. The mixtures of Ed's strategies that cause Jan to choose $movie$ are (p_1^*, p_2^*) that solve

$$V_B((p_1, p_2), (1, 0)) \geq V_B((p_1, p_2), (0, 1)) \text{ and } (p_1, p_2) \in M_A.$$

		Jan: movie	Jan: football
Ed	movie	(2,1)	(0,0)
	football	(2,0)	(1,2)

Figure 7: Modified Battle of the Sexes.

$$\Longrightarrow \ (1)p_1 + (0)p_2 \geq (0)p_1 + (2)p_2 \text{ and } (p_1, p_2) \in M_A$$

So $\{(p_1, p_2), (1,0) | 1 \geq p_1 \geq \frac{2}{3} \text{ and } p_2 = 1 - p_1\}$ are Nash Equilibria.

Sessions:

Solving the Example Games:

Nash.m finds the pure, mixed, and any continuum of mixed Nash Equilibria for all two person Normal Form games with a finite number of strategies. The representation of the Prisoners' Dilemma Game in Figure 4 is `In[1]` in the following session.

```
In[1]:= game1={{{-1/2,-1/2},{-10,0}},{{0,-10},{-5,-5}}}

               1      1
Out[1]= {{{-(-), -(-)}, {-10, 0}}, {{0, -10}, {-5, -5}}}
               2      2
```

The payoffs to individual players can be extracted from `game1` by standard list operations. For example, `In[2]` illustrates how to extract player 2's (B's) payoff, given that player 1 (A) has chosen action 2 and player 2 has chosen action 1.

```
In[2]:= game1[[2]][[1]][[2]]

Out[2]= -10
```

In general, `game1[[i]][[j]][[k]]` gives player k's payoff, given that player 1 chooses i and player 2 chooses j. The Normal Form can be recovered by the command in `In[3]`:

```
In[3]:= MatrixForm[Transpose[game1]]

                    -0.5  -0.5    -10   0

Out[3]//MatrixForm=  0     -10     -5    -5
```

We use the program `Nash.m` to find the Nash Equilibria.

```
In[4]:= <<Nash.m

In[5]:= Nash[game1]

Out[5]= {{{0, 1}, {0, 1}}}
```

The output is the set of probability mixtures for players 1 and 2. Each player puts the full probability weight on the second strategy.

In the next part of the session, we find equilibria for the Battle of the Sexes game, Figure 5, and its extension, Figure 7.

```
In[6]:= game2={{{2,1},{0,0}},{{0,0},{1,2}}};

In[7]:= Nash[game2]
```

```
                                    2   1      1   2
Out[7]= {{{0, 1}, {0, 1}}, {{-, -}, {-, -}}, {{1, 0}, {1, 0}}}
                                    3   3      3   3
```

For games with only 2 strategies per player and mixed strategy equilibria Nash.m can present the graphical solution of the problem

```
In[8]:= Brgraph[{{{2,1},{0,0}},{{0,0},{1,2}}}]
```

```
Out[8]= -Graphics-

In[9]:= game3={{{2,1},{0,0}},{{2,0},{1,2}}};

In[10]:= Nash[game3]
```

```
                                     2   1
Out[10]= {{{0, 1}, {0, 1}}, {{-, -}, {1, 0}}, {{1, 0}, {1, 0}}}
                                     3   3
```

Notice that `Out[7]` is equal to the Nash Equilibria we previously found, but `Out[10]` is a subset of the Nash Equilibria we found. We can find the full set of Nash Equilibria by using the following rule: Any time one player plays the same strategy in two equilibria, then any convex combination of these two equilibria is also an equilibrium. Using `Out[10]`, `{{2/3, 1/3},{1, 0}}` and `{{1, 0},{1, 0}}` are equilibria where Player 2 has the same strategy. This implies that any convex combination of these strategies is also an equilibrium. In other words,

```
In[11]:= ans=%10;
```

```
In[12]:= t*ans[[2]]+(1-t)*ans[[3]]
```

```
                t   t
Out[12]= {{1 - -, -}, {1, 0}}
                3   3
```

`Out[12]` is a Nash Equilibrium for all t between 0 and 1.

In the package Nash.m the command Convex will transform observed solutions to solutions in terms of convex combinations directly.

```
In[13]:= Convex[{{{0,1},{0,1}},{{2/3,1/3},{1,0}},{{1,0},{1,0}}}]
```

```
                                2   1
Out[13]= {{{0, 1}, {0, 1}}, {{-, -}, {1, 0}}, {{1, 0}, {1, 0}},
                                3   3

           t1  t1               2 + t1  -(-1 + t1)
>    {{1 - --, --}, {1, 0}}, {{------, ----------}, {1, 0}},
           3   3                  3          3

           t2  t2               2 + t2  -(-1 + t2)
```

```
>    {{1 - --, --}, {1, 0}}, {{------, ----------}, {1, 0}},
           3   3                  3         3

       2 + t1 + t2 - 2 t1 t2  1 - t1 - t2 + 2 t1 t2
>    {{---------------------, ---------------------}, {1, 0}},
                 3                      3

       2 + t1 - t1 t2  1 - t1 + t1 t2
>    {{--------------, --------------}, {1, 0}},
             3               3

       2 + t2 - t1 t2  1 - t2 + t1 t2
>    {{--------------, --------------}, {1, 0}},
             3               3
```

The problem here is that the convex combinations are not in the most simplified form. To eliminate redundancies the function `ReduceSoln` is provided. The second argument of `ReduceSoln[]` corresponds to there being two "t"'s, t1 and t2, in the output of convex.

```
In[14]:= ReduceSoln[%13,2]

                                        2  1
Out[14]= {{{{0, 1}, {0, 1}}}, {{{-, -}, {1, 0}}}, {{{1, 0}, {1, 0}}},
                                        3  3

            t1  t1
>    {{{1 - --, --}, {1, 0}}}}
            3   3
```

The Package: Nash.m

```
BeginPackage["Nash`"]

Nash::usage ="Nash[game_] finds the Nash Equilibria of game, a game in
normal form. Example input: Nash[{{{2,1},{0,0}},{{0,0},{1,2}}}]. Nash
returns the probability weights on the different pure strategies."

IsNash::usage ="IsNash[game_,strategies_] returns True if strategies is
a Nash Equilibrium of game and False otherwise.
Example input: IsNash[game,{{2/3,1/3},{1/3,2/3}}]."


Brgraph::usage="Brgraph[game_,step_:.01] plots the best response graph of a 2 by 2 by 2
game. It plots the best response of player 1 on the x-axis given the action
of player 2 on the y-axis. It then plots the best response of player 2 on the
y-axis given the action of player 1 on the x-axis. The intersection points
of the two are the equilibria. Step is the interval size for plotting.
 (Note that this doesn't show the shaded area in the best response
correspondence when it exists in a continuum of equilibria .)"

Convex::usage="Convex[solns_] takes the solutions of Normal Form game
generated by Nash.m and generates the convex combinations that are
also Nash equilibria. Convex will output the entire set of Nash equilibria
of the original game; however, the output won't be in the simplest form.";
ReduceSoln::usage ="ReduceSoln[solns_,highestt_] eliminates redundant representations
of Nash equilibria from the output of Convex (solns). highestt is the highest
numbered t in the output of Convex. For example, t4 is valued 4."


Begin["`Private`"]

Convex[solns_]:=Block[{t1,i,x,t2,t3,t4,t5,t6,doit,conv1,convex,make,representation},


 representation[{z_, s_},i_] := ToExpression[StringJoin["t",ToString[i]]] z + (1 -  ToExpression


  make[{a_, b_}] := Function[x, x[[2]] == b || x[[1]] == a];

  convex[solutionset_] :=
    Table[Select[solutionset, make[solutionset[[i]]]],
    {i, 1, Length[solutionset]}];
```

```
        conv1[a_,b_,ii_]:=Table[Map[Function[x,representation[{a[[i]],x},ii]],
 b[[i]] ],{i,1,Length[a]}];

  conversion[solnns_,ii_]:=Union[Simplify[Flatten[conv1[solnns,
    convex[solnns],ii],1]]];


  doit[solnns_,0]:=solnns;
  doit[solnns_,x_]:=doit[conversion[solnns,x],x-1];


  doit[solns, Length[solns[[1,1]]]+Length[solns[[1,2]]]-2] ];

ReduceSoln[solns_,hightestt_]:=Block[{x,i,j,k,z1,endlist,expand},
  expand[x_,j_]:= Union[Table[x/.Table[
  ToExpression[StringJoin["t",ToString[i]]]->Mod[Floor[k/2^(i-1)],2],
  {i,1,j}],{k,0,2^j-1}]];
endlist=Map[Function[x,expand[x,hightestt]],solns];
uendlist=Union[endlist];
Table[solns[[Position[endlist,uendlist[[z1]] ][[1]]]],{z1,1,Length[uendlist]}]
  ];



Brgraph[game_,step_:.01]:=Block[{V,BR,l1,l2,pl1,pl2},
  If[Length[Dimensions[game]]!=3 || Dimensions[game][[1]]!=2 ||
    Dimensions[game][[2]]!=2, Return["The game is not a 2 by 2 by 2 list"]];

  V[i_,{p1_,p2_}]:={p1,1-p1}.Transpose[game,{2,3,1}][[i]].{p2,1-p2};


  BR[1,p2_]:=ConstrainedMax[V[1,{p1,p2}],{p1<=1},{p1}];
  BR[2,p1_]:=ConstrainedMax[V[2,{p1,p2}],{p2<=1},{p2}];

  l1=Table[{p1/.BR[1,p2][[2]],p2},{p2,0,1,step}];
  l2=Table[{p1,p2/.BR[2,p1][[2]]},{p1,0,1,step}];

  SetOptions[ListPlot,DisplayFunction->Identity];
  pl1= ListPlot[l1,PlotJoined->True];
  pl2= ListPlot[l2,PlotJoined->True];
  SetOptions[ListPlot,DisplayFunction->$DisplayFunction];

  Show[pl1,pl2,DisplayFunction->$DisplayFunction]
];
```

```
IsNash[a_,S_]:=Block[{l},
  l=Dimensions[a][[1]];
  Isnash[a,S]
];

Isnash[a_,S_]:=Block[{m1,m2,Eu,br1,br2,t},
   Eu[2,st_]:=N[S[[1]].a[[Range[1,l],st,2]]];
   Eu[1,st_]:=N[S[[2]].a[[st,Range[1,l],1]]];
   m1=Max[Table[Eu[1,t],{t,1,l}]];
   m2=Max[Table[Eu[2,t],{t,1,l}]];
   br1=Table[If[Eu[1,t]==m1,0,1],{t,1,l}];
   br2=Table[If[Eu[2,t]==m2,0,1],{t,1,l}];
   If[br1.S[[1]]+br2.S[[2]]==0,True,False]
];

Square[a_]:=Block[{n,l},
 n[i_]:=Dimensions[a][[i]];
 l=Max[n[1],n[2]];
Table[ If[i<=n[1] && j<=n[2],a[[i,j]],{Min[a]-1,Min[a]-1}],{i,1,l},{j,1,l}]]

Nash[a_]:=Block[{n,l,anew,MapD,Dropd,solns},

   If[Length[Dimensions[a]]!=3,Return["Not a two-player game!"]];
   n[i_]:=Dimensions[a][[i]];
   l=Max[n[1],n[2]];
   If[2!=Dimensions[a][[3]],Return["Payoffs aren't defined for two players"]];
   If[n[1]!=n[2],anew=Square[a];
      solns=Nash[anew];
      Dropd[i_][x_]:=Drop[x,n[i]-l];
      MapD[x_]:=MapAt[Dropd[1],MapAt[Dropd[2],x,{{2}}],{{1}}];
      Return[Map[MapD,solns]]
   ,Return[NashSq[a]]];

]
NashSq[a_]:=Block[{t1,t2,t3,l,p,pp,a1list,a2list,blist,f,pos,nq,pn,nlist,
   eqn1,eqn2,eqns1,eqns2,ans1,ans2,i,j,NashE},
   l=Dimensions[a][[1]];
   pp=Table[p[i],{i,1,l}]; a1list={}; a2list={};
   blist=Table[Mod[Floor[j/2^i],2],{j,1,2^l-1},{i,0,l-1}];
   For[t1=1,t1<=Length[blist],t1++,
   { num=Apply[Plus,blist[[t1]]];
     f[x_]:=If[Apply[Plus,x]==num,True,False];
```

```
    slist=Select[blist,f];
    For[t2=1,t2<=Length[slist],t2++,
    { pos=Flatten[Position[slist[[t2]],1]];
      eqn1=Table[(pp*blist[[t1]]).a[[Range[1,l],pos[[t3]],2]],{t3,1,num}];
      eqn2=Table[(pp*blist[[t1]]).a[[pos[[t3]],Range[1,l],1]],{t3,1,num}];
      eqns1=Table[eqn1[[i]]==eqn1[[i+1]],{i,1,num-1}];
      eqns2=Table[eqn2[[i]]==eqn2[[i+1]],{i,1,num-1}];
      ans1=Solve[Join[eqns1,{Apply[Plus,pp*blist[[t1]]]==1}],pp];
      ans2=Solve[Join[eqns2,{Apply[Plus,pp*blist[[t1]]]==1}],pp];
      AppendTo[a1list,Flatten[(pp*blist[[t1]])/.ans1]];
      AppendTo[a2list,Flatten[(pp*blist[[t1]])/.ans2]];
  }]}];
  nq[x_]:=Apply[And,Table[NumberQ[N[x[[i]]]],{i,1,Length[x]}]];
  pn[x_]:=Apply[And,Table[N[x[[i]]]>=0 && N[x[[i]]]<=1,{i,1,Length[x]}]];
  a1list=Union[Select[Select[a1list,nq],pn]];
  a2list=Union[Select[Select[a2list,nq],pn]];
  nlist=Flatten[Table[{a1list[[i]],a2list[[j]]},{i,1,Length[a1list]},
    {j,1,Length[a2list]}],1];
  INash[S_]:=Isnash[a,S];
  NashE=Select[nlist,INash]
];
End[]

EndPackage[]
```

Figure 8: The **Nash** package

Figure 8 displays the code of the package Nash.m. The procedures of the package Nash.m fall into one of two categories, housekeeping and computational. To insure correct inputs the procedure `Nash[]` determines if the tensor representation of the two person game has been input correctly. If the inputs are not a square game, Nash.m uses the procedure `Square[]` to convert a non-square game to a square game with an equivalent solution. If necessary, Nash.m converts the solution back to the solution of a non-square game. Also, with the procedure, convex, we give the user the chance to get a symbolic representation of all the partially mixed strategy games. To see how this works we use a mathematica session.

First let's consider an input that is not a two-person game.

```
In[1]:=Nash[random input]

Out[1]= Not a two-player game!
```

Now consider a different game,

```
In[2]:= a={{{2,3,4},{5,4,9}},{{4,5,6},{7,8,9}}}

Out[2]= {{{2, 3, 4}, {5, 4, 9}}, {{4, 5, 6}, {7, 8, 9}}}
```

`In[3]` and `Out[3]` indicate that if payoffs are input for more than 3 players then Nash will indicate that it is not a two player game.

```
In[3]:= Nash[a]

Out[3]= Payoffs aren't defined for two players
```

The following example shows how the procedure Square works to augment the game to be an equal number of strategies per player by adding strategies that would not be played.

```
In[4]:= a={{{2,3},{4,5},{6,1}},{{1,3},{2,9},{1,8}}}

Out[4]= {{{2, 3}, {4, 5}, {6, 1}}, {{1, 3}, {2, 9}, {1, 8}}}

In[5]:= Square[a]

Out[5]= {{{2, 3}, {4, 5}, {6, 1}}, {{1, 3}, {2, 9}, {1, 8}},

{{0, 0}, {0, 0}, {0, 0}}}
```

The computational procedures of the package are `IsNash[]` and `NashSq[]`. To demonstrate how `IsNash[]` and `NashSq[]` work we will mimic the actual code, but simplify it in such a way to better illustrate the key ideas of these procedures. First we consider `IsNash[]`. `IsNash[]` takes a game and a candidate for the equilibrium for the game and determines if the candidate is in fact an equilibrium. In what follows, `a` will be the game and `S` the candidate for an equilibrium.

```
In[6]:= a={{{2,4},{1,3}},{{3,5},{1,2}}}

Out[6]= {{{2, 4}, {1, 3}}, {{3, 5}, {1, 2}}}

In[7]:= S={{.5,.5},{.2,.8}}

Out[7]= {{0.5, 0.5}, {0.2, 0.8}}
```

To arrive at the conclusion that S is not a Nash Equilibrium, `IsNash[]` finds what would be the maximum that could be obtained by responding optimally to the other player‘s strategy.

First it computes the expected utility which would be achieved by responding with each of the strategies. The function defining this calculation is

```
In[8]:= Eu[1,st_]:=N[S[[2]].a[[st,Range[1,2],1]]];
```

For Player 1, the expected utility of choosing action1 given Player 2's choice of `S[[2]]` is

```
In[9]:= Eu[1,1]

Out[9]= 1.2
```

For strategy 2, the expected utility is

```
In[10]= Eu[1,2]

Out[10]= 1.4
```

We find the maximum by the following:

```
In[11]:= m1=Max[Table[Eu[1,t],{t,1,2}]]
```

```
Out[11]= 1.4
```

Next `IsNash[]` asks which pure strategies yield the maximum payoff to Player 1 in response to Player 2.

```
In[12]:= br1=Table[If[Eu[1,t]==m1,0,1],{t,1,2}]

Out[12]= {1, 0}
```

Note that those that yield the maximum receive weight zero. The reason for this is that we want to check which if any non-optimal strategies receive any positive weight in the candidate strategy. For example strategy 1 in game a recieves .5 from Player 1 which is not optimal The following calculation allows us to determine if non-zero weight is placed on an optimal strategy by Player 1.

```
In[13]:= If[br1.S[[1]]==0,True,False]

Out[13]= False
```

The procedure `IsNash[]` duplicates these operations for the player2 to attempt to establish mutual best response properties for each of the player strategies.

`NashSq[]` uses the principle that a player is indifferent between playing any pure strategy that is given positive weight in a mixed strategy that is a best response. `NashSq[]` considers every subset of pure strategies of Player 2 and asks what play of Player 1 would make Player 2 indifferent among playing each strategy in that subset. NashSq similarly determines a set of such strategies for Player 2. The Cartesian Cross Product of Player 2's set of such strategies with Player 1's become the sets of strategies tested by `IsNash[]`.

To see how this works we use the battle of the sexes game.

```
In[14]:= a={{{2,1},{0,0}},{{0,0},{1,2}}}

Out[14]= {{{2, 1}, {0, 0}}, {{0, 0}, {1, 2}}}
```

First, the number of strategies for each player is determined.

```
In[15]:= l=Dimensions[a][[1]]

Out[15]= 2
```

Then a list of symbols representing potential probability weights is generated.

```
In[16]:= pp=Table[p[i],{i,1,l}]

Out[16]= {p[1], p[2]}
```

The term, "blist", will generate a representation of the set of subsets of strategies. Since we are talking about a square matrix, such a representation will be the same for both players.

```
In[17]:= blist=Table[Mod[Floor[j/2^i,2],{j,1,2^l-1},{i,0,l-1}]

Out[17]= {{1, 0}, {0, 1}, {1, 1}}
```

Note here `{1,0}` would be the case when all the weight is placed the 1st strategy, `{0,1}`, all weight placed on strategy 2, and `{1,1}`, some weight placed on both strategies.

Thus there are as many equations to create to solve for `p[1]` and `p[2]` as there are subsets, namely 3 subsets.

```
In[18]:= Length[blist]

Out[18]= 3
```

NashSq works separately with pure strategies, mixtures with 2 pure strategies, mixtures with 3 pure strategies, etc. To identify which type of strategy to work with we used the function f,

```
In[19]:= f[x_]:=If[Apply[Plus,x]==num,True,False];
```

We will show how determining equations works for one member of "blist", namely the third member, since this is the mixed strategy. In what follows, num refers to the number of strategies which will receive positive weight, thus for mixed strategies it is two.

```
In[20]: num=2;

In[21]:= slist=Select[blist,f]

Out[21]= {{1, 1}}
```

The following two expressions calculate the expected utility to Player 2 for strategies 1 and 2 separately given Player 1 plays {p[1],p[2]}

```
In[22]:= pos=Flatten[Position[slist[[1]],1]]

Out[22]= {1, 2}
```

```
In[23]:= eqn1=Table[(pp*blist[[3]]).a[[Range[1,l],pos[[t3]],2]],{t3,1,num}]

Out[23]= {p[1], 2 p[2]}
```

Note that p[1] is the expected utility for Player 2 if Player 2 plays strategy 1 while 2 p[2] is the expected utility for Player 2 if Player 2 plays strategy 2.

For Player 2 to be indifferent these strategies must be set equal.

```
In[24]:= eqns1=Table[eqn1[[i]]==eqn1[[i+1]],{i,1,num-1}]

Out[24]= {p[1] == 2 p[2]}
```

We then add the restriction that p[1] + p[2] = 1 and solve for p[1] and p[2].

```
In[25]:= ans1=Solve[Join[eqns1,{Apply[Plus,pp*blist[[3]]]==1}],pp]

                     2          1
Out[25]= {{p[1] -> -, p[2] -> -}}
                     3          3
```

This is a candidate for a solution and Nash.m will also generate strategies Player 1 might play to get Player 2 to follow the pure strategies, and then repeat the process to find strategies Player 2 might play relative to all subsets of Player 1.

Performance:

Nash.m can solve two player games that have more than two strategies per player. For example, the game in Figure 9 has 4 strategies per player. The game is found in Harsanyi and Selten, and applying **Nash.m** reveals one pure equilibrium strategy they omit and numerous mixed strategies they omit.

Harsanyi and Selten (4×4):

	b1	b2	b3	b4
a1	(-1,-1)	(-1,-1)	(1,1)	(-1,-1)
a2	(-1,-1)	(-1,-1)	(0,2)	(0,2)
a3	(0,2)	(0,2)	(1,1)	(-1,-1)
a4	(0,2)	(0,2)	(0,2)	(0,2)

Figure 9: Harsanyi and Selten game

```
In[26]:= a={{{-1,-1},{-1,-1},{1,1},{-1,-1}},
            {{-1,-1},{-1,-1},{0,2},{0,2}},
            {{0,2},{0,2},{1,1},{-1,-1}},
            {{0,2},{0,2},{0,2},{0,2}}};

In[27]:= Nash[a]

                                                                1  1
Out[27]= {{{0, 0, 0, 1}, {0, 0, 0, 1}}, {{0, 0, 0, 1}, {0, 0, -, -}},
                                                                2  2

                        1     1
>    {{0, 0, 0, 1}, {0, -, 0, -}}, {{0, 0, 0, 1}, {0, 1, 0, 0}},
                        2     2

                     1        1
>    {{0, 0, 0, 1}, {-, 0, 0, -}}, {{0, 0, 0, 1}, {1, 0, 0, 0}},
                     2        2

>    {{0, 0, 1, 0}, {0, 1, 0, 0}}, {{0, 0, 1, 0}, {1, 0, 0, 0}},
```

Figure 10: Sun 4 Performance Graph

```
                                                       1   1
>    {{0, 1, 0, 0}, {0, 0, 0, 1}}, {{0, 1, 0, 0}, {0, 0, -, -}},
                                                       2   2

      1    2
>    {{-, 0, -, 0}, {0, 0, 1, 0}}, {{1, 0, 0, 0}, {0, 0, 1, 0}}}
      3    3
```

In general, the length of time required to solve a game by **Nash.m** increases exponentially in the number of strategies per player. We used **ShowTime** in conjunction with

```
game[n_]:=Table[{Random[Integer,10],Random[Integer,10]},{i,1,n},{j,1,n}].
```

`game[n_]` generates random two player games with n strategies and utilities ranging in the integers 0 to 10. Figure 10 graphs the natural log of the time Nash.m requires on a Sun 4 against the number of pure strategies per player.

Conclusion:

Nash.m should prove useful to both the novice, who is learning the theory and mechanics of discovering equilibria, and the expert, who can never be completely sure he has isolated all the equilibria of a complex game. Jack Stecher's and two anonymous referees' suggestions significantly improved this paper.

References

[Friedman, 1986] Friedman, J. W. (1986). *Game Theory with Applications to Economics*. Oxford University Press, New York, NY.

[Harsanyi and Selten, 1988] Harsanyi, J. and Selten, R. (1988). *A General Theory of Equilibrium Selection in Games*. The MIT Press, Cambridge, MA.

[Kreps, 1990] Kreps, D. M. (1990). *A Course in Microeconomic Theory*. Princeton University Press, Princeton, N.J.

[Nash, 1951] Nash, J. (1951). Non-Cooperative Games. *Annals of Mathematics*, 54:286–295.

[Rapoport, 1970] Rapoport, A. (1970). *Two-Person Game Theory*. University of Michigan Press, Ann Arbor, Michigan.

[Von Neumann and Morgenstern, 1953] Von Neumann, J. and Morgenstern, O. (1953). *Theory of Games and Economic Behavior*. Princeton University Press, Princeton, N.J.