Modeling Game Theory under Ambiguous Beliefs

First draft, 28 Mar 2004; revised 7 April 2004
Kenryo INDO
Department of Management, Faculty of Economics, Kanto Gakuen University

Abstract: This paper presented an experimental system for modeling the theory of decisions and games under ambiguous beliefs. It can be stated as the three-layer modeling of a decision maker who has ambiguous beliefs represented by the belief functions (BEL), will maximize the Choquet expected utility (CEU), and to play the Nash equilibrium under uncertainty (NEUU). By using Prolog, the author has developed the model base system which realizes the three-layer modeling approximately and guides the users through the simulation and analysis interactively.

1. Introduction†

Recently the game theory, and the Nash equilibrium, has been generalized to what analyses the strategic interactions by decision makers under ambiguous beliefs. It can be seen as the three-layer modeling which integrates the systems of belief, of decision, and of game. And each of them provides the modeling for incomplete knowledge, bounded rationality, and trust respectively. Mathematically these are applications of the real-valued set function theory of graceful formalization but the computation is awkward. So I think that these theories are in need of a modeling boon to motivate the modeler or the learner to work easier and to gain further insight .

I developed an experimental system which realizes the three-layer modeling (for 2 or 3 players) approximately. Prolog language which I used to develop the system has advantage in the modeling and simulation interactively. The model base which consists of the examples in the literature would be called by the system to take form of the model space. And the script programs in order to guide course of the experimentation and the analysis of generating beliefs and equilibria.

The paper is organized as follows. Next section reviews the theories of ambiguous beliefs and their applications for decision making and games. Then I refer to some advantages of using computer in game modeling. Section 3 discusses the system architecture. Sections 4 and 5 explain the theory of the three-layer modeling and display the examples of the simulation by turns. Section 6 discusses the handling of complexity. Section 7 concludes.

2. Modeling ambiguous beliefs and game theory

As a vein of the system sciences, the game theory accompanied with the expected utility (EU) theory which explains rational choices under risk, has been broadly applied to the fields of economics, management and information. The existence theorem of equilibrium point by J. Nash for N-player (standard form) games where each player permitted to use randomized strategies. A Nash equilibrium is a profile of probabilities the players use each of which is of maximizing the EU respectively.

For many years the EU model has been criticized by psychologists through the experimentation researches (as referred in the literature). For example an additional common prize reversed easily the choices of paired lotteries (Allais paradox). And ordinary people may either dislike or like unknown probabilities (Ellsberg paradox).

These phenomena have been called the `anomalies' against theory of EU and probability, including the above-mentioned. The Choquet EU (CEU) (Schmeidler, 1989) and the Maxmin EU (MEU) (Gilboa and Schmeidler, 1989) have been argued amongst many alternative theories proposed by researchers. Dow and Werlang (1994) proposed the Nash equilibrium under uncertainty (NEUU) or the equilibrium in beliefs for 2-player games played the CEU-maximizers. Eichberger and Kelsey (2000) has extended the N-player version.

In NEUU the basic uncertainty is modeled by using non-additive probability measure or belief function(BEL) and the ambiguity averse decision making by CEU. Lo (1996) used multiple prior (MP) and MEU instead of the models above. In NEUU safer option tends to be selected because of ambiguity aversion, and is not always a Nash strategy. For instance the cooperative play occurs in the finitely iterated prisoners dilemma game (`ipd2' in the model base) illustrated by Dow and Werlang.

The role of computer in game modeling

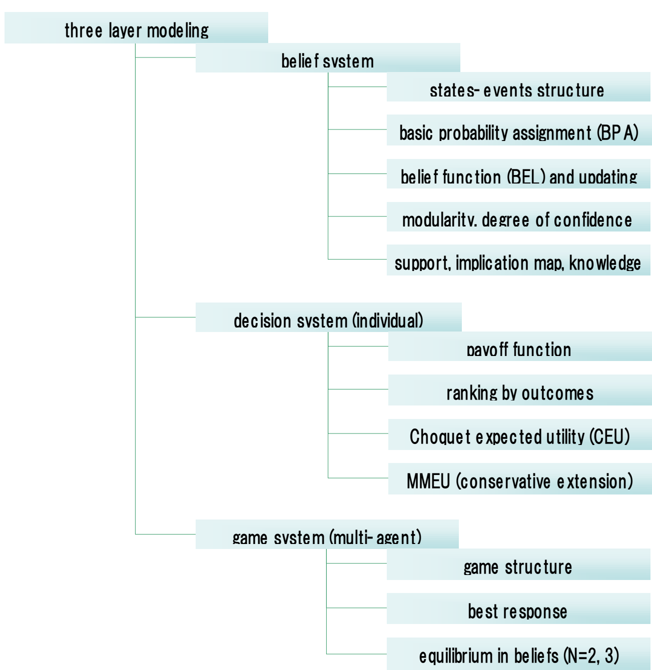

Figure 1. The three-layer modeling.

† This paper is the excerpted translation of what originally in Japanese to appear in the proceedings of JASMIN Spring meeting 2004.

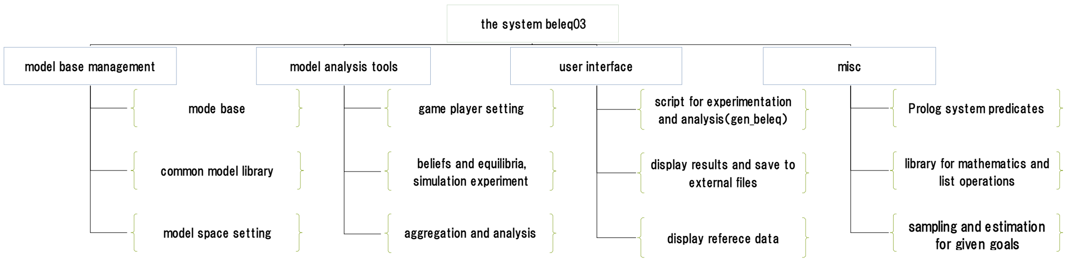

Figure 2. Components of the experimental system.

Modern computer technology is useful for modeling game theory. Let me point out three uses as follows.

- Computing equilibrium points
- Simulation of plays and visualization
- Exact /Approximate modeling of the logic

The first has been investigated from the simplex algorithm in 1950s to the homotopy method recently. The second computes the dynamics of the game play, as well as the equilibrium point, and visualizes them by the graphic functions. For example, spreadsheets in my homepage(Indo, 2004) show you the fixed point, attractor, and other adjustment processes of Nash's continuos map for 2 x 2 games and the chaos in Cournot's duopoly.

Whereas the above mentioned are utilizing the functions of arithmetic and graphics by computer, you can step into the modeling of mutual cognition by using, Prolog, a representative logic programming. It may be useful for the theoretical verification, the learning, and the prototyping of the real decision systems.

3. System architecture and developmental data

The experimental system supports the three-layer modeling in the theory of game under ambiguous beliefs (see Figure 1) to be explained particularly in next section.

(1) Modeling ambiguous beliefs or imprecise knowledge by belief function (BEL) and basic probability assignment (BPA),
(2) Modeling bounded rationality by Choquet expected utility (CEU) maximization, and
(3) Modeling game theory under uncertainty and the solution (NEUU) approximately and trust.

The system can be seen as a small normative expert system or model base system which has coded by Prolog language accordingly. And the system logically consists of the model base management tools, the model analysis tools by simulation, the user interfaces, as an example, the script program gen_beleq/0 which controls the experimentation and supports the analysis, and other common program library (see Figure 2). The source code file (beleq03.pl) is of less than 8000 lines including the comments for codes and program demos. When consulted from SWI-Prolog 5.0.9 on my notebook PC (TM5800 993MHz, 232MB RAM, Windows XP) the compiled code amounts to 365,592 bytes.

The development time was about 2 months, 2004/1/14 —3/18, with the debug history which has noted in the file. But the first two layers reuse some existent codes which I wrote previously. The part of belief function (belief0, about 1100 lines, 28,544 bytes, 2003/3/1) was reported partly in the proceedings of the last spring meeting of JASMIN, and its expanded part of Choquet expected utility (belief01, about 1400 lines, 36,292 bytes, 2003/6/29) has augmented after that meeting. These complete code are downloadable from my homepage (Indo, 2004).

4. The three-layer modeling

- Belief system (the first layer)

The modeling starts the belief system with the situation of uncertainty the decision maker faces with. Mathematically the algebra (which consists of the whole states, Ω, the events, $E \subseteq \Omega$, and their operations over events, $\subseteq$, $\cup$, $\cap$, ^{c}) and the non-additive probability measure:

$$v(A) = \sum_{B \subseteq A} m(B),$$

where m(A) represents the basic probability assignments (BPA) or called the mass, which is additive probability measure over the set of all events and assigns 0 to the empty set, $m(\phi) = 0$. Then the function v becomes a 0-1 normalized totally monotone capacity, also called a belief function (BEL)(Shafer, 1976). And $1\text{-}v(A^c)$ is called a plausibility function (PL). Thus our basic models are states, events, BPA, BEL, PL, and their updating (conditioning) rules respectively. And the inversion formula which computes BPA from BEL, the judgment of (super/sub-)modularity. A super-modular (2-monotone or convex) capacity v satisfies

$$v(A \cup B) + v(A \cap B) \geqq v(A) + v(B).$$

Further the support and the indices of the confidence and the ambiguity are useful in the game analysis to be explained later.

- Decision system (the second layer)

For an act, let the ranked utilities of the possible outcomes, $u_1 \geqq \ldots \geqq u_m$, and $A_0 = \phi$, $v(A_0) = 0$. Choquet expected utility (CEU) is the Choquet integral with respect to the convex capacity--- it can be seen as the BEL in the modeling ---defined as follows:

$$\int f dv = \sum_{k \in M} [\, v(\cup_{j \leq k} A_j) - v(\cup_{j < k} A_j)]\, u_k,$$

where each A_k and u_k (k=1,...,m) is the k-th event and the utility in the ranking respectively.

CEU can be interpreted intuitively as the equivalent max-min expected utility (MMEU) by the Mean of Min /

Min of Mean Theorem (Gilboa and Schmeidler, 1994) or the conservative extension below.

$$\int f dv = \Sigma_{B\subseteq\Omega}\ m(B)\,[\min_{\omega\in B}\ f(\omega)]$$

$$= \min_{p\in Core(v)} \Sigma_{\omega\in\Omega}\ p(\{\omega\})\, f(\omega).$$

The MMEU is the worst EU when all the free flow in each event, other than the committed mass (already assigned) by the BEL, assigned to the worst outcome among that event, and therefore is the worst EU in the core of the BEL.

- Game system (the third layer)

Let I={1, ..., N} the finite set of players. For each i∈I let v_i the BELs or the convex capacity over the product of the strategy spaces other than i which represents the belief of player i what the other players would do in the game. The support of v is defined as an event $E\subseteq\Omega$ such that

$$v(E^C)=0,\ F^C\subseteq E^C,\ \ F\neq E \Rightarrow v(F^C)>0.$$

Since v is monotone, it is the minimal event whose complement is of measure 0. Assuming that $R_j(v_j)$ denotes the set of CEU maximizing act of player j against own belief. NEUU is defined as the profile $(v_i)_{i\in I}$ which satisfies

$$\exists SUPP(v_i) \subseteq X_{j\neq i} R_j(v_j) \quad \forall i\in I,$$

where SUPP denotes the support of v_i. That is for each player i there is at least one support of her belief system such that all k-th element in the support is a best response of the opponent k.

Epistemically the player `knows' any event which includes the support. Therefore the above condition states that the profile of best responses is `mutually known.'

5. Example of the simulation and analysis

Several examples of the three-layer modeling in the literature has stored in the model base. The model space wold be reset when the user choose a model to be analyzed. The model space consists of the predicates in the common model classes (i.e., states/1, event/1, bel0/2, bpa0/2, payoff0, game/4, and so on) of each layer and the proper ones in the model. The model base contains information which common model classes to be used in the model, and the system refers this to instantiate the model space.

Gen_beleq/0 a script program provides the multiple choice menu to support the three-layer modeling as described in Figure 1 and to execute each step of the experimentation and the ex post analysis smoothly by using the system resources in Figure 2. The tentative experimentation data during generating beliefs with the supports and the best responses to be saved as temp_ceu_max_play/4 for each player in the objective game. A NEUU is the direct product of them. A sample output of gen_beleq which is displayed in Figure 3. The example is a 2 x 2 strategies standard form game, named `prudence', which is equivalent as Figure 2 in Dow and Werlang(1994).

A model base(model_base: prudence, a=1, e=2)

```
model:prudence
states([f, c])
bel0([], 0)
bel0([f], 0.3)
bel0([c], 0.7)
bel0([f, c], 1)
act(f)
act(c)
payoff0(f, f, 8)
payoff0(f, c, 8)
payoff0(c, c, 10)
payoff0(c, f, -10)
game(prudence, parameter, payoff(a), 1)
game(prudence, parameter, payoff(e), 2)
game(prudence, payoff, [c, c], [10, 10])
game(prudence, payoff, [f, f], [8, 9])
game(prudence, payoff, [c, f], [-10, 9])
game(prudence, payoff, [f, c], [8, 10])
```

```
%          f          c
%     +---------+---------+
%     |  10-e   |  10-e   |
%  f  |  10-a   |   10    |
%     +---------+---------+
%     |   -10   |   10    |
%  c  |  10-a   |   10    |
%     +---------+---------+
% Fig. prudence game.
```

Computing CEU

```
?- payoff(choquet(A),B,C).

A = f
B = [[c, f], [8, 0], [8* (1-0), 0]]
C = 8 ;

A = c
B = [[c, f], [-10, 10, 0], [-10* (1-0.7), 10* (0.7-0), 0]]
C = 4.0 ;

No
```

Experimental results (script program: gen_beleq)

```
model:prudence
number of bpa intervals:20
restricted events for generating positive-valued
bpas:non

filters on belief indices:
 confidence:[0, 1]
 ambiguity:[0, 1]

[1]
acts:[[f], [c]]
supports:
% player(1):[[c]]
% player(2):[[f]]
equilibrium belief(bpa)s and their intervals:
% player(1):[[c]]:[[0, 0.85]]
% player(2):[[f]]:[[0, 1]]
confidences:
% player(1):[0, 0.85]
% player(2):[0, 1]
ambiguities:
% player(1):[0.15, 1]
% player(2):[0, 1]
```

← If the degree of confidence is less than 0.9 then ([f], [c]) is a unique best response profile.

```
[2]
acts:[[c, f], [c]]
supports:
% player(1):[[c]]
% player(2):[[c]]
% player(2):[[c], [f]]
% player(2):[[f]]
equilibrium belief(bpa)s and their intervals:
% player(1):[[c]]:[[0.9, 0.9]]
% player(2):[[c]]:[[0, 1]]
% player(2):[[f]]:[[0, 1]]
confidences:
% player(1):[0.9, 0.9]
% player(2):[0, 1]
ambiguities:
% player(1):[0.1, 0.1]
% player(2):[0, 1]
[3]
acts:[[c], [c]]
supports:
% player(1):[[c]]
% player(2):[[c]]
equilibrium belief(bpa)s and their intervals:
% player(1):[[c]]:[[0.95, 1]]
% player(2):[[c]]:[[0, 1]]
confidences:
% player(1):[0.95, 1]
% player(2):[0, 1]
ambiguities:
% player(1):[0, 0.05]
% player(2):[0, 1]
```

Figure 3. A model base and the simulation results of generating the equilibria in beliefs.

```
condition_of_equilibrium_in_beliefs_2(J,S,SP,R,yes):-
    member(SP,S), % S: the set of supports of player J.
    forall( member(X,SP), product_of_lists([R],X) ).
condition_of_equilibrium_in_beliefs_2(_,_,_,_,no).

equilibria_in_beliefs_2([P1,P2],R,Y):-
    R=[R1,R2],
    Y=[yes,yes],
    setof((BP1,SB1),temp_ceu_max_play(1,BP1,SB1,R1),Q1),
    setof((BP2,SB2),temp_ceu_max_play(2,BP2,SB2,R2),Q2),
    setof((BP1,S1,SP1),
     (
      member((BP1,S1),Q1),
      condition_of_equilibrium_in_beliefs_2(1,S1,SP1,R2,yes)
     ),
    P1),
    setof((BP2,S2,SP2),
     (
      member((BP2,S2),Q2),
      condition_of_equilibrium_in_beliefs_2(2,S2,SP2,R1,yes)
     ),
    P2).
```

Figure 4. The modeling of NEUU for two-player

In NEUU the more degree of ambiguity aversion, the safer option is chosen. So you can use the indices of confidence and of ambiguity for filtering the equilibria (Eichberger and Kelsey, 2000).

$$\text{confidence} = \max_{E \neq \phi, \Omega} \{ v(E) + v(E^C) \},$$

$$\text{ambiguity} = \max_{E \neq \phi, \Omega} \{ 1 - v(E) - v(E^C) \}.$$

The user can set the upper and the lower bounds of intervals for each indices respectively. Figure 3 presents that there's no active filter but, [0,1], the default. However you may easily see that if the confidence index was below 0.9 the experimentation had resulted in only the pattern [1] of equilibria where ([f], [c]) is the best response profile.

As suggested in the literature that NEUU can interpret the `trust' between players. The column (player 2 with the payoffs in lower) has a strategy c that dominates f, and row know this. Clearly c is the best response if it is certain that column choose c. Thus because of the slight doubt about the rationality the row (player 1 with the payoffs in upper) select the safer nevertheless.

6. Coping with complexity

Any naïve modeling of ambiguous beliefs or game theory would make the computation intractable even if you use a modern computer technology. I have tried to contrive the system to ease the computational complexity or the cognitive load of user as follows.

1) Separating common model classes and local models
2) Making it factual in model space
3) Input rules for events
4) Intervals for beliefs and indices
5) Wrapping equilibria beliefs
6) Predicting complexity

(1) Originally the values of BPA and BEL are defined over events. In the model base these correspond to bpa0/2 and bel0/2 which should be defined explicitly only over the focal elements (i.e., the events with positive valued BPA). Similar to the NAF (negation as failure) in standard Prolog systems, by interpolation according to the rules bpa/2 and bel/3 in the common model class bpa has measure 0 if bpa0 fails.

(2) Such like as Prolog systems, rule-based inference which uses recursion with backtrack are double-edged sward. The model compiling in my system makes the successful goals into the facts according to the common model class, for instance, bel/3. Then model would be defined correctly in the model space and not to cause unintended backtracks.

(3) The rules that have an event in the argument should be used properly. If the variable is unbound it would be the sorted states generated by using sevent/1 according to the occurrence in states/1 the list of all states. Otherwise, any permutation of the event to be allowed.

(4) According to the precision (the number of intervals) the user specified, the system classifies equilibria by best responses profile and then aggregates both the beliefs (BPAs) and these indices into intervals approximately. In Figure 3, the player 1 has no intervals for event [f] of the pattern [3] in Figure 3, since only the non-trivial focal elements to be displayed.

(5) Even a very simple game has enormous number of NEUUs. (For example `lo32' a 3-player game, the approximated equilibria with 2 intervals amounts to 440000, nevertheless there are only two substantial patterns.) As shown in Figure 3, the wrapped version of NEUU (by using setof/3) corrects equilibria substatialy the same unpacked.

(6) In order to estimate the execution time and advise the user, the combinatorial theory and the statistical simulation may be useful. In the system such intelligent interactive interfaces have realized partially in the generating BPA and NEUU and the test of modularity.

7. Concluding remarks

We discussed an experimental system which supports the three-layer modeling of NEUU approximately. The extensive form remained. I gratefully acknowledge SWI-Prolog the free software provided by J. Wilemaker and Amsterdam University to develop my system.

References

[1] Indo, K., Simulating theoretical models of decision making by Prolog and Spreadsheet. http://www.us.kanto-gakuen.ac.jp/indo, 2004.
[2] Dow, J. and S.R.C. Werlang, Nash equilibrium under Knightian uncertainty: breaking down backward induction, Journal of Economic Theory, 64:305-324, 1994.
[3] Eichberger, J. and D. Kelsey, Non-additive beliefs and strategic equilibria, Games and Economic Behavior, 30:183-215, 2000.
[4] Gilboa, I. and D. Schmeidler, Maxmin expected utility with non-unique prior, Journal of Mathematical Economics, 18:141-153, 1989.
[5] ______ and ______, Additive representations on nonaddtitive measures, Annals of Operations Research, 52:143-65, 1994.
[6] Lo, K.C., Equilibrium in beliefs under uncertainty, Journal Economic Theory, 71:443-484, 1996.
[7] Schmeidler, D., Subjective probability and expected utility without additivity, Econometrica, 57:571-587, 1989.
[8] Shafer,G., Mathematical Theory of Evidence, Princeton University Press, 1976.