Physica D 147 (2000) 221–258

PHYSICA D

www.elsevier.com/locate/physd

Dynamical systems game theory and dynamics of games

E. Akiyama [a,*], K. Kaneko [b]

[a] *Santa Fe Institute, 1399 Hyde Park Road, Santa Fe, NM 87501, USA*

[b] *Graduate School of Arts and Sciences, The University of Tokyo, Komaba 3-8-1, Meguro-ku, Tokyo 153-8902, Japan*

Received 27 March 1999; received in revised form 25 February 2000; accepted 28 June 2000

Communicated by M. Sano

Abstract

A theoretical framework we call dynamical systems game is presented, in which the game itself can change due to the influence of players' behaviors and states. That is, the nature of the game itself is described as a dynamical system. The relation between game dynamics and the evolution of strategies is discussed by applying this framework. Computer experiments are carried out for simple one-person games to demonstrate the evolution of dynamical systems with the effective use of dynamical resources. © 2000 Elsevier Science B.V. All rights reserved.

PACS: 02.50.Le

Keywords: Dynamical system; Coupled optimizers; Evolutionary game

1. Problems in game dynamics

There are three purposes in the present paper and two subsequent papers. First, we attempt to construct a theoretical framework capable of describing the dynamic aspects of game environments and of agents. We call this framework the dynamical systems (DS) game. Second, we illustrate this abstract framework concretely with computer simulations to demonstrate some characteristic features of the models of DS game. Third, a novel viewpoint is provided from the standpoint of DS game, regarding the emergence of social phenomena through interactions between individuals, the development of communication among agents, and evolutionary phenomena in colonies of living organisms.

The main content of this series of papers is as follows. In the present paper, we first give the general formulation of DS game framework, and treat the issues on one-person DS games in which the game dynamics is under the control of the only player's decision. In the second paper, we deal with the issues of multiple-person DS games, especially focusing on the cooperation engaged in by players who use stable game dynamics as a norm for cooperation. There we will reveal the mechanism that enables the formation of cooperation among the players, who grow and acquire common dynamical resources. This formation is commonly observed in our real world, but it cannot be represented in the traditional game theory framework. In the third paper, we see the evolutionary phenomena of societies with the evolution of the reference of each decision maker toward others' states. Through mutual interactions, players form several rules for administration of the dynamical resources. For example, we can see the formation of a rule under which each player

* Corresponding author. Tel.: +1-505-984-8800 (ext. 297); fax: +1-505-982-0565.

E-mail address: eizo@santafe.edu (E. Akiyama).

0167-2789/00/$ – see front matter © 2000 Elsevier Science B.V. All rights reserved.

PII: S0167-2789(00)00157-3

is allotted a certain resource to manage and does not rob others of their resources (private resource). As another example, rules for the joint administration of common resources are often constructed. Such rules adopted by players change with the evolution of strategies regarding the reference to others' states.

1.1. Discussion of fundamental problems

If two or more decision makers trying to obtain optimum results interact with each other, the result for each one will depend in general not merely upon his own actions but on those of the others as well. In the theory of games, von Neumann and Morgenstern [12] had insight into the fact that this problem is not an ordinal maximum problem and that the formalization of this type of problem should take the form of matrix. They succeeded in characterizing the problem as one in which every individual "can determine the variables which describe his own actions but not those of the others", while, in addition, "those alien variables cannot, from his point of view, be described by statistical assumptions" [12].

However, game theory is not congenial to problems involving dynamical phenomena involving multiple decision makers due to the static nature of the matrix employed in conventional game theory. There are mainly two issues that we would like to consider here. The first one regards the effect that a player's actions can have on the game environment. The actions selected by any one player will certainly have an effect on the actions of others. In reality, however, it is also possible that a player's actions can affect the actual game environment itself. Through this influence, the actual game in which the player is involved can also change. Then, through such changes in the game environment, the payoffs for a player's actions may also be changed.

In addition to questions involving the effect of a player's action on the game environment, we wish to consider the issue of the connection between a player's payoff function and that player's state. (We use the word "state" here to mean any general internal properties of a player that may change, the actual condition of the player or the internal model of the outside world that the player has.) For example, consider a player participating in a contest repeatedly with the same opponents in a game environment that does not change with time. In this case, will the utility of the player's possible actions always continue to be the same? Won't the player's assessment of his possible actions vary in accordance with changes in his internal state?

Further, we would like to touch here upon the fundamental viewpoint of traditional game theory with regard to the above-mentioned situation. In traditional game theory, such a situation is sometimes represented by one (large) game. That is, from the present into the future, all possible actions of all players at all points in time are taken into account. Thus all possible bifurcation patterns of the game are derived with this situation as a whole depicted as one huge game tree. In this way, we can project the course of time into a static game and analyze its solution in the form of a game-tree or a game matrix. Strategy here means the action plan for all points in time, and the analysis of a rational solution for a game is possible only when we know all the possibilities about all players' actions from the past to the future. However, there arises an important question here: Do we always make our decisions in this way? Moreover, is it even possible for us to make such a decision in the first place?

These types of problems were first considered by Rapoport [15] and Rashevsky [16]. Later, Rössler [18] considered them using an abstract model of multiply linked coupled autonomous optimizers. These have recently been developed by Ikegami and Taiji [8,22]. Let us examine the above questions in more detail in relation to game theoretical research on evolutionary systems.

1.2. Static games and dynamical systems games

Any interaction with multiple decision making agents (players) that have different interests is called a game. Basically, game theory investigates the criteria for all the players' rational behavior. In this context, one of the most important goals these days is to use Nash's concept of equilibria to carry out equilibrium analysis. Such analysis of the equilibria of games has produced considerable results to date, especially for non-cooperative games. Theories developed for this class of games have also been extended to produce further results in the analysis of cooperative games.

It is unquestionable that this kind of equilibrium analysis is extremely effective, and that it is indispensable as a basic theoretical technique. However, when we think of real-life groups of living organisms or of social phenomena, it seems that there are examples where other approaches would be more effective. In particular, dynamic change rather than equilibrium is inherent in a large number of situations. For instance, in the real world it is not unusual that communications and mutual interactions between individuals, as well as the prevailing social structures, continually change so that a final equilibrium condition is never reached. It is not appropriate to treat such phenomena as a simple transition towards a theoretical equilibrium condition (i.e., viewing them as evolutionary learning processes), since the incomplete computation power available to individual agents will always result in some irrational decisions being made. To give a simple example, a viewpoint distinct from equilibrium analysis is clearly important for understanding the nature of the so-called open-ended evolution.

In the past, evolutionary game theory [10] has offered various verifiable hypotheses and explanations about the nature of social and ecological evolution. This type of analysis has suggested many important theories about the nature of learning, imitation, and cultural/hereditary inheritance. In more recent years, computer simulation of evolutionary process has been adopted in the study called *artificial life*. In this type of research, the general process of evolution is treated as a game. Games are played by using decision making programs representing players. The objective of such research is to understand the general process of evolution through computer experiments.

Now, in these studies on evolutionary systems, the adopted games do not change over time, by definition. In other words, the evaluation functions of the players in the games remain constant. Examples are one-shot games, like the Hawk–Dove game, and also games like the iterated prisoner's dilemma in which the same game is iterated many times. Because the games themselves in this case are static in nature, it could even be said that one of the main reasons for the success of these studies is the "static" representation of the games. For convenience, in this paper, let us refer to this form of game representation as the static game representation.

Certainly, if we think of social phenomena in the real world, there are many examples that can be investigated very effectively within the framework provided by static game representation. However, on the other hand, examples of phenomena that cannot be completely analyzed in this framework also exist. This is because the game environment of the real world that we actually live in is not fixed, and changes with every action that is carried within a strategy or any change in the value judgment system used by a player.

In real-life games, the mutual interactions between the game environment and the players can have a large effect. For example, in systems that model real-life ecosystems, the use of a given strategy by a particular individual may cause changes in the outside environment and in the game's payoff matrix itself. Further, the local payoff matrix between two individuals may change depending on the strategy chosen by any one of possible third parties.

Moreover, in real-world games, the utility attached by a player to a certain strategy may change according to the player's current state. To give just a simple example, the action of eating a banana growing on a tree has quite different utilities depending on the player's current state of hunger. This becomes more complicated when we also take into account the way that the player views the condition of the banana and the way that the player views the states of the other players nearby. Here, the way that the player views the situation can completely change the utility of the possible actions. Even for bananas of the same size, thoughts such as "this banana can still grow larger", or "this banana has probably stopped growing", or "now that it's this big, the other people nearby will take it soon", will by themselves alter the utility of the action "eating bananas". At this point, let us discuss in more detail the nature of the possible mutual interactions between strategies and the game environment. Static games mainly involve the strategies of single players, or of groups of players (in situations such as cooperation, treachery, conspiracy or co-evolution) with respect to a fixed game definition that persists until the game is over. Such games do not lend themselves to answering questions such as “What effect do the game dynamics have on the evolution of strategies?”. For example, the game environment may be oscillatory or even chaotic, and the evolution of strategies in these circumstances will be very different. Of course, it is also very difficult to use static games to naturally treat the complementary problem of how a player’s actions can produce game dynamics. For example, if a player lives by producing and consuming resources in the environment, one cannot determine using a static game description what strategies one should adopt in order to make the dynamics of such environments productive or to keep them stable.

The most clear and simple example illustrating how a player’s actions can change a game itself is that of delay effects [19]. In the real world that we inhabit, our rationality cannot be so perfect as that game theory requires. In fact, for games that continue from the past into the future, we can identify the following prerequisites for a player to make rational decisions.

1. Perfect knowledge on the rules of the game.
2. The ability to enumerate all the possible actions, one of which the player may choose at a given point in time.
3. The ability to identify all the possible situations that these actions would generate at the next point in time.
4. The ability to construct a complete representation of the game from the past to the future as a branching diagram (e.g., represented in the extensive form).
5. The ability to completely analyze this (typically) huge game tree and derive an equilibrium.

For even very small games, meeting these prerequisites is impossible in practice. (See the discussion of bounded rationality [2,21] that we touch on briefly later.) Even for the game of chess, where constructing the optimal process of inference is the main strategic goal, it is impossible to practically carry out decision making in the above manner. (Or, if it were possible, the result of the contest would be known from the beginning, so that there would be no meaning in playing the game [23].)

In real-world games, players make decisions on the basis of conclusions obtained by examining as large a local game as they can practically handle. However, in a game that can change over time, the action that appears optimal in the current environment may not be optimal when the game is viewed from a long-term perspective. For example, by applying restraint at the current point of the game, it is possible to influence the game environment of the distant future so that it offers very rich payoffs. In this way, the present decision of a player can have an effect on the future game environment; this is what we call the “delay effect”. Clearly, static game models cannot treat systems with such properties. There is a fundamental difference between the two strategies “cooperate if the opponent cooperated last time, otherwise aim for a high payoff” and “acquire a high payoff by forcing the game itself to change”.

We introduce the model what we call dynamical systems game model to handle the types of situations discussed above. The most straightforward description of this model is that it represents a game as a dynamical system in which the state of the game environment, the states of all the players, and the players’ possible actions vary in time, and all of them are described by dynamical variables.

2. General description of the dynamical systems game model

In this section, we demonstrate the basic concept of the DS game model and give a formulation of it. First, let us consider a model game.

2.1. An example of a dynamical systems game

The lumberjacks’ dilemma (LD) game. There is a wooded hill where several lumberjacks live. The lumberjacks cut trees for a living. They can maximize their collective profit if they cooperate by waiting until the trees are fully grown before felling them, and sharing the profits. However, the lumberjack who fells a given tree earlier than the others takes the entire profit for that tree. Thus each lumberjack can maximize his personal profit by cutting trees earlier. If all the lumberjacks do this, however, the hill will go bare, and there will eventually be no profit. These circumstances inevitably bring about a dilemma.

This LD game can be categorized as representing the social dilemma that arises in connection with the problem of forming and maintaining cooperation in a society, which is often represented by the classical story "the tragedy of the commons" presented by Hardin [7]. Its structure is logically similar to that of the prisoners' dilemma model if considered at the level of a static game. In other words, it can be represented in the form of an n-person version prisoners' dilemma if we project it onto static games.

Here we note several important differences in modeling social dilemma between the lumberjack's dilemma and the n-person prisoners' dilemma. The dynamics of the sizes of trees should be expressed explicitly in the lumberjacks' dilemma. The yield of a given tree, and thus the lumberjacks' profit, differs according to the time when the tree is felled. The profits have a continuous distribution, because the possible yield of a tree has continuously varying value. Thus the lumberjacks' actions cannot be definitely labeled as "cooperation" or "betrayal". Furthermore, a lumberjack's decision today can affect the future game environment through the growth of a tree.

2.2. The concept of dynamical systems games

In a DS game, players live in a certain game environment and always have several possible actions they can take. The game dynamics, g, is composed of the following three system defining rules:

1. The states of the players' surroundings (which we call the "game environment"), x, and the states of all the players, y, change according to a natural law.
2. Players make decisions according to their own decision making mechanisms, f, by referring to both the states of the players' surroundings and of all the players (including oneself).
3. Change in the game environment and players' actions affect the states of the players.

Players repeatedly carry out actions, and in the process, the system evolves according to these general rules. Using the formulation described by these rules, instead of that based on algebraic payoff matrices, DS game model can explicitly describe not only game-like interactions but also the dynamics observed in the game environment and among the players. The total game dynamics is described by the map, g, and the players' decision making, f, is embedded into g. The dynamics is expressed either in discrete time fashion, as iterated mappings or in continuous time fashion, by differential equations.

In the DS game, a player's reasoning capacity is inevitably limited. As stated in Section 1.2, players in the DS game can neither examine all possible developments in a game as a huge branching diagram nor compute an equilibrium from this huge diagram before taking action. By representing the chain of causal relationships as one huge game, analysis of traditional game theory from a static viewpoint is made possible. On the other hand, computation of such branching diagrams is impossible in the DS game, since all the states are represented by continuous variables, and the nonlinear dynamics of players' states and the environment are mutually influenced.

At each time step, a game in the sense of classical game theory is realized, to some degree, where players decide their actions within the game at that instant. In other words, each player's profit for all possible actions of all players can be determined at every point in time. Thus every player has a kind of payoff function at every move. However, this payoff function varies with time, and its change depends on the players' actions. Our DS game allows for dynamical changes of games themselves in this sense, where this change and players' actions are inseparable.

2.3. Components of the dynamical systems game

The basic components of the DS game world are the set of players, $N = \{1, 2, \ldots, n\}$, and the game environment, Ξ (Table 1). The set N is composed of n players, also called decision makers, and Ξ is composed of quantifiers that can change according to

Table 1
Components of dynamical systems game

Game environment	Ξ
Set of players	$N=\{1,2,\dots,n\}$

Table 2
Variables in the dynamical systems game

State of the game environment	x
States of players	$y=(y^1,y^2,\dots,y^n)$
Actions of players	$a=(a^1,a^2,\dots,a^n)$

the dynamics, but do not belong to the class of decision makers (e.g., salt, water, food, fuel).

The basic variables in the DS game world are the state of the game environment, x, and the state of the players, $y=(y^1,y^2,\dots,y^n)$ (Table 2).

In our DS game, x and $y^i(i \in N)$ denote multiple-component vectors. For instance, if we consider salt, water, fuel, and food in the game environment that we study, x is represented by $x = (x_{\rm salt}, x_{\rm water}, x_{\rm fuel}, x_{\rm food})$. The situation is similar for the state of player $i(i\in N)$, i.e., y^i. [1]

2.4. Dynamics of the system

We express the dynamics of the DS game by a map (though it is also straightforward to use differentiation equations). One of the advantages of representing the dynamics in a discrete time form is the ease of comparison with iterated or repeated games, such as the iterated prisoners' dilemma [3].

In the DS game, x and y vary with time according to the dynamics established in the system. Denoting the time by t, the game dynamics, g, is represented as follows:

$$g:(x(t),y(t))\mapsto(x(t+1),y(t+1)). \qquad (1)$$

As in the real world, the system we consider changes autonomously, even without players' actions. We call this property the natural law of the system. A decision made by a player may also affect the game environment, the other players, and himself. In other words, x and y can be changed both by the natural law and the effect of players' actions, a. Thus g consists of a natural law, u, and the effect of players' actions, v:

$$u:(x(t),y(t))\mapsto(x(t)',y(t)'), \qquad (2)$$

$$v:(x(t)',y(t)',a(t))\mapsto(x(t+1),y(t+1)), \qquad (3)$$

$$g=v\circ u. \qquad (4)$$

Here we have adopted the order of u and v, as in Eq. (4), to include successively the effects on the game dynamics due to players' actions and other causes. Use of the reverse ordering is just as valid

$$v:(x(t),y(t),a(t))\mapsto(x(t)',y(t)'), \qquad (5)$$

$$u:(x(t)',y(t)')\mapsto(x(t+1),y(t+1)), \qquad (6)$$

$$g=u\circ v. \qquad (7)$$

2.5. Decision making function, f

The player i refers to the state of the game environment x, that of the other players y^{-i}, [2] and that of himself in determining the action a_i to be taken, based on his own decision making mechanism f^i. We call this the decision making function. The function f^i gives player i's value judgment of all possible activities, while the function can be changed through evolution or a learning process. In other words, the function represents player i's personality. The operation of f^i is given as follows:

$$f^i:(x(t),y(t))\mapsto a^i(t) \quad (i\in N). \qquad (8)$$

We represent the set of all players' decision making functions as $f=\{f^1,f^2,\dots,f^n\}$ and the operation of f as follows: [3]

$$f:(x(t),y(t))\mapsto a(t). \qquad (9)$$

[1] For example, y^i could be $y^i=(y^i_{\text{nutritional state}}, y^i_{\text{monetary state}}, y^i_{\text{physical fatigue}},\dots)$. However, any $y^i(i\in$ a certain set of players $N)$ that appears in this paper is a mere one-dimensional variable. That is, we do not deal with, e.g., the relation between the dynamics of different variables of a player. And so, all y^i's that appears in this paper can be considered as y^i_1. It is for the future studies that y^i is implemented as a vector in the DS game framework.

[2] $y^{-i}\equiv(y^1,y^2,\dots,y^{i-1},y^{i+1},\dots,y^n)$.

[3] Precisely speaking, $(x(t),y(t))$ should be written as $(x(t)',y(t)')$, because we use Eq. (4), but we use $(x(t),y(t))$ in this definition of f for legibility and simplicity.

The DS game is characterized by g and f. Here, the game as a dynamical system is represented by embedding the players' decision making process f in the dynamics of the system g.

3. Discussion of the dynamical systems game model

Here we would like to discuss a framework for our dynamical systems game model described in the previous section. We first discuss the merits of the introduction of g and f for the dynamics of games. We also discuss bounded rationality and the iterated prisoners' dilemma for comparison. We next discuss the necessary properties that players should have in a dynamic game environment, referring to Rössler's autonomous optimizer model of an artificial brain.

3.1. Bounded rationality, the iterated prisoners' dilemma, and dynamical systems

The basic focus of this work is the situation in which decision makers interact with each other under various conditions in a dynamical environment, as in the world we live in. Here, it is important to understand the mechanisms of development or evolution of societies, emergent communications, and a player's behavior that can be observed in this situation. In order to understand these mechanisms, what kind of approach should we take?

From the standpoint of game theory, one might argue that analyses of game-like situations have already been completed. For example, it has already been shown that there are best strategies for two players playing chess, and that the results of the game can be determined before playing if both of the players have complete rationality. In addition, it has been shown that the rational behavior in the finitely iterated prisoners' dilemma (IPD) game is to always betray. The problem here, however, is that these theoretically optimal solutions often differ from the way people actually behave in such situations. The origin of this discrepancy seems to lie in the basic assumption of game theory, the players' rationality. The rationality of game theory usually requires too much ability. For example, the following assumptions are often made in game theoretical models.

- The players participating in a game are all rational decision makers.
- The rules of the game and the rationality of all players are common knowledge [1] among all players.

Under these assumptions, finding the theoretical solution for determination of the players' rational strategies becomes the primary purpose, and with these basic assumptions the construction of several theories concerning the solutions have been made possible. However, one problem confronted is the player's inability to determine what exactly rational behavior should be. For example, in a game such as chess (two-person zero-sum game), we can perceive the existence of the best strategy, but this does not necessarily mean that we can know concretely what is the best strategy. A similar situation occurs in other games as well. This characteristic has been discussed by Simon [21], who referred to it as bounded rationality. This complicating factor leads to some difficulty in applying game theory in real situations in which the computational power of players is bounded.

Problems involving this bounded rationality are not confined to a player's computational ability. Even if a player can easily compute the rational solution, it is not unusual for him to behave in a manner contradictory to that rationally following the result of this computation. For example, in the finitely IPD game, the Nash equilibrium strategy for both players is to always defect (betray). In this case, it is not required that the players have enormous computational power to determine the correct solution by backward induction. However, as shown by some experiments related to the IPD, such behavior is not always observed. As a famous example, we show here the experiment conducted by Melvin Dresher and Merrill Flood [5]. (This experiment was conducted in 1950 before Tucker coined the phrase prisoners' dilemma. [4])

[4] The story of the so-called the prisoners' dilemma is what Tucker presented to dramatize the game used in this experiment.

Table 3
Payoff matrix used by Dresher and Flood

		Player 2	
		D	C
Player 1	Cooperate	$-1, 2$	0.5,1
	Defect	0,0.5	$1, -1$

The payoff matrix used in this experiment is shown in Table 3. [5]

The prisoners' dilemma was repeated 100 times with two players in this experiment. In this game, the Nash equilibrium is $(\text{player}\,1, \text{player}\,2) = (\text{defect}, \text{defect})$ if the players recognize each step of this experiment as an isolated game, and it is also the 100 times repetition of $(\text{defect}, \text{defect})$ if they regard the entire experiment as one repeated game. In any case, the theoretical solution for both of the players is to always defect. Nevertheless, the actual experiment exhibits far more cooperative behavior than that predicted theoretically.

In addition to the experiments with the (iterated) prisoners' dilemma, there have been a number of experiments conducted using real people. Also in these experiments, people often behaved quite differently from the way predicted theoretically, even in simple games where players need little deduction to find the Nash equilibria (e.g., Chapter 5 of where the participants behave quite differently [4]). The above may be regarded as the natural consequences since we cannot have the perfect rationality presumed in game theory. Then, how should we develop the game theory to deal with the players' bounded rationality?

It can be said in recent years that research regarding bounded rationality has been started [2], and bounded rationality is still one of the hottest topics in game theory. In research regarding the IPD, there have been several approaches to the problem of bounded rationality. One of these uses computer simulations to explicitly determine the limits of players' computational power. One advantage of using computers to study the behavior of games is that in this case, a player is constructed as a computer program, so that we can fully understand the structures of the players and investigate the players' behaviors under various conditions by setting their abilities as we like. Another advantage of using computers is that they allow for the detailed investigation of a large system of games, and experiments regarding players' evolution or learning are also possible, for instance, by introducing mutation algorithms into the players' programs. For example, computer simulations have been carried out by representing the players by finite automata [13,20]. It has been shown in these studies that cooperation is realized in the IPD with a finite number of iterations, if the memories of the players are incomplete. In a sense, these results are similar to the cooperative behaviors we often see in reality in situations similar to the IPD.

In another approach to the bounded rationality problem, a finite IPD is regarded as an approximation of an infinite IPD and one seeks the equilibrium of the infinite IPD. This approach can be understood more concretely in the light of Nash's comments regarding the experiment by Dresher and Flood described above.

We can summarize two of the topics dealt with by Nash as follows. First, Nash emphasizes that when the prisoners' dilemma is iterated 100 times, we should regard it as one large multi-stage game, not merely as repetitions of an isolated game. Nash also points out that a game iterated 100 times is a finite game but it is too long for an ordinary person to determine the theoretical equilibrium by applying backward induction from the end of the repetition. The person may make a decision, feeling the 100 times iteration to be infinite one. In this case, we should apply the theoretical equilibrium of the m infinite IPD to the results of the 100-stage IPD experiment.

If Nash's assertions are correct, the realization of cooperation in a 100-stage IPD can be rationalized theoretically. The theoretical background of this rationalization of cooperation is the so-called Folk theorem, which is a basic theory for infinitely repeated games. The Folk theorem states that a super game G^{∞}, which

[5] The payoff matrix was made asymmetric. When the players select $(\text{player}\,1, \text{player}\,2) = (\text{defect}, \text{defect})$ or (cooperate, cooperate) in the prisoners' dilemma game with a symmetric payoff matrix, we cannot deny the possibility they would select the symmetric actions based on the feeling that the mutually gained same profit is the equilibrium profit and the rational solution for both the players.

is an infinite repetition of a game G, has points of Nash equilibria that satisfy the following conditions. [6]

1. Each such point can also be realized in the game G by the combination of the players' strategies. [7] In other words, each point exists within the so-called feasible set of the game G.
2. The average payoff for each player at each such point is larger than the smallest payoff ensured by each player's best response in G. In other words, each player should be individually rational.

The cooperative states realized in IPD using such strategies as "tit for tat" apparently satisfy the above conditions; it can therefore be concluded that these cooperative states are equilibria. If it is assumed that when we come across an IPD-like situation, (i) we recognize it as an infinite IPD game, and (ii) we determine our actions based on a calculation of the equilibrium of an infinitely iterated game, this combination of players' bounded rationality and the Folk theorem is quite reasonable.

At this point, let us turn our attention to dynamical games. Take, e.g., the lumberjacks' dilemma game, introduced in Section 2.1. In the lumberjacks' dilemma game, the behavior of felling a tree brings players some degree of profit, and therefore this action is always more profitable than the behavior of doing nothing or waiting. If the players are all rational, this game's equilibrium is characterized by the situation in which all players continue to cut trees forever, while there is little need for computational power to identify this equilibrium. Do we ourselves have, however, always behave in this way? Aren't there cases in which we wait for trees to grow? Aren't there cases in which we grow and cut trees alternately? This situation is similar to that in the above-described IPD experiments where players' behaviors are sometimes in contradiction to the theoretical conclusion that assumes a player's complete rationality. Then, if we assume that players inevitably have bounded rationality like we ourselves, what kind of approach is possible in the lumberjacks' dilemma game?

The lumberjacks' dilemma game can be regarded as a kind of multi-stage game if described in discrete time form, although one step of the repeated game is usually called a stage game. It is impossible, however, to apply a logical chain of the [players' bounded rationality] → [approximation for an infinitely iterated game] → [the Folk theorem] to the lumberjacks' dilemma, because the Folk theorem is applied only for games that involve an iteration of a single, unvarying stage game, while the DS game changes dynamically in time.

We have formulated the framework in Section 2, the dynamical systems game. In order to deal with the above-mentioned problems, the DS game model is quite simple, with two important features, the game dynamics and the players' decision making mechanisms embedded in the game dynamics. By applying this framework, we can consider the behaviors of players with bounded rationality and the resulting dynamics of the game itself through both experimental results and an analysis of the model.

3.2. Autonomous optimizer

In the DS game model, formulated in Section 2, the variables x, y, and a (see Table 2) change with time following g and f. Furthermore, f changes on a slower time scale of evolution or learning. The basic concept of this model is related to the autonomous optimizer, introduced by Rössler as a model for complex biological systems (in particular for brains), though the DS game may make its way from the viewpoint of game theory toward that of dynamical systems. Hence, the concepts of game theory, such as the player and the payoff (for each feasible action), are explicitly introduced into the DS game. As a result, we can know the short-range and long-range rationality in DS games. This suggests the possible superiority of the DS games to traditional games. Here we consider the DS game model in relation to Rössler's model, in particular focusing on the consideration of players' states and the decision making function.

[6] More strictly speaking, this Nash equilibrium is realized only when the players' payoff discount factor of G^{∞}, which discounts the payoffs of future stages, becomes sufficiently close to the upper limit 1.0. In this case, the normalized payoff is identical to the average payoff at all stages. Fundenberg and Tirole [6] have discussed the Folk theorem in detail.

[7] Each one's strategy can be the mixed (statistical) strategy, e.g., if there are two pure strategies A and B, the strategy of 20% strategy A plus 80% strategy B is possible.

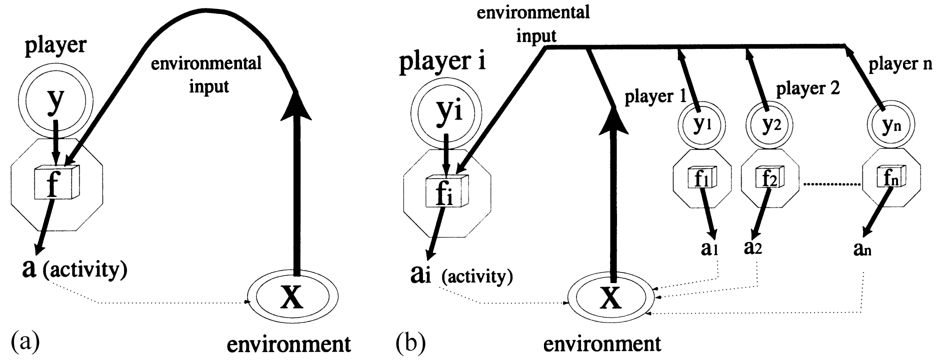

Fig. 1. (a) Conceptual figure of a player. A player has a decision making function, f, that uses the state of the game environment, x, and the players' states, y, as input information and determines the player's action, a, as an output. a then affects x. The function f changes in response to evolution or learning and maximizes (or minimizes) a kind of optimality functional. (b) Key map of the dynamical systems game. When the players are coupled via everyone's environmental input, the situation becomes a game. Each decision making function refers to states of the environment and the players (including himself) as inputs. Players' actions then affect the game environment. Thus the dynamics in the game-like interaction can be represented better by dynamical systems game than by game theoretical models that use algebraic payoff matrices. (More precisely, y, a, f in (a) could be rewritten as y_1, a_1, f_1, respectively, by seeing (a) from the viewpoint of one-person DS game model.)

DS games are described by the dynamics of the system g and the decision making function f, which also correspond to the concept of the autonomous optimizers' world. That is, the DS game consists of two classes influencing each other: the dynamical system of the game and the players (the key map of a player is shown in Fig. 1(a)).

Autonomous optimizers are a class of systems that autonomously try to maximize (or minimize) a certain functional that is generated by the system itself [17]. Optimizers are a subclass of dynamical systems, and it is therefore possible to describe the whole world, including the optimizers themselves, as dynamical systems. This, however, is not a useful approach when we are interested in the function or the behavior of an optimizer itself because knowing the dynamics does not necessarily result in an understanding of the optimizer, and extracting the behavior of the optimizers from the dynamics is actually impractical [18].

When multiple optimizers are coupled, another problem must be considered [19]. In this case, information regarding a particular optimizer's behavior acts as environmental input for other optimizers. This is the situation we refer to as a game, and here exists the feature common to the model of autonomous optimizers and the DS game model, where players correspond to optimizers. Optimizers are affected by the environment, which is treated as a variable, and, carrying out this action, by the states of other optimizers, while an optimizer's state acts as a signal for others. By using all signals as inputs, the optimizers determine their next action and also affect the environment. Among the close relationships between autonomous optimizers and the DS game, we focus our discussion on the role of the state variable, which is also used in the DS game.

In most experiments using game theoretical models that employ iterated games, a player usually refers to the history of all the players' past actions in order to determine his next action. In some other cases, a player refers to the payoffs that players have received in previous games. By contrast, in the DS game, a player refers to the states of the game environment x and the states of the players (including one's own state) y to determine his behavior. The decision making function f^i of player i projects x and y onto the action of player i, a^i. (A key map of the DS game is shown in Fig. 1(b).)

Although the information referred to in decision making for this model is not based on the past actions, but the current states, the past history information can be embedded into x and y if the players evolve (learn) and adapt to the varying environment, because the natural law (Section 2.4) of the game is completely given from the outset in this model. Specifically, we can theoretically obtain past history information to some extent by knowing x and y within the limit of precision of the parameters in f that use x and y as inputs. (Note that the dynamics and parameters in our DS game model take real number values.) The form of f is refined as additional experience is obtained. Thus past experiences can be embedded in f.

As a very simple example, consider the following set of circumstances:

1. Apples that were on a plate have disappeared during a short period while I was not watching (x).
2. I (player i) am now very hungry (y^i).
3. Furthermore, the facial expressions of the persons sitting next to me indicate that they are very satisfied (y^{-i}).

Then I can guess that the other people must have eaten the apples. If the room is a closed system, the inference becomes even more valid. I may take revenge on these people if my decision making function f^i has been sufficiently trained to analyze the situation. Actually, the experiments carried out in this study show that the more sophisticated the way of f's reference to x and y are, the more advanced behaviors can be observed.

The meaning of the word "state" tends to be somewhat vague when we say a player's state. Especially when we refer to another person's state, its meaning inevitably involves a value judgment. When we look at another person's facial expression, our assessment of how the person looks, such as "he seems to be lost in thought", or "he seems to be angry", may vary according to our past experiences and our current state of mind. This situation can be expressed in more detail with the terminology of the DS game as follows:

- y^i is a state observable by other players, and it acts as a signal for them.[8] (It corresponds to the state in an autonomous optimizer.)
- f^i is the inner structure of player i invisible to others that is implemented by value-judging mechanisms. It determines how he recognizes the state of outside objects (the others and the environment) and of himself.

The player k with state y^k refers to player i's state y^i, and makes his decision. In fact, the statements "he seems to be lost in thought" and "he seems to be angry" are both described in $f^k(y^k, y^i)$. As a result, player k makes a decision such as "let us leave player i alone, who is lost in thought" or "let us appease player i, who is angry". Here we cannot consider apart from the decision making mechanism how player i's state is observed by player k. That is, from player k's viewpoint, y^i is the state based on the pure external expression of player i, while when f^k is applied to (y^i, y^k), it is the state of player i as I imagine it to be. (Rössler [19] has pointed out the possibility that the study of coupled autonomous optimizers will bring about an understanding of the development of human language, especially the usage and comprehension of the personal pronoun 'I'.)

Now we briefly consider the relevance of this discussion to the model for the lumberjacks' dilemma. Theoretically, the best strategy may always be to fell the biggest tree at every point in time in this game, although this strategy only considers the size of the trees but not the states of the players. From the viewpoint of the DS game, however, players' states sometimes play an important role. The evaluation of the action to fell a tree depends on whether I am not in a satisfied state and whether the next player seems to be satisfied. Actually, the delay effect mentioned above is partly based on the evaluation of states.

4. Lumberjacks' dilemma game

As an application of the DS game framework, we present in this paper the lumberjacks' dilemma (LD) game, whose outline is given in Section 2.1. In this section, the model of the LD game and the concrete procedure of its experiment are explained.

1. *Game world, ecology of the lumberjacks' dilemma game*. In the game world of the LD game, there

[8] Note here that what belongs to player i in the DS game is y^i and f^i.

are several wooded hills in which the lumberjacks live. Let us suppose that the lumberjacks in the population can be classified into s species and the number of hills is h. The lumberjacks who belong to a particular species follow the same strategy, and adopt the same decision making method.

Each lumberjack selects a favorite hill to live on. Competing or cooperating with other lumberjacks who have selected the same hill, he fells trees that grow in time. Thus, a lumberjack from the population becomes a player on a hill and plays the LD game there. Each game on a hill is completely isolated from the games on the other hills, i.e., the lumberjacks on a given hill have no idea how the games are played on other hills.

2. *The hill, the place where the LD game is played*. Let us denote the number of trees on each hill by m, and that of the lumberjacks by n. Now, n lumberjacks compete over m trees and cut them into lumber. Several lumberjacks of the same species can live on the same hill. These n players play a game repeatedly. At the conclusion of the game (when all players have reached $t = T$), each player's average score over T steps is measured. This gives the fitness of each player.
3. *One generation of the game*. On each of the h hills, an LD game (a T-round repeated game) is played once, with similar games played on all the hills. The playing of the LD games on all the hills is called one generation of the game. The fitness of a species is given by the average of the fitness of all the players of that species over all the hills. Each species leaves offspring to the next generation, in a manner that depends on the fitness. A species with low fitness is replaced by mutants of a species with high fitness. Though the same procedure is repeated in the next generation, the lumberjacks in the next generation know nothing about the plays of the previous generation.

4.1. Game world: the lumberjacks' dilemma ecosystem

In this section, we explain the lumberjacks' dilemma ecosystem. The LD ecosystem is the game world wherein the lumberjacks live and leave their progeny. There are several hills in this world, and on each hill, some lumberjacks play the lumberjacks' dilemma game.

Here, we describe the components of the game world and their corresponding variables. The component elements of the lumberjacks' dilemma ecosystem include the following:

- a set of lumberjack species: $S = \{1, 2, \dots, s\}$;
- a set of the hills: $H = \{1, 2, \dots, h\}$;
- a generation variable: generation $= (1, 2, \dots)$;
- a variable representing the number of species made extinct by the selection process, k.

A lumberjack species is defined as the set of all lumberjacks in the population having the same strategy. Throughout all experiments in this paper, the parameters s and h are set as $s = 10$ and $h = 60$.

4.1.1. Lumberjack species

The attributes of lumberjack species i include the following:

- a decision making function, f^i;
- a fitness variable, fitness_i.

4.1.2. The hill

The hill is the stage of the lumberjacks' dilemma game and is composed of the following:

- the set of players: $N = \{1, 2, \dots, n\}$;
- the set of the resources: $\Xi = \{1, 2, \dots, m\}$;
- a time variable whose value indicates the round number: $t\,(t = 1, 2, \dots, T)$.

Within the general DS game framework, a lumberjack on a hill is considered a player and a tree on a hill is considered a resource. We use these terms in the following, although we use lumberjack and tree when emphasizing points specific to the lumberjacks' dilemma. Throughout the experiments in this paper, the parameter T is commonly set to 400, which is also each lumberjack's lifetime.

4.1.3. Bring a lumberjack to the hill from the population

Each lumberjack in the population randomly selects a hill, goes there, and plays the LD game for his entire

life. The procedure followed in the actual experiment is as follows:
1. Select a hill in the LD ecosystem (the hill i).
2. Select a lumberjack species in the population randomly and bring a lumberjack of that species to the hill i. This lumberjack will be called a player in that hill.
3. Repeat procedure (2) until the number of players reaches the limit n. Lumberjacks of the same species can exist in the same hill.

The above procedures are applied to all the hills (from hill 1 to h). As a result, each hill comes to have n players, and the LD game is played on each hill.

The species of player i on the hill $\eta(\in H)$ is denoted by $S(\eta, i)$, or we abbreviate it as $S(i)$ when the hill of the player is not important. Among the total nh lumberjacks on all the hills, the number of lumberjacks from species $\sigma(\in S)$ is denoted by number(σ), while the average score of player i on hill η in the LD game is denoted by average(η, i), when $\eta \in H$ and $i \in N$.

4.1.4. Fitness and the change of generations

The LD games on all the h hills as a whole are considered one generation of the game. After a generation is over, several species are selected with regard to fitness and replication, accompanied by a mutation process, before the next generation begins. The fitness of the species σ can be calculated as the average score of all the players of all the hills who belong to the species σ as follows:

$$\text{fitness}_\sigma = \frac{\sum_{\eta\in H}\sum_{i\in N,\, S(\eta,i)=\sigma}\text{average}(\eta, i)}{\text{number}_\sigma}.$$

Before the next generation begins, the k species with the lowest fitness are removed from the population in a selection process. The surviving $(s-k)$ species can leave their descendants to the next generation, and these participants will have the same decision making functions as their respective ancestors. The extinct species are replaced by k new species, which are mutants of k species randomly selected from among the surviving $(s-k)$ species. The process of mutation is explained in detail later. Throughout all experiments reported in this paper, the parameters s and k were set to $s = 10$ and $k = 3$. The above procedure is then repeated in the next generation.

4.2. The lumberjacks' dilemma game on a hill

Here we give a detailed explanation of the game played by the n lumberjacks (players) on each hill.

4.2.1. Players and resources

On each hill, there live n lumberjacks (players) competing for m trees (the resource of the hill). Let us denote the state of the resource of a hill at time t by $x(t)$. $x(t) = (x_1(t), x_2(t), \ldots, x_m(t))$ is an m-dimensional vector whose components represent the sizes of the m trees. Each player possesses a one-dimensional variable that represents his state and has a decision making function (strategy). For example, the state of player i is denoted by $y^i(t)$, and the decision making function by f^i, denoted by $y(t) : (y^1(t), y^2(t), \ldots, y^n(t))$ and $f = (f^{S(1)}, f^{S(2)}, \ldots, f^{S(n)})$. Each component of $x(t)$ and $y(t)$ is represented by a positive real number. In the LD game, a player's state has two important features. First, it increases according to the size of the lumber obtained from a tree. Second, it decreases if the player takes no action and does not obtain any lumber. That is, a player's state may be considered to represent the degree of nourishment, wealth, etc. These states govern the lumberjacks' fitness.

Players decide their actions by referring to the sizes of the trees, $x(t)$, and the states of players, $y(t)$.[9] The totality of all players' actions is denoted by $a(t) = (a^1(t), a^2(t), \ldots, a^n(t))$. Each player's individual action can be one of $m+1$ possible actions: do nothing, cutting tree 1, cutting tree 2, ..., or cutting tree m. These actions are represented by $0, 1, 2, \ldots, m$, and the set of all these feasible actions is denoted by A.

Thus, the properties of players and resources are as follows:

- the state of the resource (the size of the trees): $x = (x_1, x_2, \ldots, x_m) \in R_+^m$;
- the state of the players: $y = (y^1, y^2, \ldots, y^n) \in R_+^n$;

[9] To be precise, their decisions are not based on $x(t)$ and $y(t)$, but on $x(t)'$ and $y(t)'$. This point will be touched upon later.

- the actions of the players: $a = (a^1, a^2, \dots, a^n) \in A^n$;
- the decision making functions of the players: $f = (f^{S(1)}, f^{S(2)}, \dots, f^{S(n)})$.

4.2.2. One round of an LD game

The same procedure is repeated T times in the LD game, and this yields the game dynamics in a hill. The unit of this repetition is called a round. The factors governing the behavior of the system in a round are the following:

1. *Natural law*. The states of the players and of the resource (the sizes of trees) change according to a natural law.
2. *Decision making of the players*. Each player selects his action by considering the states of the players (including his own) and of the resources.
3. *Effects of the actions*. Players' actions affect the state of the resource on the hill. Namely, the sizes of trees cut by the players will be reduced accordingly. The lumber cut from a tree will be divided equally among all the players who cut the tree together, obtaining lumber increases the value of state of a player.

(1) *Natural law*. The natural law (Section 2.4) of a DS game affects game dynamics, but it has nothing to do with the decision making of players. In LD games, natural laws have two functions: to decrease the value of the players' states, $y^i(t)' = u_N(y^i(t))(i \in N)$, and to increase the sizes of the trees, $x_k(t)' = u_\Xi(x_k(t))(k \in \Xi)$.

The decrease of the values of the players' states, which is defined by the map u_N is represented as $y^i(t)' = u_N(y^i(t))$. The value of player i's state $y^i(t+1)$ at the next round remains $y^i(t)$ unless he acquires lumber. Here we choose a damping form of this map given by $u_N(z) = \kappa z (\kappa < 1)$. Throughout the experiments, the constant coefficient κ was set to 0.8.

The growth rule for the trees is given by the map u_Ξ, according to which the size of the tree i, x_i will be changed to x_i':

$$x_i(t)' = u_\Xi(x_i(t)).$$

If a tree is not cut, its size at the next round $x_i(t+1)$ is given by $x_i(t)'$. For u_Ξ, we use two types of maps. In the experiments described in this paper, we mainly used the following three-dimensional polynomial function:

$$u_\Xi(x) = 0.7x^3 - 2.4x^2 + 2.7x.$$

The corresponding graph for $y = u_\Xi(z)$ is shown in Fig. 2(a). We call this map Ξ a convex map because of its shape. The growth process of a tree from an initial condition $x(0) = 0.10$ is shown in Fig. 2(b) for the case in which it is not felled by any players. As shown by this figure, the tree grows rapidly until round 3, but is almost saturated at round 4. Thus, it is seen that waiting too long is not necessarily a good strategy for the lumberjacks.

Another natural law for the growth of trees we use in this paper is the following function:

$$u_{\Xi'}(z) = \min(1.5z, 1.0),$$

where z is the size of the tree. In this case, the tree grows at a rate of 50%, but its size never exceeds 1.0.

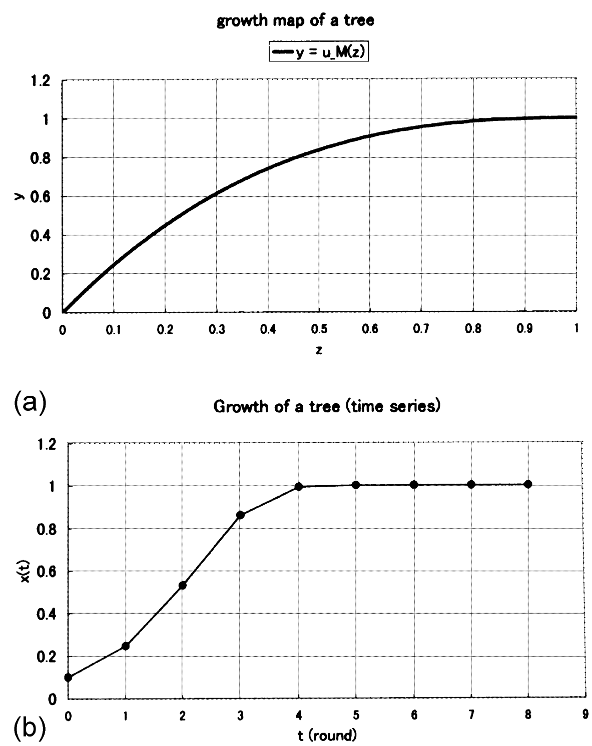

Fig. 2. Growth of a tree described by a convex map: (a) the growth law of the trees; (b) the time series for the size of a particular tree.

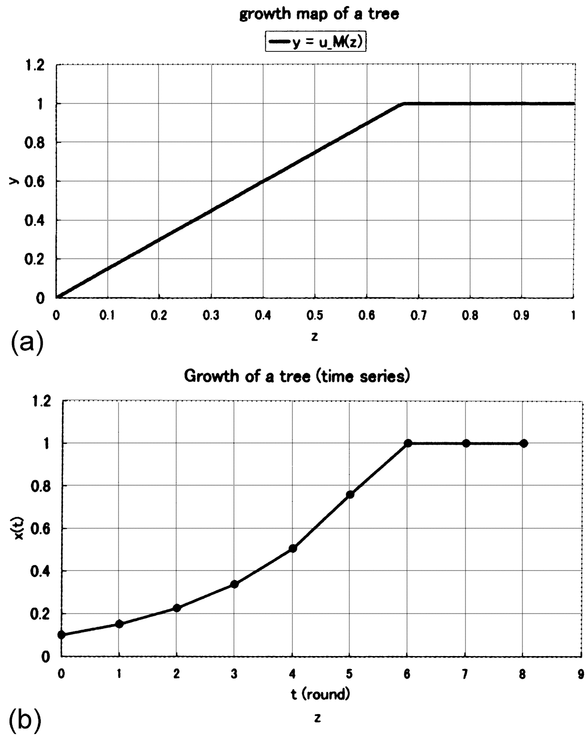

Fig. 3. Growth of a tree described by a linear map.

The growth of a tree following $u_{\Xi'}$ is shown in Fig. 3. For convenience, we call this map $u_{\Xi'}$ a piecewise linear or, simply, linear map.

(2) *Decision making of the players*. All players have their own mechanisms to decide their actions: to decide which tree to cut, or to refrain from any action. Each player chooses their next action $a(t)$ in a manner that depends on the state of their surroundings, x and y. A detailed explanation of the decision making function f is given later (Section 4.3).

(3) *Effect of the players' actions*. The players' actions $a(t)$ may reduce the sizes of trees $x(t)'$, and the values of the states of the players who cut trees increase accordingly. In the actual experiments, the size of tree i is set to become, in the next round, γ^{ν_i} times as big as it was, where ν_i is the number of players who cut it ($\gamma < 1$).

$$x_i(t+1) = \gamma^{\nu_i} x_i(t)'.$$

Note that $\sum_{i\in\Xi} \nu_i \leq n$, because each player can cut at most one tree in a given round. Throughout the experiments, the parameter γ was set to $\frac{1}{3}$. We note that the results of the simulations with $\gamma = \frac{1}{2}$ are not significantly different.

Acquisition of a resource changes a player's state. In this LD game, a player i's state, $y^i(t)$ is taken to be a linear function of the size of the lumber that he acquires. Suppose that $a^i(t) = \mu$, player i can be said to have felled a tree μ unless $a^i(t) = 0$. Here we denote the size of the lumber by Δ^i. Δ^i is, of course, zero if player i has selected the action 0, i.e., do nothing. Otherwise, it can be determined by the following:

$$\Delta^i = \frac{x_\mu(t)' - x_\mu(t+1)}{\nu_\mu} = \frac{(1-\gamma^{\nu_\mu})}{\nu_\mu} u_\Xi(x_\mu(t)).$$

Hence Δ^i is the increase in value of the state of player i:

$$y^i(t+1) = y^i(t)' + \Delta^i.$$

4.2.3. *Players' utility in a round*

As stated in Section 4.2.1, a player's state may be considered as his nutritional state, monetary state, etc. We assume that the utility of player $i (\forall i \in N)$ is simply the current state, y_i. Regarding one round of the LD game, player i's utility for the round is maximized by choosing Δ^i as large as possible. The utility of a player is an increasing function of the size of the tree he cuts and a decreasing function the number of people who cut the same tree.

4.2.4. *Iterated game*

In the LD game, the above-described procedures constituting one round are repeated[10] until the number of rounds reaches the maximum, T. After the final round, each player's average score for the T rounds is calculated, and the individual player's scores increase or decrease the fitness of the species that he belongs to (Section 4.1.4).

[10] An LD game can be considered as a kind of iterated game, though the payoff for a given action usually changes with time. In this way the LD game differs from ordinary existing iterated games, such as the IPD (Section 3.1).

The average score (utility) of player $i (\in N)$ in the hill $\eta(\in H)$ is given by

$$\text{average}(\eta, i) = \frac{\sum_{t=1}^{T} y^i(t)}{T}.$$

All of the above procedures for the T rounds constitute the LD game for a given hill.

4.3. Decision making function

Each player decides his own action, a, based on the states of his surroundings, which are denoted[11] by x and y. The function for player $i (\in N)$ that represents the decision making mechanism is defined as the decision making function of player i, $f^{S(i)}$, where $S(i)$ denotes the species that player i belongs to. $f^{S(i)}$ is the inner structure of player i and is invisible to other players. The player i's decision making function acts as follows:

$$a^i(t) = f^{S(i)}(x(t), y(t)).$$

Here, f can depend on the state of all the players or only on the state of the nearest neighbors. There are many possibilities for the choice of f, but any structure for f is valid as long as the map uses only x and y as inputs. In the LD game, we have chosen f to be very simple as shown below.

4.3.1. Reference to the players' state

In Section 4.2.1, the set of players and the states of players were defined as follows:

$$\text{set of the players}: \quad N = \{1, 2, \dots, n\},$$
$$\text{state of the players}: \quad y = (y^1, y^2, \dots, y^n).$$

However, this representation is given from our viewpoint as observers of the experiments. We have assigned indices to the players, player 1, player 2, ... The order of this assignment of numbers is meaningless from each player's point of view.

Here, we stipulate that the players on a hill from the viewpoint of a particular player (player i) are described as follows:

$$\text{set of players}: \quad \tilde{N} = \{1, 2, \dots, n\},$$
$$\text{state of the players}: \quad \tilde{y} = (\tilde{y}^1, \tilde{y}^2, \dots, \tilde{y}^n).$$

The correspondence between (N, y) and $(\tilde{N}, \tilde{y})$ depends on how player i recognizes the outer world. First, player i identifies himself as player n within own decision making function; thus, y^i corresponds to the last component of $\tilde{y}$, $\tilde{y}^n$.

Regarding the arrangement of other components, there are various possibilities. For example, let us consider the case that each player distinguishes the others by their location, as was the case in the experiments considered in this paper. When n players are situated in a circle, we can make a one-to-one correspondence between N and $\tilde{N}$, and also between the components of y and $\tilde{y}$ of player i. Player i regards, for instance, the next player on his left as player 1, the second to next as the player 2, ..., and he regards the next player on his right as the player $n-1$, etc. Insofar as player i can distinguish the other players by numbering them by some means, this numbering method is all right for his decision making whether others may be situated in a line, zigzagged, or arranged randomly. As a second method of identifying players, any given player (who identifies himself as player n) numbers the other players from 1 to $n-1$ in order of decreasing the value of their state.

In the present model, a player's own state, $\tilde{y}^n$, usually has a special role among input variables for the player's own decision making function. Throughout the development or evolution of the decision making function, however, we do not impose a special role for $\tilde{y}^n$ in advance. A player's own state and those of other players are not explicitly given different roles as input variables for the decision making function. At the outset, the only distinction between $\tilde{y}^n$ and the other player state variables is its location. It is through the evolution of the decision making function that the own state $\tilde{y}^n$ begin to have a special role in the decision making function.

4.3.2. Motivation system

To implement concretely the decision making function, $f : (x, \tilde{y}) \mapsto a$, we introduce the motivation map mtv_r, for each feasible action r ($r \in A$). We

[11] Here, x and y are used for simplicity, though, strictly speaking, x and y here should be written as x' and y'.

can know a player's incentive to take the action r by mtv_r, given by a real number. In the LD game, mtv_0 denotes the incentive to do nothing, and $\mathrm{mtv}_r (r \in A,\ r \neq 0)$ denotes the incentive to fell the tree r. Here, the set of motivation maps, mtv, can be defined as follows:

$$\mathrm{mtv}=\{\mathrm{mtv}_r | r \in A\}=\{\mathrm{mtv}_0, \mathrm{mtv}_1, \mathrm{mtv}_2, \ldots, \mathrm{mtv}_m\}.$$

The structure of a player's mtv varies throughout his evolution. The functioning of mtv_r can be described as follows:

$$\mathrm{mtv}_r : R_+^m \times R_+^n \ni (x, \tilde{y}) \mapsto \mathrm{mtv}_r(x, \tilde{y}) \in R.$$

Each player selects the action whose motivation has the largest value among the set $\{\mathrm{mtv}_r\}$. In this sense, mtv_r can be said to be the utility function for the action r. By using $\{\mathrm{mtv}_r\}$, the concrete representation for the decision making function $f^{S(i)}$ of player $(i \in N)$ can be defined as [12]

$$f^{S(i)}(x, \tilde{y}) = a^i \text{ if } \mathrm{mtv}_{a^i}(x, \tilde{y}) \geq \max_{r \in A} \mathrm{mtv}_r(x, \tilde{y}).$$

Let us call the above-described mechanism using the motivation map the motivation system. In determining the action to be taken, the relevant information regarding the set $\{\mathrm{mtv}_r\}$ is not the absolute value of each mtv_r, but the identity of the element with the largest value.

Many kinds of implementations for the maps $\{\mathrm{mtv}_r | r \in A\}$ may be appropriate as long as each is a map from $(x, \tilde{y})$ onto the real numbers. In this work, each map is defined as a one-dimensional function of the state valuables x and $\tilde{y}$ as follows:

$$\mathrm{mtv}_r : (x, \tilde{y}) \mapsto \sum_{k \in M} \eta_{kr} x_k + \sum_{l \in \tilde{N}} \theta_{lr} \tilde{y}^l + \xi_r. \qquad (10)$$

Here, $\{\eta_{kr}\}$ and $\{\theta_{lr}\}$ are real matrices and $\{\xi_r\}$ is a real vector. These coefficients determine the player's strategy, and tiny differences between the coefficient values assigned to strategies can sometimes decide which player is superior.

The formulation using this linear programming may be the simplest way to define the structure of mtv for players to cope with the dynamical game environment. Of course, more sophisticated decision algorithms using maps from $R_+^m \times R_+^n$ onto R can also be used.

[12] There is the possibility that there are two incentives $\mathrm{mtv}_i(x, \tilde{y})$ and $\mathrm{mtv}_j(x, \tilde{y})$ that have identical values $(= \max_{r \in A} \mathrm{mtv}_r(x, \tilde{y}))$. However, the possibility of this occurring approaches zero as the generations proceed and the strategies become more complicated, since mtv is determined by several real numbers. For such events with identical incentives, the action with the smallest index is selected in the numerical experiment here.

4.3.3. Mutation of the decision making function

In the LD game ecosystem, new mutant species with higher fitness can replace species with lower fitness. The mutation process is implemented by slightly changing the parameters of the decision making function of the parent species, i.e., the matrices η and θ, and the vector ξ. Every component of η, θ, and ξ of the new species is chosen as a random number from normal distribution with variance σ around a mean value which is equal to the value characterizing the parent species. (Throughout the experiments, σ was set to 0.10.)

4.4. Initial conditions

We have conducted several experiments with various conditions by changing the number of trees, the number of players, the type of decision making functions, etc. Here, the other initial settings that have not yet been described are chosen as follows.

The setting for the first round of the game on each hill is as follows:

- value of each $x_i (i \in \Xi)$: 0.10;
- value of each $\tilde{y}_j (j \in N)$: chosen from random numbers from a normal distribution with mean 0.10 and variance 0.10.

The coefficient parameters η, θ and ξ for the decision making functions of the initial 10 lumberjack species in the LD ecosystem are generated as random numbers from a normal distribution with mean 0.0 and variance 0.1. A coefficient around 0.0 implies that the lumberjack in the first generation gives little consideration to his surroundings and his own state. Through evolution the players begin to consider the outside world and their own states.

4.5. Discussion of the payoff structure in LD games from the viewpoint of static games

Let us now touch upon what should be noted about the payoff structure of the LD game modeled in this section. From the viewpoint of static game, LD games with a convex-map-type natural law for the growth of trees and those with a linear-map-type natural law are essentially the same. Consider, e.g., one-person, one-tree LD games. In the static game that corresponds to any single round of an LD game, the score for the action of cutting a tree in the payoff matrix is always larger than that for the action of waiting, irrespective of the choice of convex-map-type or linear-map-type. Furthermore, the action of waiting sometimes brings the player a long-range profit in either type of LD game. In case of multiple-person games, LD games of both types involve a social dilemma and such a game could take the form of an n-person prisoners' dilemma if represented as a static game. In fact, the players can maximize their collective profit by the mutual cooperation in waiting for the sufficient growth of the trees, but each of them can maximize their personal profit by betraying the others and cutting the trees earliest. *Thus, several types of DS games that differ the level of the concrete description of the dynamics can be classified into the same static game.*

5. Evolutionary simulation of one-person LD games

5.1. Introduction

5.1.1. Characteristics of one-person games

One may call this LD game a one-person game, although, strictly speaking, by definition, a one-person game cannot be called a game. [13]

One-person games are simply maximization (or minimization) problems, i.e., the rational solution for a player in a one-person game can always be decided only by optimizing some functional, while in multiple-person games it is usually not possible for a player to decide his best behavior without considering others' decision making processes. Hence, the difficulty in solving for the optimal strategy in one-person games is just a technical rather than a conceptual one, in spite of its significance [12]. From the game theoretical point of view, what matters most in ordinary optimization problems, such as the traveling salesman problem, is the possibility of constructing the payoff matrix itself. Once the payoff matrix of a one-person game is constructed, solving for the optimal behavior is merely a matter of determining the strategy whose payoff is the largest.

Of course, the dilemma that some multiple-person games involve cannot appear in a one-person game including one-person, lumberjacks' dilemma games, although the name involves the term dilemma. The word dilemma used with regard to multiple-person games usually corresponds to the situation in which every player's rational decision making for maximizing his personal utility, paradoxically, results in lowering the utilities of all the players.

That is, if the players cooperated, they could all acquire higher utilities (Pareto dominant), but their rationality does not allow this acquisition. On the other hand, in one-person games, a player's rational decision making always maximizes his utility.

Now let us consider the case of a one-person LD game. In such a game, the best strategy for a player is, simply, to always wait for sufficient growth of a tree and then cut it, because there is no competitor who tries to rob the player of the profit in this game. Thus, the game is simply a maximization problem. Here, the player can completely control the sizes of the trees and thus can increase the average score per round.

5.1.2. Main interests

There are two main points of interest in one-person LD games. First, we are interested in how a player maximizes his utility in a game with dynamics. Note that there exists a dilemma-like situation even in the one-person LD game. By cutting a tree the player always realizes a higher score at the next step. However, the strategy of cutting a tree at each step is usually not good for long-term well-being. Here we

[13] Game is nothing, but a mathematical formulation representing problems among more than one decision makers.

Table 4
Game G: an example payoff matrix of a one-person static game

Strategy	Payoff
A	5
B	8
C	4
D	1

Table 5
Example payoff matrix of a one-person dynamical systems game

$$\begin{bmatrix}5\\8\\4\\1\end{bmatrix} \to \begin{bmatrix}5\\6\\4\\2\end{bmatrix} \to \begin{bmatrix}4\\3\\5\\3\end{bmatrix} \to \cdots$$

need a strategy that takes into consideration the dynamics of the game. For comparison, let us consider a static game, the game G, represented by the payoff matrix in Table 4. Here, as a matter of course, the player should select the strategy B, by which he can get a score of 8. Thus, it is easy to determine the solution of a one-person game, once the payoff matrix is given as a rule. In case of the game consisting of the iteration of game G, the best strategy is merely the iteration of B (all-B). In a one-person dynamical systems game, however, a player's behavior in each round often changes in accordance with the dynamics of the game, which can be represented by a dynamical payoff matrix (Table 5).

The second point of interest here is in the evolution seen in one-person LD games that have (other than the number of players) the same conditions as multiple-person LD games (that will be investigated in the following papers). We are especially interested in how the decision making function evolves in one-person LD games with consideration made only of the game environment (Section 2.2) and the player's own state.

We have conducted three experiments for each kind of LD games with the same conditions, but with different seeds. (Random numbers are used to generate the initial species, creating mutants of the species from the previous game, and deciding on which hill a player of a particular species will live.) For each kind of game, a typical example of the three experiments is presented here.

In the following section, an evolutionary simulation of a one-person convex-map-type LD game is investigated. The result of a one-person, two-tree convex-map-type LD game is given in Appendix A. The evolutionary phenomena of the linear-map-type LD game are discussed below (in the latter part of Section 6).

5.2. One tree in a hill

5.2.1. Rough sketch of the evolution (fitness chart)

First, we consider the evolutionary simulation of a one-person, one-tree convex-map-type LD game. In Figs. 4(a)–(c) (each of which is called a fitness chart), the fitness that the fittest species of each generation (the generation's fittest species) possesses, which we call the fitness value of the generation, is plotted with generation. In each fitness chart, the horizontal axis corresponds to the generation, while the vertical axis corresponds to the fittest value of each generation.

In Fig. 4, in each case, the fittest value does not decrease and increases stepwise with generation, though the value is almost saturated at about 1.5 in quite early generations. This non-decreasing form is rather natural, since this game is a one-person game. The fitness of any species in a one-person game is determined solely by its strategy (since there is no interaction with other species). Hence a new species can enter the population only by rising its own fitness, not by lowering the fitness of other strategies.

As shown in Figs. 4(b) and (c), the fitness value can increase at later generations, but the increase becomes much smaller. Furthermore, the frequency at which increases occur also becomes smaller in later generations. For example, beyond the plots in Fig. 4, new fittest species appear at the 314th, 3604th, 5847th, and 8983rd generations.

5.2.2. Game play at early generations (action chart, resource chart, state chart, and decision making function)

In Figs. 5(a)–(c), the dynamics of a player of species ID-00001 is plotted. In this experiment, this species

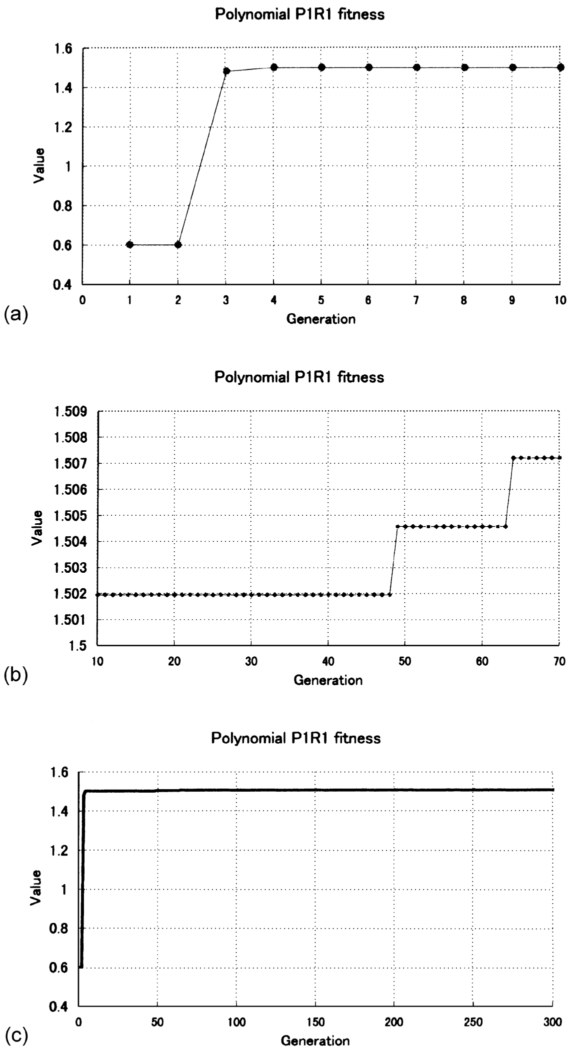

Fig. 4. Fitness chart in an LD game with a lumberjack and a tree: the fitness chart is shown: (a) up to the 10th generation; (b) from the 10th to the 70th; (c) up to the 300th. In all the figures, the horizontal axis represents the generation, and the vertical axis represents the fitness value of the generation, which is the fitness value of the fittest species of each generation.

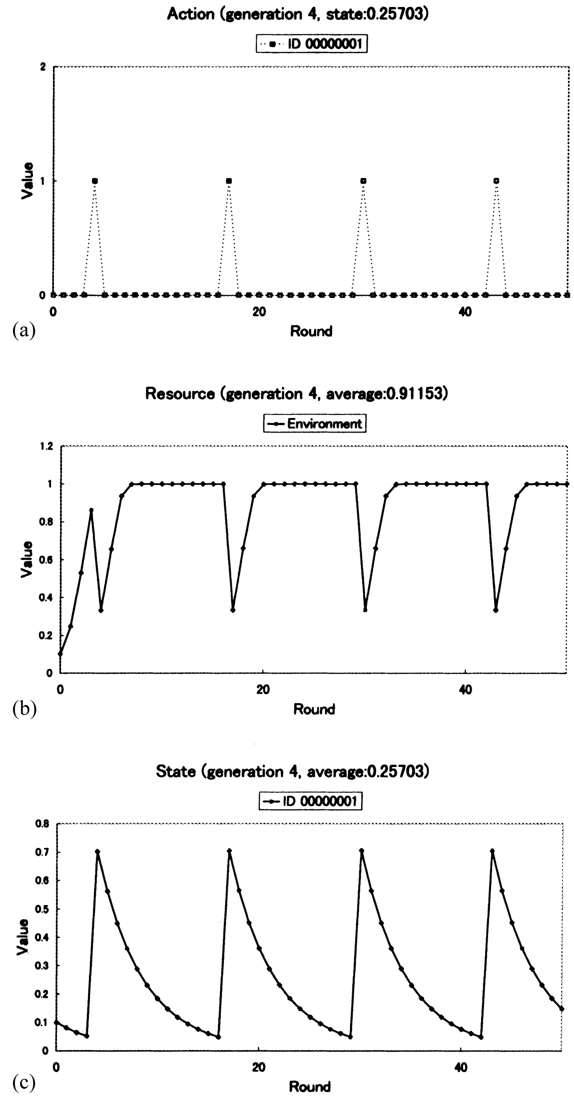

Fig. 5. (a) Action chart — the dynamics of the player's action. (b) Resource chart — the dynamics of the size of the tree. (c) State chart — the dynamics of the player's state. In the action chart, the average of the player's state (the average score) for T rounds appears in the title, and the name of the lumberjack species to which the player belongs is shown in the legend (in the square box). In the resource chart, the average size of the tree for T rounds is attached to the title. For example, (a) and (b) show that the player of the species ID-00000001 has an average score of about 0.25 and the average size of the tree is about 0.91.

existed up to the fourth generation. Figs. 5(a)–(c) display the dynamics of the player's action, the size of the tree, and the player's state, respectively. In all these figures, these quantities are plotted versus the round as a horizontal axis. Here, only the behavior up to the 50th round (among the $T = 400$ rounds) is plotted, since this behavior repeats periodically beyond this point. In Fig. 5(a), the action of the player in each round is shown, where for $n \geq 1$ action n consists of cutting tree n, while action 0 is simply to do nothing. The vertical axes of Figs. 5(b) and (c) represent the size of the tree and the state of the player, respectively. We call figures like Figs. 5(a)–(c) as the action chart, the resource chart, and the state chart, respectively.

Fig. 5(a) shows that this player basically repeats the pattern of cutting the tree and doing nothing for 11 rounds, then cutting the tree again, and repeating this indefinitely. Fig. 5(b) plots the increase of the tree size while the player is waiting, but the growth slows considerably after about three rounds. In this sense, this player waits too long for the growth of the tree. In fact, the fitness of the player is not good, and it becomes extinct within this generation.

The action of a player of the third fittest species (given by ID-0000000E) in the fourth generation is shown in Figs. 6(a)–(c). This player cuts the tree before its growth rate slows considerably and gains more profit than the player of Fig. 5. Thus, species whose strategy effectively manages the game dynamics to acquire the long-term profit (lumber) can survive to the next generation. (The game dynamics managed by the fittest species of the fourth generation has a similar characteristic to that of later generations, as shown later (period-2 dynamics presented in Section 5.2.4).)

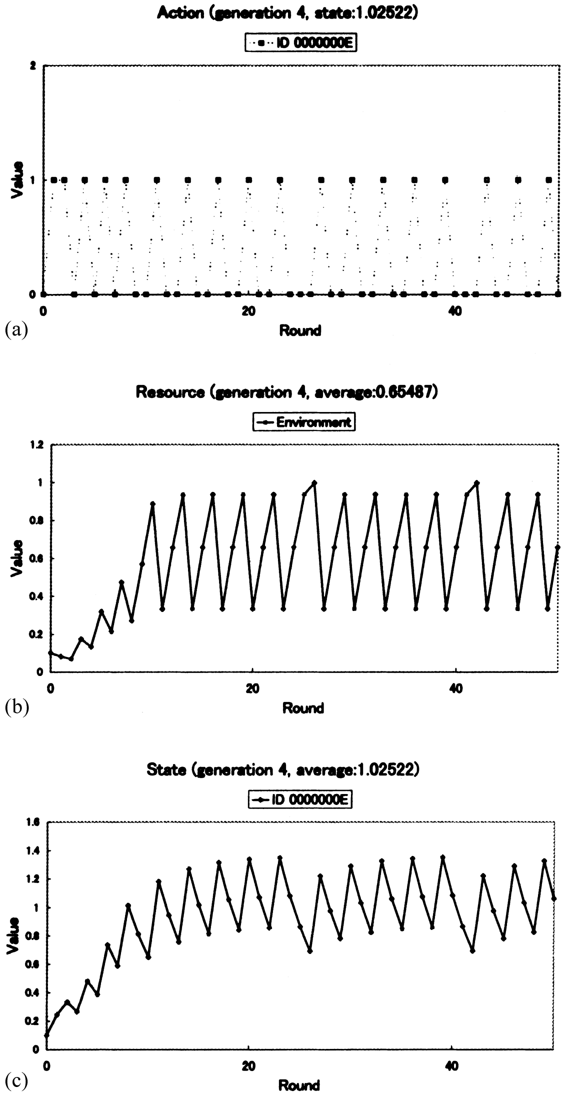

Fig. 6. The third fittest species of the fourth generation: (a) action chart; (b) resource chart; (c) state chart.

5.2.3. Decision making function

The third fittest species at the fourth generation, ID-0000000E, has the decision making function illustrated in the radar chart (see Fig. 7). A decision making function in this paper is represented as follows: [14]

$$\mathrm{mtv}_r(x, y) = \sum_{k\in M}\eta_{kr}x_k + \sum_{l\in N}\theta_{lr}y^l + \xi_r \quad (r \in A).$$

In the present simulation, the parameters were set as $N = \{1\}$, $\Xi = \{1\}$, and $A = \{0, 1\}$ (corresponding to waiting and cutting a tree, respectively). In Fig. 7, mtv_0 is plotted by the dotted line and mtv_1 by the solid line. The coefficient of x_1, η_{1r} corresponds to the axis labeled “environment” (the single environmental resource, the tree), the coefficient of y^1, θ_{1r} corresponds to the axis labeled “Me” (the single player), and the constant term, ξ_r, axis is labeled “const”.

Let us consider the meaning of the decision making function of the player of species ID-0000000E. As seen in Fig. 7, the dotted line crosses the Me axis at a positive value (referred to as θ_{10}) and the solid line crosses it at a negative value (referred to as θ_{11}). Hence, as the value of the state of the player (y^1) increases, the incentive to wait also increases, while the incentive to cut the tree decreases. In the same way, this player stops waiting if the size of tree (x_1) increases, because the dotted line cross the environment axis at a negative value (η_{10}), i.e., he waits when satis-

[14] We use y instead of $\tilde{y}$ here for simplicity.

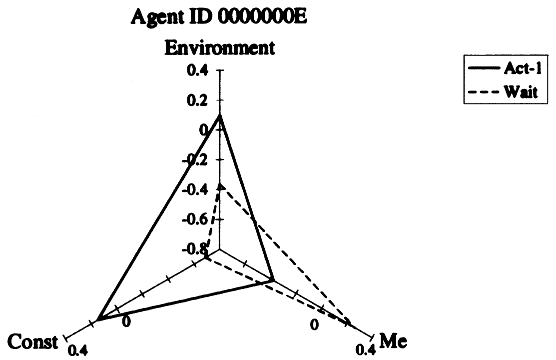

Fig. 7. The decision making function of the third fittest species in the fourth generation. The parameters used in the decision making function of the third fittest species in the fourth generation, ID-0000000E, are completely represented by this radar chart.

fied (his state has a large value) and cuts the tree when the tree has grown large. This is reasonable decision making.

If a player does not consider a particular parameter, the decision making function takes a value of zero for the corresponding coefficient. For example, the player described by Fig. 7 hardly refers to the size of the tree (x_1) as for the incentive to cut the tree (mtv_1), since the solid line crosses the environment axis at a point near zero (η_{11}). In other words, this player's incentive to cut the tree depends only on his satisfaction, not on the size of the tree (mtv_0, however, depends significantly on both x_1 and y^1).

5.2.4. Change of game dynamics with evolution

Figs. 8(a) and (b) depict the behavior on a hill occupied by a player of the fittest species at the 64th generation (ID-000000C5). This player manages the growth of the tree with a period-2 cycle, i.e., as seen in the action chart (Fig. 8(a)), he cuts the tree every second round. Fig. 8(b) shows that the size of the tree changes periodically between two values as follows:

1. growth of the tree based on the natural law (*0.3064* → 0.6222);
2. decision of the player: "wait" (0.6222 → <u>0.6222</u>);
3. growth of the tree based on the natural law (*0.6222* → 0.9194);
4. decision of the player: "cut the tree" (0.9194 → <u>0.3064</u>): The tree becomes $\frac{1}{3}$ as tall as before, and the player acquires lumber in the quantity $0.9194 - 0.3064 = 0.6130$, approximately.

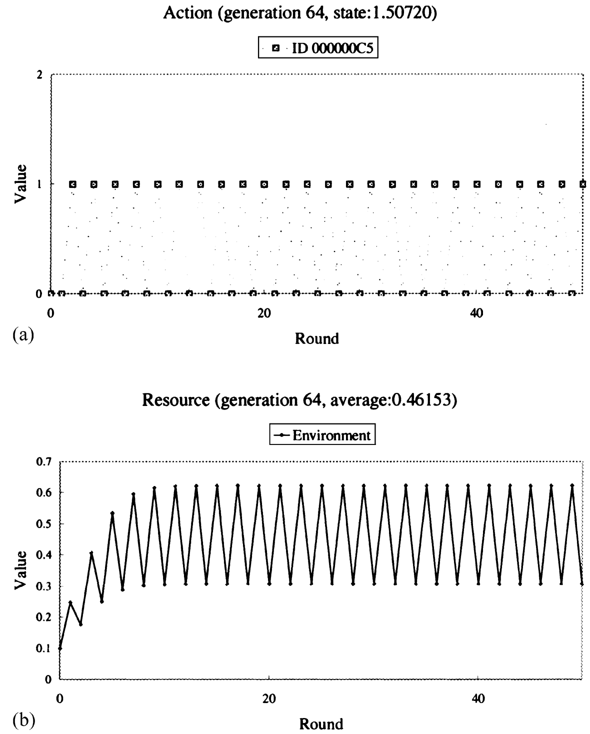

Fig. 8. Managing the game environment with period-2 cycle (the 64th generation).

If the player cuts the tree every third round, he can obtain more lumber from the tree per cutting, but the value of the player's average state over T rounds is smaller in this case. The management of the tree growth with period-2 cycle, which is similar to that in Fig. 8, is also adopted by the fittest species from the fourth generation to the 3604th generation.

Fig. 9 depicts the LD game at the 277th generation. The fittest species, ID-00000345, behaves as shown in the action chart in Fig. 9(a), but the behavior is quite similar to that observed in Fig. 8(a). The periodic growth dynamics of the tree, with the size varying between 0.3 and 0.6, is also realized in this case. However, the species ID-00000345 has, of course, larger fitness than the species ID-000000C5 of Fig. 9. A difference between the dynamics of these two species exists only in the transient, before the state dynamics of the player and the resource completely falls into a

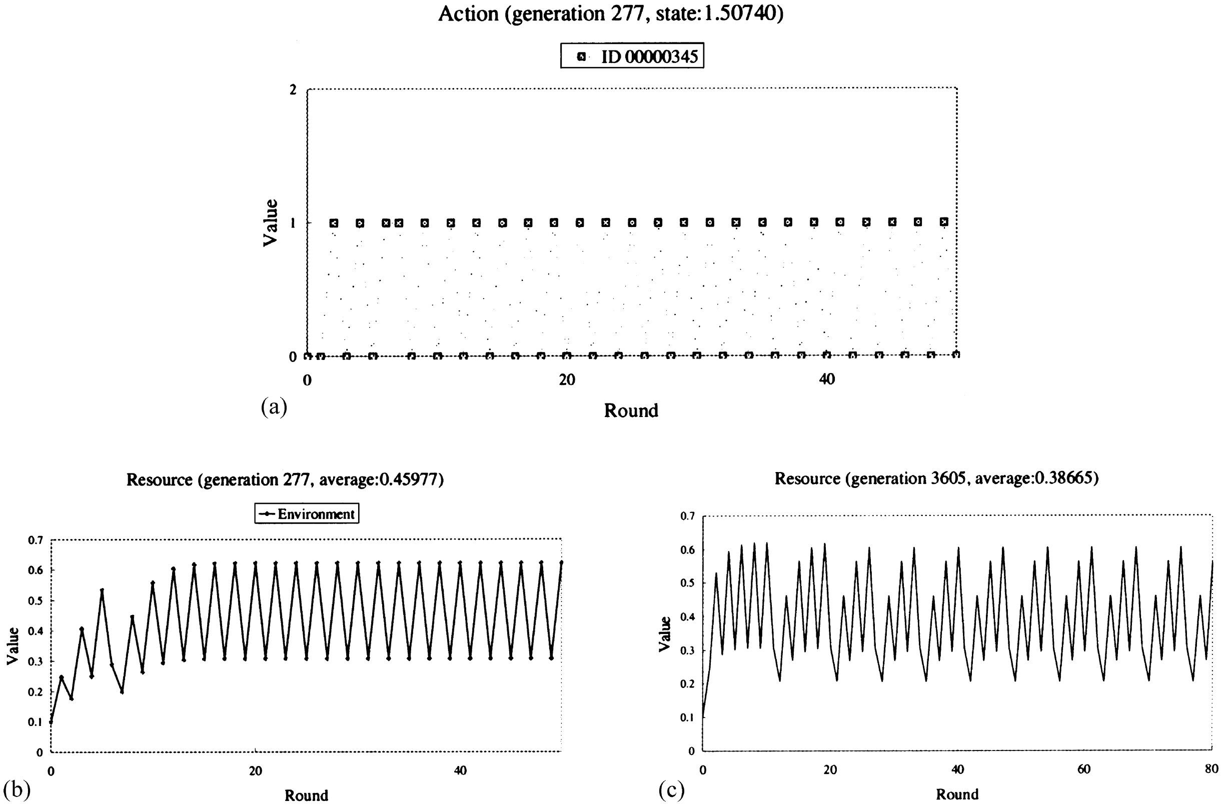

Fig. 9. Evolution in later generations. The action chart (a) and the resource chart (b) of the fittest species of the 277th generation. (c) The resource chart of the 3605th generation.

periodic cycle. This period-2 dynamics is characteristic of the fittest species over many generations, but the variations in the dynamics are exhibited in the transient, before the dynamics falls into a cycle.

The dominance of period-2 dynamics ends at the 3605th generation, when the fittest species becomes one that oversees period-7 dynamics (Fig. 9(c)). A new dominant species of period-23 appears at the 5848th generation and one of period-11 appears at the 8984th generation. Hence, it is seen that the time between changes of the fittest species is long in later generations.

5.2.5. Dynamics of payoff matrix

The fitness chart, action chart, resource chart, and state chart, introduced in the previous section, are useful when we study the dynamics of games. On the other hand, in traditional game theory, a game is usually described by a payoff matrix (in normal-form representation [15]) or, in some cases, by the repetition of a game (iterated game). Here we discuss the relevance of these charts in describing the dynamics of payoff matrices.

If we use the payoff matrix in a DS game, the matrix should change with time. The dynamics of the payoff matrix seen in the fourth generation is shown in Fig. 10(a), which corresponds to the LD game of Fig. 5 in the previous section. The payoff for each action 0 or 1 is plotted versus the round. The payoff for each action is here assumed to be *the size of the lumber that the player obtains by the action*. Hence, the payoff for the action 0 (waiting), shown

[15] A game represented in extensive form, which has a branching diagram, is possible to be transferred into a game in normal form, and vice versa.

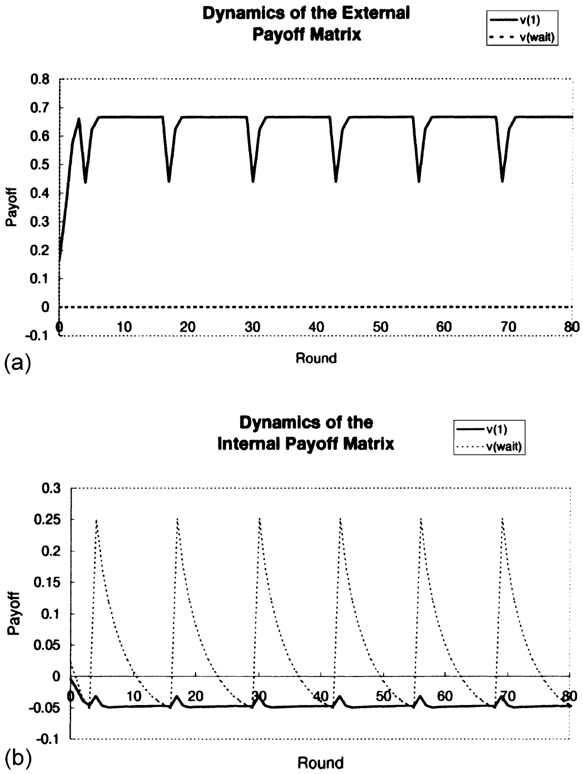

Fig. 10. The dynamics of the payoff matrix (the fourth generation): (a) external payoff matrix; (b) internal payoff matrix. In each figure, horizontal axis shows the round and the vertical axis shows the payoff for each action.

by the dotted line, is always zero, while the dynamics of the payoff for the action 1 (cutting the tree) shown by the solid line is similar to the resource chart in Fig. 5(a). This dynamics is called the dynamics of the external payoff matrix. Note, in any case, the payoff of action 1 is larger than that of action 0.

Meanwhile, in Fig. 10(b), the dynamics of the payoff matrix for each feasible action is given as *the incentive to the action*. For example, the payoff shown by the dotted line is the value of mtv_0 and that shown by the solid line is the value of the mtv_1. We refer to the payoff matrix corresponding to this figure as the internal payoff matrix. The player decides his action simply according to this payoff matrix in every round. In the one-person DS game, each internal payoff is simply the incentive to a certain action r, mtv_r. Thus, this payoff matrix corresponds to that in traditional game theory. In traditional game theory, however, the payoff matrix is given explicitly as a rule, while in the DS game, the payoff matrix is constructed by the player according to his decision making function that determines his way of referring to the game environment (and also to other players' state, in case of multiple-person games).

The dynamics of the internal and external payoff matrices of a player of the fittest species at the 277th generation are shown in Figs. 11(a) and (b) (cf. Figs. 9(a) and (b) in the previous section). By constructing such dynamics of the payoff matrix as shown in Fig. 11(b), the player can acquire relatively high average score in this one-person LD game. Such a construction is possible through the experiences in a game that follows a particular dynamical law.

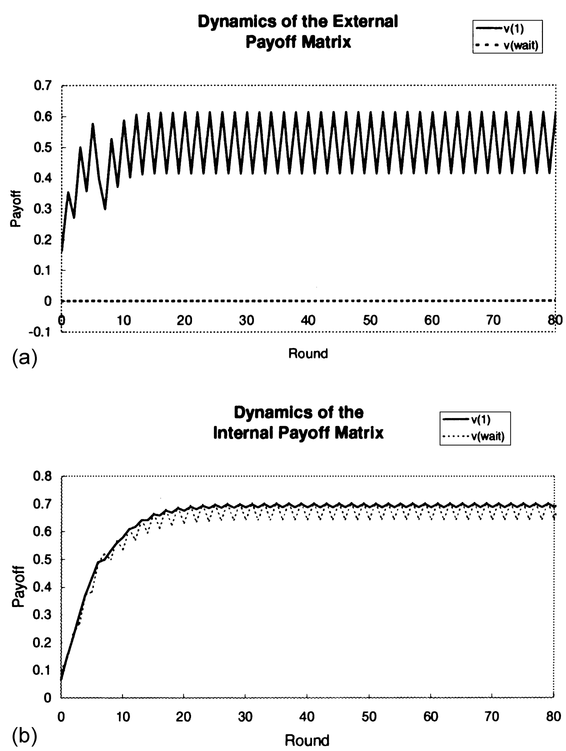

Fig. 11. The dynamics of the payoff matrix (the 277th generation): (a) external payoff matrix; (b) internal payoff matrix.

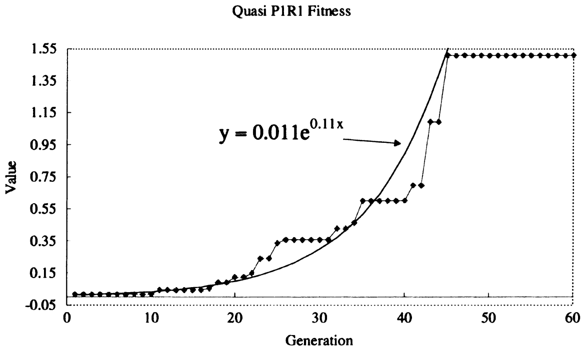

Fig. 12. A simulation of the one-person linear-map-type LD game. The fitness chart of the one-person linear-map-type LD game from the first generation to 60th is plotted with an exponential fitting curve. The optimal game dynamics are realized at the 45th generation.

5.2.6. *Evolutionary simulation of a linear-map-type LD game*

Let us briefly give the result of a sample computer simulation of a one-person linear-map-type LD game. The fitness chart in Fig. 12 for the linear-map-type game shows that the fittest value increases gradually, but not step by step with generation. (Compare this with the fitness chart for the convex-map-type LD game in Fig. 4.) Furthermore, in the present case, the optimal game dynamics are easily realized by the 45th generation.

6. Characteristics of dynamics in LD games

6.1. *The attractor of game dynamics*

In this section, we investigate the evolution of game dynamics by introducing the concept of attractors of game dynamics. In the one-person LD games considered in this paper, we have observed the interplay and mutual influence of the structure of the game and the dynamics exhibited within it: the dynamics of the game environment determined by players' actions are represented by the resource chart, while the players' utilities for the feasible actions are represented by the dynamics of the internal payoff matrices.

The evolution of algorithms toward the maximization of fitness seen in the previous section is, from the viewpoint of DS games, regarded as the process of making the game environment more and more productive by the players. The players attempt to construct a game environment that can bring them greater profits. For example, the player of Fig. 13(a) succeeded in constructing a productive game environment, while the player in Fig. 13(b) failed to do so.

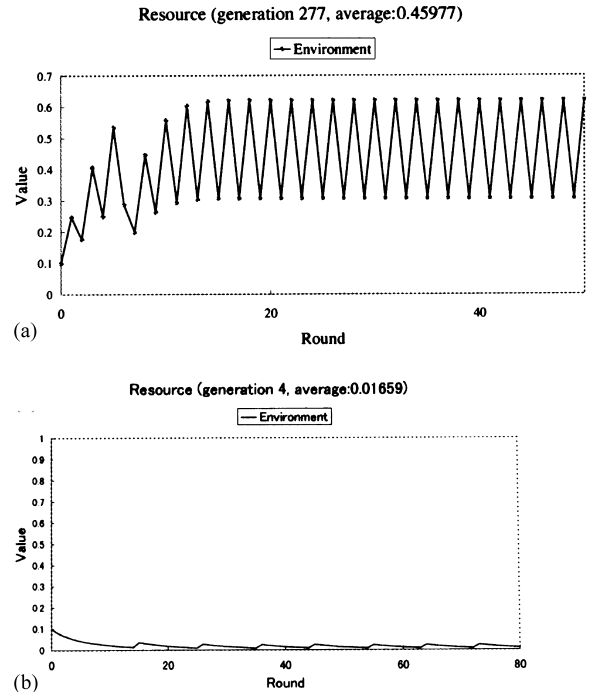

Fig. 13. Two resource charts in the same generation (the 277th generation) for a one-person, one-tree LD game: (a) an example of players who succeeded in constructing a productive game environment; (b) an example of players who failed to do so.

Let us consider Fig. 14. Here the resource chart observed at the 13th generation for a one-person, two-tree convex-map-type LD game (Appendix A) is plotted. As seen, the dynamics eventually falls into a periodic cycle. This periodic dynamics continues up to the final round. It is therefore considered an attractor of the game dynamics. In general, the game dynamics can be divided into an attractor part (which is periodic in the present case), and a transient part. The dynamics corresponding to an attractor, of course, need not be periodic, but can be quasiperiodic, chaotic, and so forth.

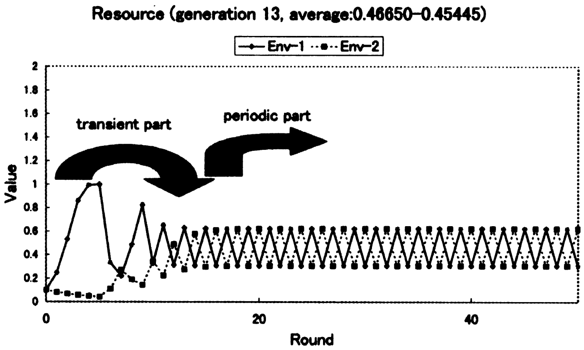

Fig. 14. Periodic part and transient part for a one-person, two-tree game.

In every one-person LD game presented in this paper, the game dynamics observed in every hill changes its pattern with generation, through the evolution of strategies. The effect of evolution on game dynamics can be classified into the following two types:

1. Evolution that changes the pattern of the attractor part, which involves bifurcation to a different attractor.
2. Evolution that changes the transient part.

For example, in the one-tree LD game discussed in Section 5.2, period-2 dynamics appears in the early generation (in the fourth generation). This period-2 attractor dominates over many generations until the pattern with period-7 pervades (in the 3605th generation). During these generations of period-2 attractor, the evolution progresses only about the transient part. The evolution that changes the attractor part is hard to occur. One can say that the evolution with a bifurcation to a new attractor brings about a big innovation, while the evolution about the transient part is a minute improvement to make the game environment productive. Here the big innovation does not necessarily imply a large increase in payoff, but, rather a large structural change in dynamics. Both types of evolution can change the fitness of strategies, but the manners in which they affect the game dynamics are completely different. Only a change in the attractor leads to qualitative innovation in the game dynamics. In two-person (or more) games, the difference is clearer. For example, a change in the transient alters the way in which a certain type of cooperation is formed, while that of the attractor implies a change in the type of cooperation itself.

For the sake of contrast, let us interpret the present LD game as a conventional static game. In this case, we need to construct a mapping from a time series of actions to a payoff function. There are two ways to construct such payoff functions of static games. One involves a map from the set of actions (action 0, action 1, ...) onto scores and another is a map from the parameter space of lumberjacks' decision making functions onto the average score from 400 rounds. No matter which type of map is selected, we can find the change of the payoff caused by the change of the players' decisions in a static game, and furthermore we can find the equilibrium from the payoff structure, at least theoretically. With the static game model, however, we cannot investigate how the change of the payoff is introduced by a change of the attractor or transient dynamics. The evolutionary process of how the strategies change the dynamics cannot be analyzed. The distinction between big innovation (as bifurcation) and minute improvement, which is especially important in multiple-person games, cannot be understood.

6.2. *Relation between game dynamics and strategies*

In this section, we discuss the relation between players' decision making and game dynamics. Traditional static game theory usually deals with the mathematical structure of the payoff function of each player: [strategy space] $\mapsto$ R (payoff). For the LD game, we can also describe it as follows:

$$\times_i A^i \mapsto R \quad (i \in N),$$

where A is the set of possible actions, N the set of players, and R denotes the payoff in each round of the LD game. If we assume that players have enough computational power to allow for the all players' actions in all the T rounds, this structure can also be described as follows:

$$\times_i \{\text{the parameter space of } i\text{th player's decision making function}\} \mapsto R \quad (i \in N),$$

where R denotes the average score from all the rounds. One representation of the above payoff function is given by the average score landscape, to be presented later. As stated in the previous section, the concrete game dynamics produced by the players' decision making is ignored in the above payoff functions. Hence, we cannot discuss the role that the structure of the game dynamics plays in the evolution of decision making.

On the other hand, the structure involving the players' decision making and the game dynamics is investigated using AGS diagram to be introduced in Section 6.2.1. To prepare for its discussion, let us interpret the DS game as an orbit in the phase space.

The dynamics of DS game are described by the time series of $x(\in R_+^m)$ and $y(\in R_+^n)$. Let us call the $(m+n)$-dimensional phase space the game environment = player (GP) space, for simplicity. The game dynamics corresponds to an orbit in the GP space.

In the present paper, each motivation map in the decision making function is given by a one-dimensional map (Section 4.3.2), and the GP space is divided into several subspaces by the following $(m+n-1)$-dimension hyperplane:

$$\mathrm{mtv}_p(x, y) = \mathrm{mtv}_q(x, y) \quad (p, q \in A,\ p \neq q).$$

The player's action is determined uniquely by the subspace in which the present orbit point (x, y) resides. Of course, these hyperplanes are different for each player. For example, the dimension of the GP space is 2 when an LD game is composed of one person and one tree $(N = 1, M = 1, A = \{0, 1\})$. This GP space is divided into two subspaces by a straight line (i.e., a two-dimensional hyperplane) defined by the equation $\mathrm{mtv}_0(x, y) = \mathrm{mtv}_1(x, y)$. The player selects action 0 if the current phase is within the subspace $\{(x, y) \in R_+^{m+n} | \mathrm{mtv}_0(x, y) > \mathrm{mtv}_1(x, y)\}$ and action 1 otherwise. In this way, the structure of partitioning of GP space and the current phase determines the orbital point in the next phase uniquely, and such a partitioning of GP space enables the player to select his or her action for any given game states.

In general, we can use a more complicated decision making function with motivation maps of dimension greater than 1 that divides the GP space into subspaces with several hypersurfaces (i.e., not necessarily hyperplanes). With this implementation, complex information processing and strategies are possible.

6.2.1. AGS diagram

Here let us consider a simplified model of a one-person, one-tree LD game in order to investigate the effect of the change of the decision making function upon the attractor of the game dynamics. We make two simplifications. First, the player never refers to his own state, y. That is, the player makes his decision by considering only the size of the tree. Second, the player cuts the tree if its size exceeds a certain value, called the decision value, x_d.[16] The decision value uniquely determines the time series of the phase (x, y). The attractor of the time series can correspond to a fixed point, periodic, quasiperiodic, or chaotic motion, depending on the dynamical law — including the natural law — of the system.

As in a bifurcation diagram, we have plotted the x value at the attractor in Fig. 15(a), as a function of x_d. The figure shows how the attractor of game dynamics changes with the parameter in the decision making function. Let us call such a figure an AGS diagram (the transition of the attractor of the game dynamics versus the change of the strategy). With the AGS diagram, one can study how the nature of the game dynamics shifts among various states (fixed point/periodic/chaotic game dynamics, or productive/unproductive game dynamics, etc.) with a change in the decision making.

The following two characteristics of Fig. 15(a) are noted:

1. For each decision value, its corresponding attractor is always a periodic cycle.

---

[16] The decision value x_d is introduced by stipulating $x_\mathrm{d} = -(\xi_1 - \xi_0)/(\eta_1 - \eta_0)$ for the parameters in the player's decision making function. This can be seen by noting the following two points:

1. x_d must satisfy the condition $\mathrm{mtv}_0(x_\mathrm{d}, y) = \mathrm{mtv}_1(x_\mathrm{d}, y)$ because the change of a player's decision between action 0 and action 1 occurs at this point.
2. The parameters characterizing reference to the player himself are set as $\theta_0 = \theta_1 = 0.0$, because the player is supposed to refer only to the size of the tree.

Thus, the player will cut the tree if and only if $x \geq x_\mathrm{d}$.

2. There are infinite number of plateaus, in which the attractors remain unchanged over some range of decision values. For examples of such plateaus, see the period-2 and period-3 plateaus in Fig. 15(a).

In the simulation of the one-person, one-tree LD game (Section 5.2), the period-2 attractor of the dynamics of x dominates the population for many generations (up to the 3600th generation). This attractor corresponds to the two parallel segments plotted around $x_{\mathrm{d}} = 0.8$ in Fig. 15(a). (We refer to such parallel segments as plateau.)

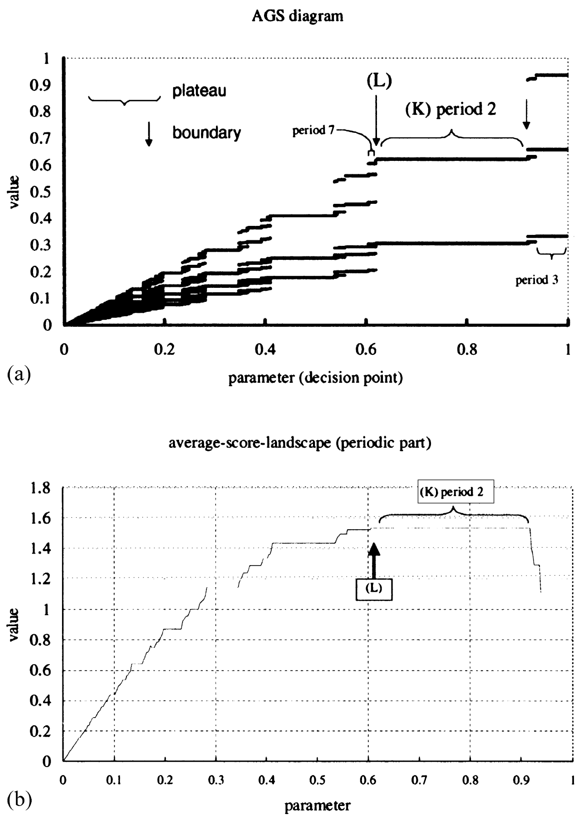

Fig. 15. (a) AGS diagram of a one-person, one-tree LD game. Change of the attractor with the strategy is plotted. In this figure, the player fells the tree when the size of the tree, x, exceeds the decision value x_{d}. A set of values of x at the attractor (all the values that x takes between the 200th and 400th rounds) are plotted with the decision value represented by the horizontal axis with an increment of 0.0025. For example, the two parallel straight segments around $x_{\mathrm{d}} = 0.8$ show that the dynamics of x are attracted to the period-2 cycle between values around 0.3 and 0.6. We call this type of figure an AGS diagram. (b) Average score landscape: the average score is plotted as a function of the decision value x_{d}. The average score is the score that the player gets during the time that x is at an attractor.

The AGS diagram displays the change of the attractor with the change of the control parameter, like the bifurcation diagram often used in the analysis of dynamical systems. This diagram is effective to extract characteristics of dynamical systems of the DS game. However, the AGS diagram possesses a meaning in addition to that of bifurcation diagram. The difference lies in the decision controlled by the control parameters. In the DS game, the control parameter is given by the decision maker, who is within the system. This contrasts with the external bifurcation parameter in dynamical systems. In other words, what attractor we actually observe among the attractors in the AGS diagram is decided by the player existing in the system.

Corresponding to Fig. 15(a), the average score that the player obtains during the attractor part of the dynamics is plotted in Fig. 15(b) as a function of the decision value x_{d}. We call this diagram the average score landscape. The optimal decision value seems to exist in the period-2 plateau around 0.8, as far as we can see with the scale of this figure. Consequently, the best strategy for the player seems to construct this period-2 dynamics, which is indeed observed in the early stage of the LD game simulation. However, by examining the close-up of the left-hand side of this period-2 plateau in Fig. 15(a) (indicated by the arrow with (L)), infinite number of plateaus are found accumulating there, where dynamics more profitable than those of period-2 exist.

6.2.2. Structure of dynamics in AGS diagrams

Let us study the AGS diagram in Fig. 15(a) in more detail. Fig. 16(a) is a close-up of the part of Fig 15(a) around the arrow indicated by L (from $x_{\mathrm{d}} = 0.621$ to 0.623). Fig. 16(b) is a close-up of the boundary part of Fig. 16(a) indicated by the arrow with P, and (c) is a close-up of (b), and (d) of (c). This sequence of figures shows that this boundary part, next to the period-2 domain, has a fractal structure, and it

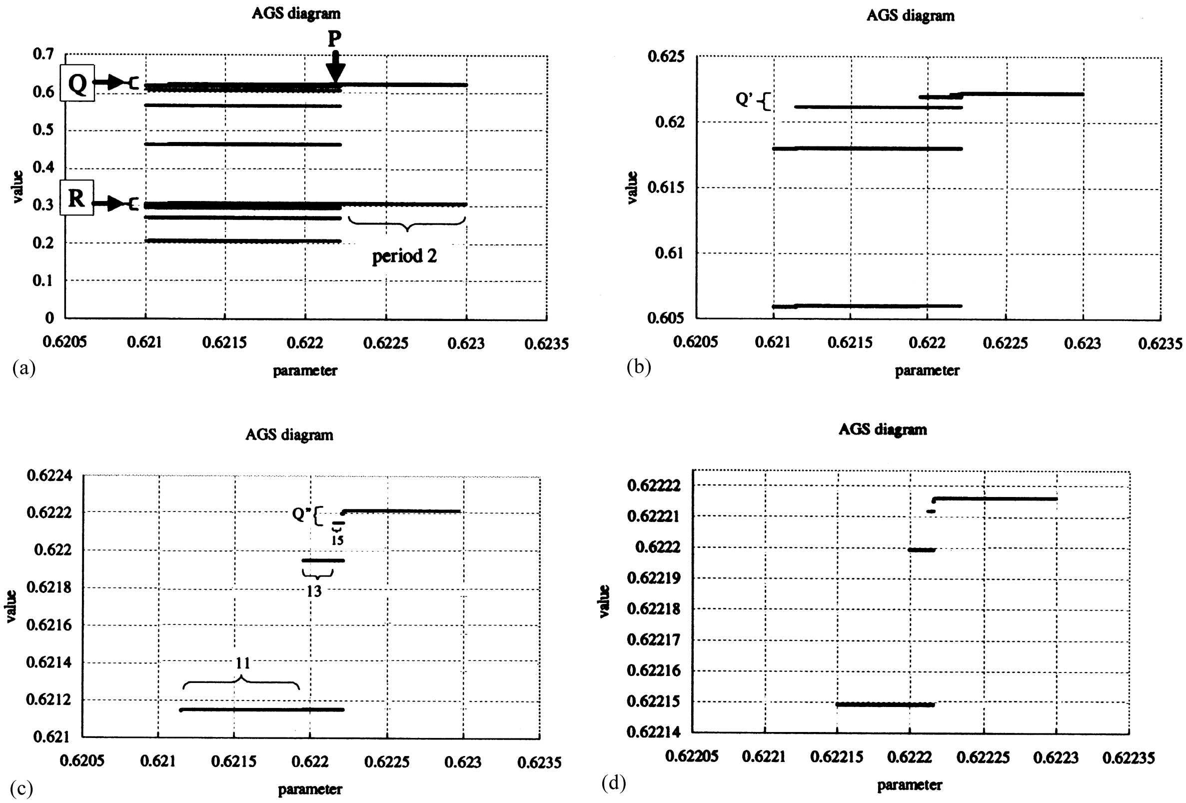

Fig. 16. Fractal structure of the AGS diagram: (a) close-up of a part of Fig. 15(a) (from $x_d = 0.621$ to 0.623); (b) close-up of the region indicated by Q in (a); (c) close-up of Q′ in (b); (d) close-up of Q″ in (c). As shown in (a)–(d), the left-hand side of the period-2 plateau has a fractal structure and contains infinite periodic attractors.

contains infinite periodic attractors. We call this region domain L. The pieces labeled 11, 13 and 15 in Fig. 16(c) correspond to dynamics of periods 11, 13 and 15. [17]

A fractal structure also exists in the average score landscape, as shown by the successive close-ups given in Figs. 17(a)–(d). In the domain L with the accumulation of infinite periodic attractors, some attractors have a higher average score than that of the period-2. In particular, the highest average score is achieved in the period-11 domain. This one-person LD game has many fractal domains like the domain L of Fig. 15(a). [18]

6.2.3. *The effect of dynamical structure on the evolution*

The average score landscape exhibit a complete fractal structure if we consider only the attractor part of the game dynamics (see Fig. 18(a)). On the other hand, the average score for a finite number of rounds, including the transient part, can exhibit a more complicated structure, although it consists of a finite number of plateaus (compare Fig. 18(b) with Fig. 18(a)). Here,

[17] The period increases by jumps of 2 as x_d increases. This is due to the existence of the similar structure in the lower part, R, in Fig. 16(a). Such period adding structure is rather common in phase lockings from quasiperiodicity [9].

[18] These local structures of the AGS diagram and the average score diagram in this LD game are examples of the so-called devil's staircase (e.g., in the Cantor function), a general feature observed in phase locking from quasiperiodicity (see, e.g., [11,14]).

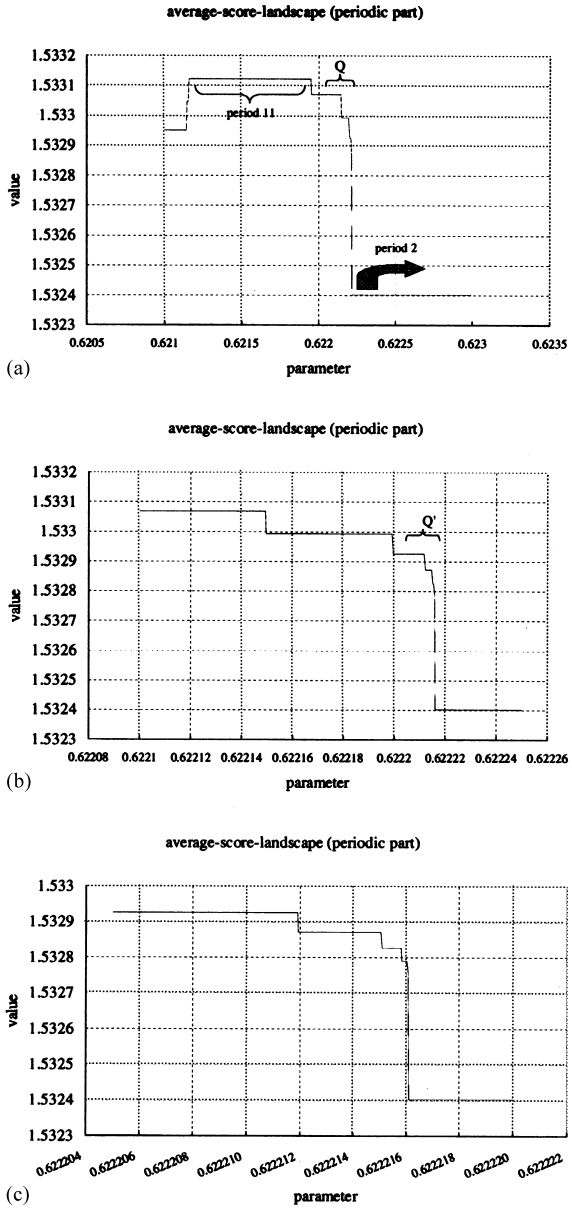

Fig. 17. Fractal structure in the average score landscape: (a) close-up of the part from $x_d = 0.6221$ to 0.62215 in Fig. 15(b); (b) close-up of the part indicated as Q in (a); (c) is that of (b). As shown in (a)–(c), the average value landscape contains a fractal structure around the boundary to the period-2 domain.

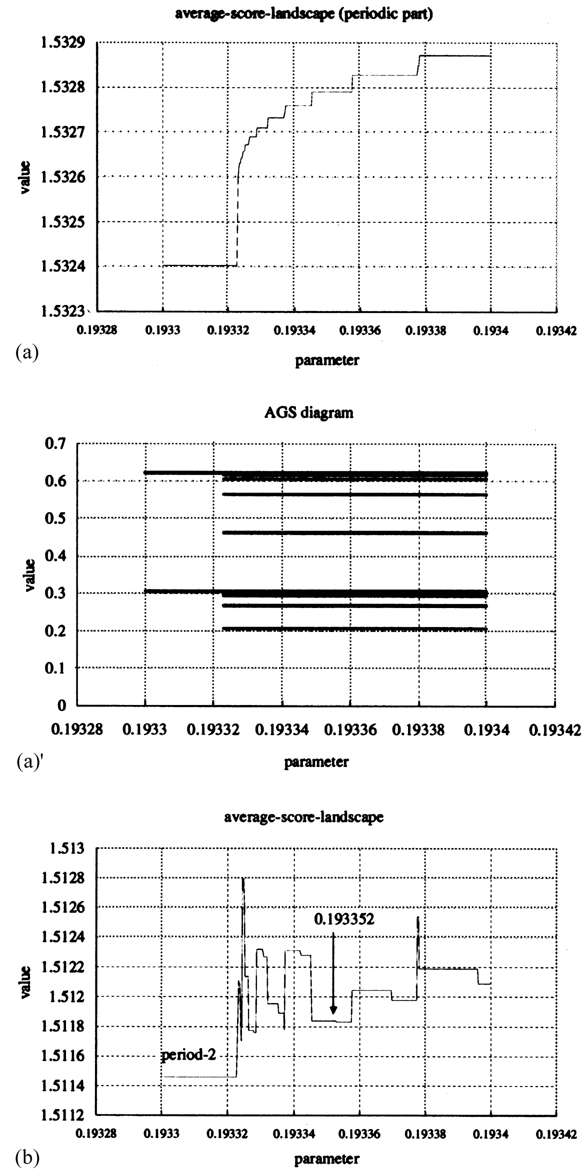

Fig. 18. Complicated structure in the average score landscape: (a) and (b) are the average score landscapes based on the decision making function of a player who appears in the one-person, one-tree simulation (the 5848th generation). (a)′ is the corresponding AGS diagram. In each figure, the average score in the LD game is plotted as a function of the parameter of the player's decision making function, η_{11}. (a) gives the average score only with the attractor part of the dynamics, while (b) gives the average score over all rounds from the first to the 400th, including the transient part. The monotonically increasing devil's staircase, observed in (a) is replaced by a complicated but finite structure in (b). The player in question has a decision value $\eta_{11} = 0.193352$, as indicated by the arrow in (b), and is not of the fittest species.

Fig. 18 displays sample average-score landscapes plotted as functions of the parameter η_{11}. These figures were obtained from the decision making function of a player who actually appears in a one-person, one-tree

LD game, though this player does not belong to the fittest species. As shown in Fig. 18(a) (with the attractor part only), the player's average score increases monotonically with η_{11}. Fig. 18(a)′ is the corresponding AGS diagram. (In this one-person LD game, any decision making function results in a periodic state as an attractor, and the AGS diagram always consists of parallel straight segments.)

The average score landscape for a finite number of rounds including the transient part, on the other hand, as plotted in Fig. 18(b), does not change monotonically with η_{11}, and has a rugged landscape. In this case, not only the nature of the attractor but also the path how the attractor is reached has some importance for evolution. Of course the evolutionary process differs according to the number of iterations of the game. For instance, the importance of the attractor part increases as the number of iterations increases.

Note that a complex landscape is also seen for the fittest species. Fig. 19 gives an example of the average score landscape plotted for the decision making function of the fittest species (ID-2A44) at the 3605th generation. The average score is plotted as a function of the decision parameter η_{10}, in (a) for the attractor part and in (b) over 400 rounds with the transient part. (Of course, in the evolutionary LD games in this paper, the transient part is not ignored when the average score of a lumberjack is calculated.) Here, the species in the figure adopts a period-7 attractor of the game dynamics. This period-7 dynamics corresponds to the period-7 plateau in Fig. 15(a), and it is more productive for the player than the period-2 dynamics, though it is less productive than the dynamics in the domain L. As shown in Section 5.2.4, this species dominates the population for many generations (about 2200). Why is this species able to dominate for so long? The answer is given in the landscape of Fig. 19(b) obtained for a finite number of rounds with transients. As shown in the figure, the score has a local maximum at the plateau around $\eta_{11} = 0.193352$. Although the strategy with this decision value is not the globally optimal strategy, a large amount of mutational change is needed to escape from this local maximum. In a game with a finite number of rounds, the landscape is rugged, in contrast with the case of an infinite rounds, and the evolution process does not necessarily reach the optimal strategy easily.

6.2.4. Dynamical structure of linear-map-type LD game

Fig. 20(a) is the AGS diagram of the one-person linear-map-type LD game. This figure shows that the

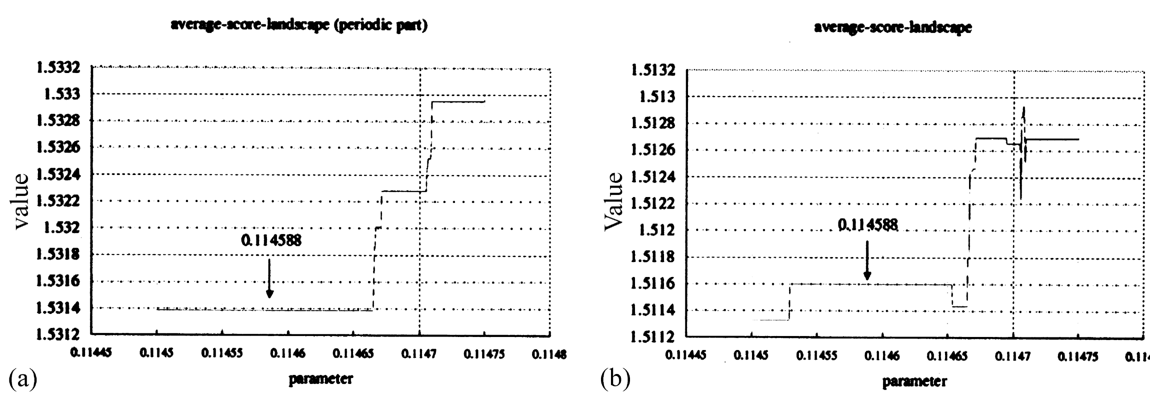

Fig. 19. Effect of the structure of game dynamics on the player's evolution. (a) and (b) are the average score landscapes constructed using the decision making function of the player who is from the fittest species (ID-2A44) of the 3605th generation in the actual one-person, one-tree LD game of Section 5.2.4. The horizontal axis corresponds to the parameter of the decision making function, η_{10}, while the vertical axis corresponds to the average score. In (a), the average value only about the periodic part is plotted. The average score for all rounds, including the transient part (from the first round to the 400th), is plotted in (b). The actual value for this player's η_{10} is 0.114588, as is indicated by the arrow in both of these figures.

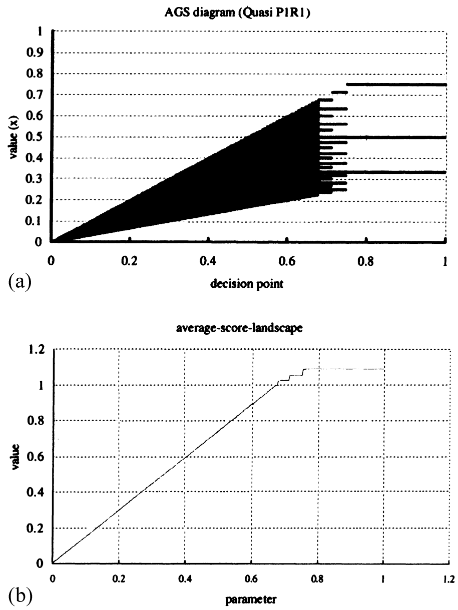

Fig. 20. The linear-map-type one-person LD game. (a) The AGS diagram for the decision value x_d. Quasiperiodic attractors appear for $x_\mathrm{d} \leq \frac{2}{3}$. (b) The average score landscape from the 200th to the 400th round. The landscape is given by a straight line for $x_\mathrm{d} \leq \frac{2}{3}$.

dynamics are attracted to quasiperiodic motion if $x_\mathrm{d} \leq \frac{2}{3}$, although the AGS diagram of the convex-map-type LD game has a periodic attractor for any decision value (Fig. 15(a)). Fig. 20(b) is the corresponding average score landscape, where the landscape is not stepwise, but a straight line for $x_\mathrm{d} \leq \frac{2}{3}$. As shown by this landscape, the optimal decision value satisfies $x_\mathrm{d} \geq 0.75$ for the one-person linear-map-type LD game. In the linear-map-type LD game of Section 5.2.6, we have observed the smooth and quick evolution toward the optimization of the fitness. Such evolution is caused by the no-rugged structure seen in Fig. 20.

7. Discussion

7.1. Dynamical structure and evolution of strategies in DS game

In order to discuss the relation between the structure of dynamics in DS games and the evolution of strategies, let us consider, as a real example, the three-field system of crop rotation that prevailed in medieval Western Europe [24].

This situation seems to have some relation to the viewpoint of DS games, because here a player's decision has some effect on the variables that determine the game environment, while the state of these variables affects the player's state (e.g., the player's nutritional state, which might be changed by the amount of available crops).

The three-field system, which was a modification of the two-field system that had prevailed around the Mediterranean was an effective method for farmers or peasants during that time to manage the dynamics of the states of their farmland and to harvest the crops every year consecutively. Due to this innovation, they became able to maintain the farmland's fertility, to prevent moisture from decreasing too much, and to control weeds. With the three-field system, each farmland is divided into three equal parts, the winter field, the summer field, and the fallow, and these three fields are used alternately with a period of 3 years. In case of the two-field system, farmland is divided into the summer field and the fallow, used alternately with a period of 2 years. (In addition, at the time of the industrial revolution, there appeared a 4-year crop rotation, called the "Norfolk System" that included a year of growing feed for livestock. However, the structure and the concept of this system were completely different from the two/three field systems.) Not all of the areas that used the two-field system switched to the three-field system, because this innovation was caused partly by the climate change that occurred when the two-field system spread from the Mediterranean region into Western Europe. However, it is sure that the development of the three-field system was based on the experience of the two-field system.

Now, let us consider the three-field system, which appeared as an improvement of the two-field system, from the viewpoint of DS game modeling. (Note that, the following description of the three-field system is quite simplified because we ignore the medieval social system, the climate, and issues of livestock feed and excreta [24].) Approximately speaking, the feasible actions in the three-field system can be classified into giving the field the year off (fallow)[19] and using the field to produce a crop.

The main issue here is in what order these actions are carried out. In addition, it is also an important issue to determine how many equal parts the farmland should be divided into. In fact, these two issues are inseparable. For example, suppose that the best way to manage the farmland is to lay a given plot of land fallow for three consecutive years, then produce a crop for four consecutive years, and repeat this 7-year cycle indefinitely. Then the farmland must be divided into seven (or a multiple of seven) plots in order to harvest a crop every year as uniformly as possible. The action of leaving the land unused for 3 years allows the field to regain its growing potential, although it does not offer any short-term gain, while the action of using the land to raise crops, of course, gives immediate reward but depletes the soil of nutrients. In the three-field system, the dynamics of the state of the fields is approximately cyclic with a period of 3 years. However, it is hard to believe that the period of the growth of each kind of crop and the period of the state dynamics of the fields are precisely periodic. If some adjustment of the periodic dynamics is used effectively, a more productive system may be achieved by adopting more complicated dynamics, probably, with a longer period. Of course, the evolution to a better strategy may require complicated innovation, and farmers may stay a strategy with some local optimum.

When an infertile field is kept as fallow for some length of time, there is a possibility that the field will recover its productivity. On the other hand, if the field is kept as fallow too long, long-range productivity will decrease. One needs a careful balance to manage the dynamics of this game environment. A farmer may try to change the current period-3 system into one of period-4 or of period-5. However, among these systems, the probability that they can encounter more profitable systems than three-field system is possibly small. The period-2 or-3 dynamics is, as it were, metastable dynamics for farmers.

We have so far considered the three-field system from the viewpoint of innovation regarding the dynamics of the game environment in a one-person DS game. Such consideration from the DS game point of view has an advantage in case we consider the issues of the real-world game. By modeling concretely the real situation as a DS game that describes concretely the nature of the dynamics observed in game environment, we may theoretically determine a strategy with a suitable game dynamics. For example, suppose that the set of strategies — the decision making practices of the players — that determines the game dynamics (e.g., the above-mentioned period-3 dynamics) is given as a rule explicitly. Traditional game theory is able to determine the solution in the strategy space, at least, as long as it is a one-person game or a two-person zero-sum game. Furthermore, we can find out where metastable strategies exist in the strategy space provided that the distance between strategies can be defined. However, conventional game theory cannot provide any information about what kind of strategies are allowed in the dynamics of the world. For example, game theory can confirm that a seven-field system is a solution if it is given as an element of the strategy set, but, otherwise, it cannot find this solution in principle. On the other hand, when we describe a model as a DS game, it can be shown that there is a decision whose result is a period-7 cycle by using, e.g., an AGS diagram, which is based on the method used in (discrete) dynamical system.[20] Another advantage of the viewpoint of DS game is that we can investigate the relation

[19] Although it is the year off, some degree of labor is necessary, such as removing weeds, plowing, allowing the cattle to graze, and so on.

[20] Game dynamics in DS games can be described by the initial state, (x, y), and by the composite of two types of maps (Section 2.4), the natural law, u, and the effect of players' actions, v, as follows:

$$\cdots v \circ u \circ v \circ u \circ v \circ u \circ v \circ u.$$

Here, each v differs according to the value of $(x(t), y(t))(t = 0, 1, 2, \ldots)$. (More precisely, $u(x(0), y(0))$, $u(v(u(x(0), y(0))))$, $u(v(u(v(u(x(0), y(0)))))), \cdots$). For example, in the one-person, two-tree LD game, v can be v_1 (fell tree 1), v_2 (fell tree 2) or the identity operator (fell no tree). An example of the composite corresponding to the above map is as follows:

$$\cdots v_2 \circ u \circ u \circ v_1 \circ u \circ v_2 \circ u \circ u \circ u \circ v_2 \circ u \circ v_1 \circ u.$$

between the evolution of the players' strategies and the resulting shift of the game dynamics. Game theory can examine if some strategy is a rational solution (e.g., at an equilibrium point) at the level of the payoff structure. Execution of the strategy, however, may be difficult to realize through evolution if the solution in the payoff structure is within a chaotic domain in the AGS diagram, in the sense that the size of algorithms to calculate the payoff cannot be shortened by any method.

7.2. Models of game theory and models of physics

Finally, let us discuss the general advantage of the DS game model over other models. DS game modeling is suitable to study the evolution/learning of the decision makers living in the world that can be described by dynamical systems. Models of game theory certainly have a strength in dealing with the issues of decision makers who interact with each other. On the other hand, descriptions using the models of dynamical systems are relevant to investigate the nature of our dynamic world from the outside.

By applying models in physics, we can know, e.g., the trajectories of all the possible physical states. However, the game viewpoint is indispensable if there exits decisions of players in the course of the dynamics whose decisions are based on their own norms of rationality and whose decisions can affect the physical states. Only from the game theoretical point of view we can consider which state will be selected by decision makers from among physically realizable states, at least at the present time. In dynamical systems, one can study the change of dynamics with the change of external control (bifurcation) parameters, but one cannot discuss which control is possible within the system. To study the collective behavior of a group of players (e.g., the dynamics of stock prices), methods of, e.g., statistical mechanics, in which players can be regarded as statistically particles, may be useful. (In other words the individuality of each player could be disregarded.) However, for the study of the groups of living organisms or human society, the distinction between interacting particles and interacting decision makers is essential. The latter are active agents in their game environments and each makes decisions, basically, so as to increase his utility (fitness). [21]

On the other hand, descriptions consisting of game theory models are not congenial to dynamics by nature, and so, cannot touch upon issues that can be studied only at the levels of dynamics. For example, the convex-map-type LD game, which has the devil's staircase in the AGS diagram, shows the evolutionary phenomena with a stepwise innovation. On the other hand, the evolution is smooth, with rapid convergence to an optimal fitness, for the simple dynamic structure in the linear-map-type LD games. As stated in Section 4.5, a linear-map-type game has the same structure as a convex-map-type game from the static point of view. However, a difference often appears at the level of game dynamics and evolution of strategies (At the level of multiple-person games, the present modification does not eliminate the common social dilemma that exists in the previous version.) That is, several different DS games would be categorized into the same static game if modeled by conventional game theory, but there is the possibility that these games have completely different natures at the level of the dynamical structure, especially when evolution and learning are involved. [22]

7.3. Summary

What is needed in DS game modeling is only the description of g and f. With these, we can simply model, in a sense, common situations where decisions are embedded into a world that is basically subject to natural laws. Of course, this simplicity of the modeling does not necessarily imply the facility of the analysis. However, descriptions from the viewpoint of DS games and analysis using AGS diagram enable us to investigate the relation between the nature of game dynamics and the evolution of strategies. DS games

[21] Of course, it is an important issue how a decision maker — or an autonomous optimizer — emerges from a system described by pure dynamical systems, but this is beyond the scope of the present paper.

[22] This difference becomes more fatal in the multiple-person games, as the formation of cooperation becomes important as will be described in subsequent papers.

can deal with issues that involve both aspects of our world: the world as a dynamical system and the world inhabited by decision making subjects. Both of these aspects are indispensable even in one-person games, because players' decision making mechanisms, changing through evolution and learning, can be affected by exploring the structure of the AGS diagram.

Acknowledgements

The authors would like to thank T. Ikegami and S. Sasa for useful discussions. This work is partially supported by Grants-in-Aid for Scientific Research from the Ministry of Education, Science and Culture of Japan. The first author is supported by Japan Society for the Promotion of Science under the Contract No. RFTF96I00102.

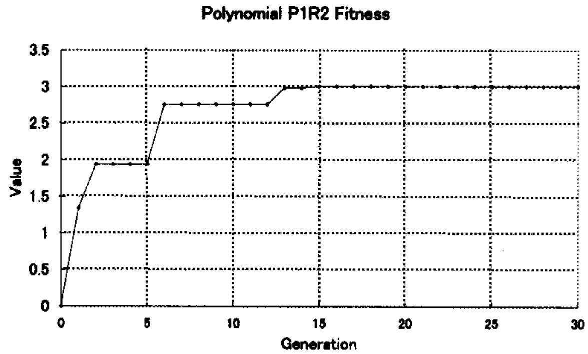

Fig. 21. Fitness chart in the early generations of a one-person LD game with two trees (from generation 1 to 30).

Appendix A. Managing multiple dynamical resources

In this section, we present the results of a one-person, two-tree convex-map-type LD game to

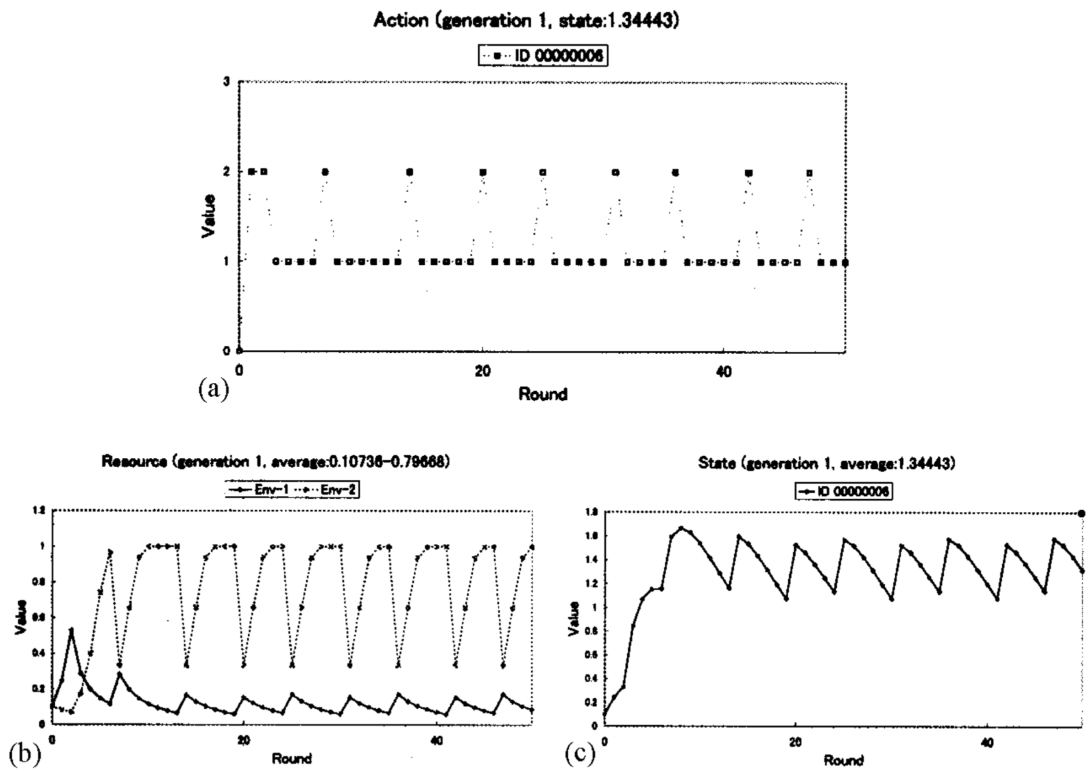

Fig. 22. The fittest species in the first generation. (a) The action chart. The player's action is plotted as a function of the round number. In addition to the actions 0 and 1 (doing nothing and cutting tree 1), action 2 (cutting tree 2) is available for the player in this game. (b) The resource chart. (c) The state chart.

study how a player manages multiple dynamical resources. Fig. 21 is the fitness chart for this simulation. As in the case of a one-tree, one-person LD game, the fitness value increases stepwise and monotonically. The fittest species can change at later generations, but the change is not frequent, and the resulting increase of the fitness value becomes much smaller.

A.1. Raising one particular tree: the behavior of the fittest species in early generations

The fittest species in the first generation behaves as in the action chart (Fig. 22(a)). The player of this species usually cuts tree 1 for several rounds successively, and as a result, this tree becomes smaller and smaller (Fig. 22(b)) as does the amount of lumber he obtains from each cutting (Fig. 22(c)). On the other hand, tree 2 becomes larger during this time. Then, the player cuts tree 2, and thereby obtains a great profit (Fig. 22(b)). In particular, he succeeds in raising the tree 2 and gains a large profit from it. From the first round to the end, he never simply does nothing, but he mostly cuts tree 1 and allows tree 2 to grow, cutting it only occasionally. In this generation, the players of other species with

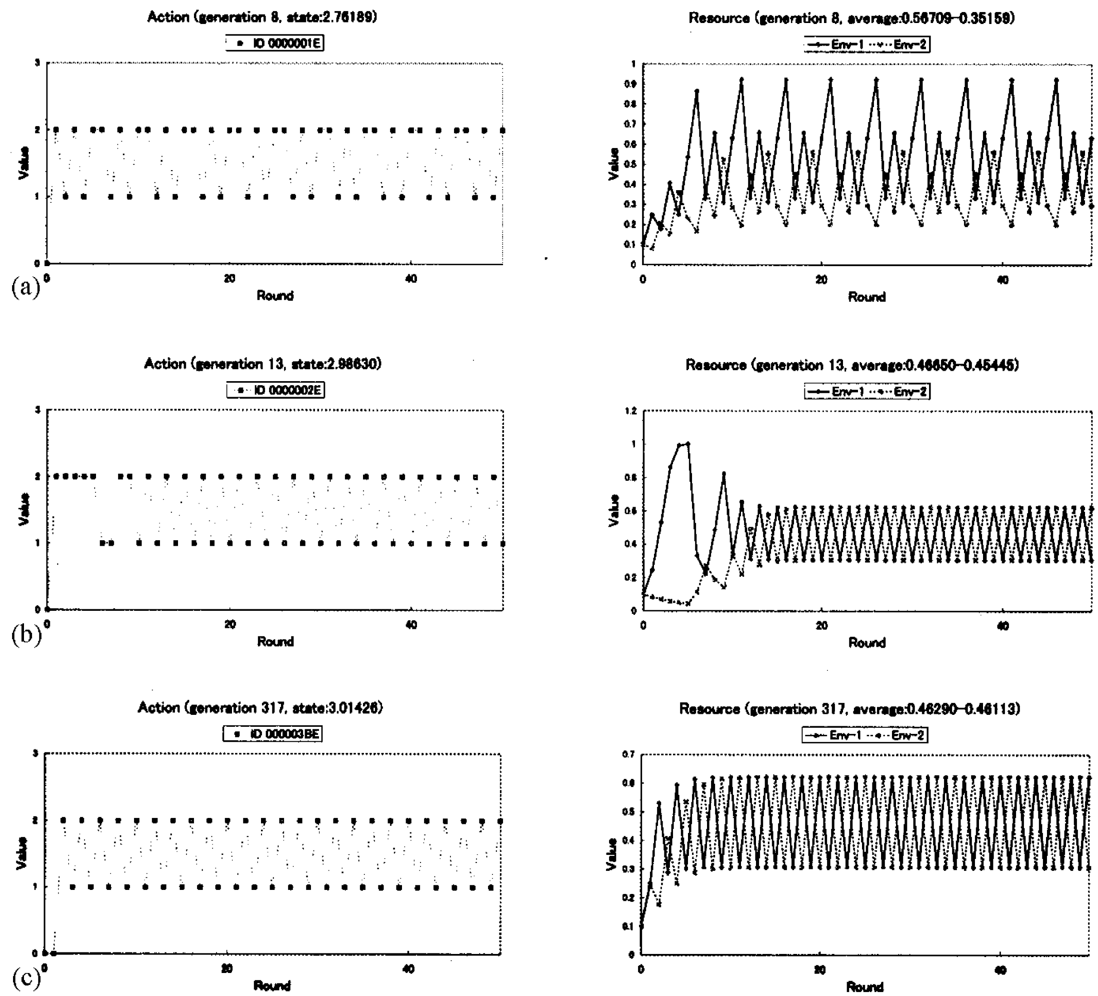

Fig. 23. Evolution with increasing generation number. The figures on the left-hand side are the action charts and those on the right-hand side are the resource charts. The figures in row (a) correspond to the LD game for the player from the fittest species in the eighth generation, those in row (b) for that in the 13th generation, and those in row (c) for that in the 317th generation.

lower fitness cut only a particular tree or cut no tree.

A.2. Evolution toward optimization

A rough sketch of the evolution in this one-person, two-tree LD game is as follows:

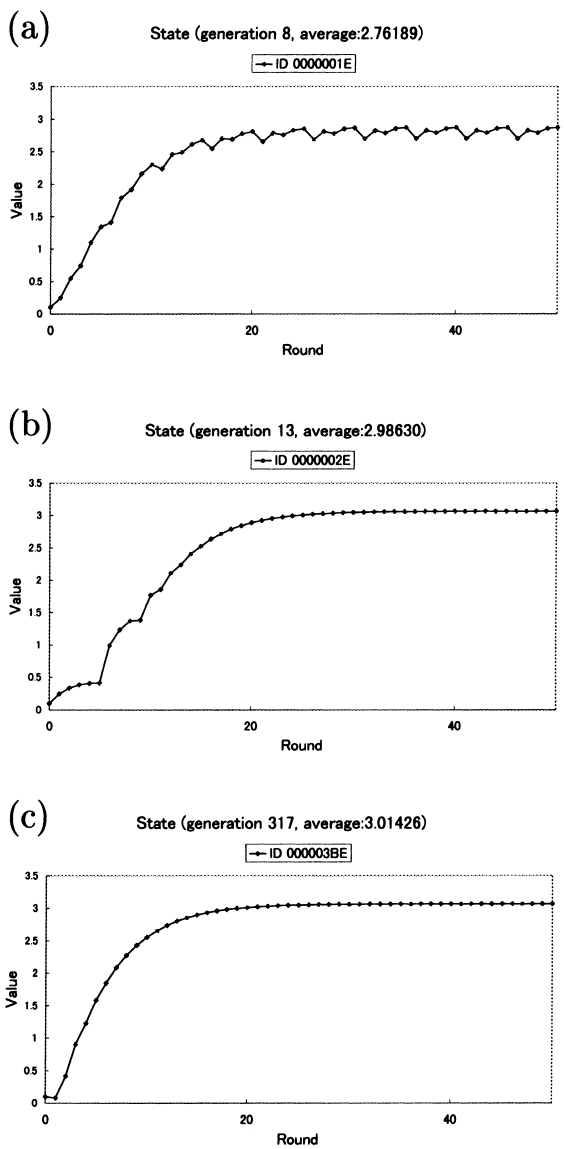

Fig. 24. Evolution with increasing generation number. These are the state charts corresponding to the action and resource charts in Fig. 23.

1. The frequency of cutting tree 2 increases, while that of cutting tree 1 decreases. As a result, the player is able to obtain some profit also from tree 2 (Fig. 23(a)).
2. The player starts managing periodic dynamics for the sizes of trees that bring him the best profit after about the 15th round, during which rounds the frequencies of cutting tree 1 and tree 2 are equal (Fig. 23(b)). He cuts the two trees by turns.
3. The number of the round of which the dynamics first becomes periodic decreases (Fig. 23(c)). Also, as the generation number increases, the round at which the state of the player first reaches its maximal value (approximately 3.05) becomes earlier (Fig. 24(a)–(c)).

The dynamics for each of the two trees (shown in the resource chart (Fig. 23(c))) are almost the same as those observed in the one-tree LD game in this paper (Fig. 9(b)).

References

[1] R.J. Aumann, Agreeing to disagree, Ann. Statist. 4 (1976) 1236–1239.

[2] R.J. Aumann, Rationality and bounded rationality, Nancy L. Schwartz Lecture, J.L. Kellogg School of Management, Northwestern University, 1986.

[3] R. Axelrod, The Evolution of Cooperation, Basic Books, New York, 1984.

[4] D.D. Davis, C.A. Holt, Experimental Economics, Princeton University Press, Princeton, NJ, 1993.

[5] M.M. Flood, Some experimental games, Research Memorandum RM-789, RAND Corporation, Santa Monica, CA, 1952.

[6] D. Fundenberg, J. Tirole, Game Theory, MIT Press, Cambridge, MA, 1992.

[7] G. Hardin, The tragedy of the commons, Science 162 (1968) 1243–1248.

[8] T. Ikegami, M. Taiji, Imitation and cooperation in coupled dynamical recognizers: advances in artificial life, in: D. Floreano, J. Nicoud, F. Mondada (Eds.), Proceedings of the Fifth European Conference, ECAL'99, Lausanne, Switzerland, Springer, Berlin, 1999.

[9] K. Kaneko, Collapse of Tori and Genesis of Chaos in Dissipative Systems, World Scientific Publishers, Singapore, 1986, 264 pp. (based on Thesis of 1983).

[10] J. Maynard Smith, Evolution and the Theory of Games, Cambridge University Press, Cambridge, 1982.

[11] J. Nagumo, S. Sato, On a response characteristic of a mathematical neuron model, Kybernetik 10 (1972) 155–164.

[12] J. von Neumann, O. Morgenstern, Theory of Games and Economic Behavior, Princeton University Press, Princeton, NJ, 1944.
[13] A. Neymann, Bounded complexity justifies cooperation in the finitely repeated prisoner's dilemma, Econ. Lett. 19 (1985) 227–230.
[14] E. Ott, Chaos in dynamical systems, Cambridge University Press, Cambridge, 1993.
[15] A. Rapoport, Mathematical theory of motivation interaction of two individuals, Bull. Math. Biophys. 9 (1947) 17–27.
[16] N. Rashevsky, A problem in mathematical biophysics of interactions of two or more individuals which may be of interest in mathematical sociology, Bull. Math. Biophys. 9 (1947) 9–15.
[17] O.E. Rössler, Adequate locomotion strategies for an abstract organism in an abstract environment: a relational approach to brain function in Physics and Mathematics of the nervous system, in: M. Conrad, W. Guttinger, M. Dalcin (Eds.), Springer Lecture Notes in Biomathematics, Vol. 4, 1974, pp. 342–369.
[18] O.E. Rössler, Chaos in coupled optimizers, Ann. NY Acad. Sci. 504 (1987) 229–240.
[19] O.E. Rössler, Fraiberg–Lenneberg speech, Chaos, Solitons and Fractals, Vol. 4, No. 1, Elsevier, Amsterdam, 1994, pp. 125–131.
[20] A. Rubinstein, Finite automata play the repeated prisoner's dilemma, J. Econ. Theory 39 (1986) 83–96.
[21] H. Simon, Models of Man, Wiley, New York, 1957.
[22] M. Taiji, T. Ikegami, Dynamics of internal models in game players, Physica D 134 (2) (1999) 253–266.
[23] E. Zermelo, Über eine Anwendung der Mengenlehre auf die Theorie des Schachspiels, in: E.W. Hobson, A.E.H. Love (Eds.), Proceedings of the Fifth International Congress of Mathematicians, Vol. 2, Cambridge University Press, Cambridge, 1913, pp. 501–504.
[24] L. White, Medieval Technology and Social Change, Oxford University Press, London, 1962.