The Sixteenth International Symposium on Artificial Life and Robotics 2011 (AROB 16th '11),
B-Con Plaza, Beppu,Oita, Japan, January 27-29, 2011

Modeling and solution for optimization problems with incomplete information
– A general framework and an application to cruising taxi problems –

Makoto OHARA and Hisashi TAMAKI
Kobe University, 1-1 Rokkodai-cho, Nada-ku, Kobe
(Tel: +81-78-803-6102; Fax: +81-78-803-6102)
(ohara@al.cs.kobe-u.ac.jp)

Abstract: The purpose of this study is to construct a general model for optimization problems with incomplete information, and to assist designing solutions. The term "incomplete information" means that information of target system, e.g. the dynamics and the status variables, is not obtained enough to optimize. Before description of proposed model with incomplete information, an optimization model with "complete" information is structured. The occasions with "incompleteness" of information are explained on this optimization model, and approaches to resolving them are described. Then, the optimization model with incomplete information is defined. Moreover, in order to investigate the validity of the proposed framework, it is applied to cab-dispatching (of cruising taxis) problems, where the objective is to distribute a set of cabs efficiently by indicating the adequate location to each taxi driver. Through some computational examples, the effectiveness and the potential of the proposed approach is confirmed.

Keywords: Uncertainty, Optimization Model, Distributing Taxi Problem

1. INTRODUCTION

This research deals with a class of optimization problems with incomplete information, in which the information about the structure as well as the dynamics of the target system is not sufficient. To this class of the problems, in this paper, a general framework for both modeling the problem formally and designing a solution structure systematically is newly proposed. This class includes such types of optimization problems as containing some uncertainty in systems, e.g. due to the spatially partial or temporarily late observability, and/or some unobservability in decision-making.

So far, lots of researches have been presented on several types of optimization problems with uncertainty, from the viewpoints of the description of the problem as well as the design of solutions [1∼3]. However, there have been few studies on general frameworks of both optimization models and their solutions. By introducing informational viewpoints, a variety of uncertain aspects may be dealt with in a uniform way, which is the keynote point of our research.

For confirming validity of the proposal aproach, it is applied distributed cruising taxi problems. In these cab-dispatching problems, there are many unobservable state variables, e.g., the occurrence of passengers and the positions of the cabs of competitive companies. It is difficult or it might be impossible to optimize the dispatching beforehand due to incompleteness. In designing the solution, a set of rules (rule-set) for indicating the priority areas in the information layer, and a genetics-based machine learning (GBML) method is adopted to adjust a rule-set in the supervisor layer. As a result of the computational examples, the effectiveness and the potential of the proposed approach is confirmed.

2. OPTIMIZATION PROBLEMS

2.1 Optimization Model with Complete Information

Optimization problems are defined as problems to maximize/minimize evaluation values about behaviors of systems. Components of the optimization problems are referred as; "Target System" is the target of evaluation for optimization, and is controlled directly, "Relevant System" interacts with Target System, in other words, Relevant System may be possible to be controlled indirectly, "Environment" acts Target System, and can't be conrolled, "Controller" decisions a direction of Target System's behaviors, "Supervisor" affects the decision-making of Controller.

In order to optimize, these components have to convey decisions for others, i.e., "Control": Controller sends operations, which indicate what to do, to Target System, "Supervision": Supervisor sends orders for decision-making to Controller.

For making decisions, the components know about others, i.e., "Observation": The states of Target System, Relevant System and Environment are sensed by Controller, "Evaluation": Controller sends results, which are clued to judge the effectiveness of decision-making, to Supervisor. These components and information-exchanges are collectively called "Optimization Model" in this paper.

To understand easily, Optimization Model is identified as a hierarchical manner; "Physical Layer" shows the physical side of the problems, and includes continuous-time systems, i.e. Target System, Relevant System and Environment, "Information Layer" shows the information side of the problems, and includes short-rate discrete-time system, i.e. Controller, "Supervisor Layer" shows an optimizer for the problems, and includes long-rate discrete-time system, i.e. Supervisor.

Definition of Time Constant: Target System is optimized during $\mathcal{T}$ $(= [0, T))$. Controller evaluates behaviors of Target System every time interval τ^E, and Physical Layer is observed each sampling cycle τ^O. The time horizon $\mathcal{T}$ is discretized to n^E terms by the interval τ^E, and each k-th term $\mathcal{T}^{(k)}$ is discretized to n^O periods by the cycle τ^O. Then, $\mathcal{T}^{(k)}$ corresponds the time interval $\left[\, k\tau^E, (k{+}1)\tau^E \,\right)$, and $m^{(k)}\, th$ period corresponds $\left[\, k\tau^E{+}m\tau,\; k\tau^E{+}(m{+}1)\tau \,\right)$. The set of periods $\mathcal{T}^D$ is defined as below;

$$\mathcal{T}^{O(k)} = \{0, \ldots, n^O\} \quad (n^O = (\tau^E/\tau^O){-}1). \qquad (1)$$

If notation "(k)" isn't be needed, it is omitted in following. The information-exchanges on the time axis are illustrated in Fig. 1.

Elements: The elements of the optimization model (and correspondent elements of Dispatching Cruising Taxi Problems which are described later) are shown in Table 1.

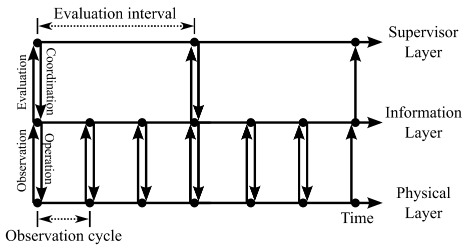

Fig. 1 Information-exchanges between each layer.

Input/Output: Each component's inputs/outputs with complete information is defined as below;

$$a^S(t) = o^R(t), \quad e^S(t) = o^V(t), \quad a^R(t) = o^S(t),$$
$$e^R(t) = o^V(t), \quad u(t) = v\left[\lfloor t/\tau \rfloor\right],$$
$$y[m] = \left(x^S(m\tau), x^R(m\tau), x^V(m\tau)\right),$$
$$p[0^{(k)}] = c[0^{(k)}], \quad r[n^{E(k)}] = Z[n^{E(k)}].$$

Optimization Problems: The objective function and the restriction is formulated as;

$$\min\ Z = F\{y[0], \ldots, y[n^O]\} + P(G)\{y[0], \ldots, y[n^O]\}$$
$$\text{s.t. } v[m] \in H[m] \quad ({}^\forall m \in \mathcal{T}^O)$$

Controller obtains the states of Target System by observations. Then, the state restriction is used in a penalty P.

2.2 Incompleteness

2.2.1 Classification

A variable y is observation quantities of a status variable x. The relation of y and x with complete information is as follow;

$$y[m] = x(m\tau^O). \qquad (2)$$

The classes of incompleteness in the observations are;

(i) Accuracy (error): An observed value includes an error $\sigma(t)$ [%].

$$y[m] = x(m\tau^O) + \sigma(m\tau^O). \qquad (3)$$

(ii) Time delay: A status value of time $\tau(t)$ ago is observed.

$$y[m] = x(m\tau^O - \tau). \qquad (4)$$

(iii) Unobservable: A status value is unobservable.

$$y[m] = *. \qquad (5)$$

The symbol $*$ means uncertain value.
The observations are classified completeness, incompleteness (i), (ii), (iii), or combination of theirs.

2.2.2 Completion

Completion functions are required for each incompleteness;

- for incompleteness (i) : f^C

$$y[m] = f^C(x(m\tau^O) + \sigma(m\tau^O)) = x(m\tau^O), \qquad (6)$$

- for incompleteness (ii) : f^E

$$y[m] = f^E(x(m\tau^O - \tau)) = x[m], \qquad (7)$$

- for incompleteness (iii) : f^P

$$y[m] = f^P(*) = x[m]. \qquad (8)$$

The functions f^C, f^E and f^P each are called "Correction", "Estimation" and "Prediction".

In order to implement completion of incompleteness, information as follows are needed. "History": an history about a status variable to observe is effective for Correction and Prediction. "Same kind of a status variable": if a system is consists of subsystems, e.g. multi-agent, status variables of each subsystem, which are same kind of a state to observe, help Estimation. "Other kind of a status variable": if a status variable can't be observed, other state values of a same system might be valid for Estimation. "Knowledge": knowledge about a system or dynamics assist all completions.

2.3 Optimization Model with Incomplete Information

Optimization Model with incomplete information has completion functions defined in Section 2.2.2. Completion Module converts observed values, and sends complementary values to Controller. An overview of Optimization Model with incomplete information and an inside of Completion Module are illustrated in Fig. 2.

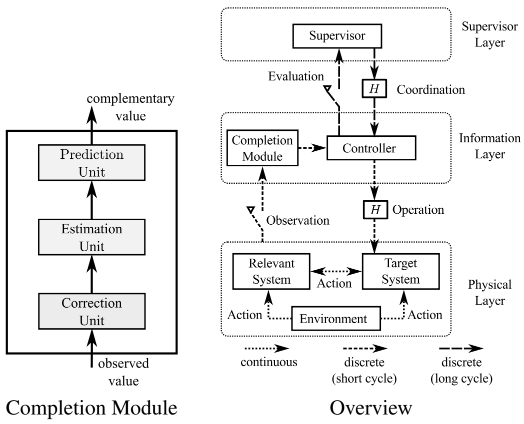

Fig. 2 Hierarchical model of optimization problem with incomplete information

3. DISPATCHING CRUISING TAXI PROBLEMS

For confirming validity of the proposal optimization model, the model applies Dispatching Cruising Taxi Problem.

3.1 Outline

Since the revision of Road Transportation Law in Japan, the taxi business is becoming saturated by new entries. In particular, rookie drivers have a lot of trouble finding passengers,

Table 1 Elements of optimization problems and corresponding with Dispatching Cruising Taxi Problems

Layer	Component/System†	Symbol‡	Element†	
Physical	Target System (Cooperative Company)	$\bullet\ x^S$	state	(position, assigned passenger and number of assignment)
		$+\ u$	control input	(dispatching area)
		$+\ a^S$	relevant action	(position, destination and assigned cab)
		$+\ e^S$	environment action	(traffic jam)
		$\star\ T^S$	transition function	(move rule)
		$-\ o^S$	output action	(position and assigned passenger)
		$\star\ U^S$	output function	(take rule)
	Relevant System (1) (Passengers)	$\bullet\ x^R$	state	(position, destination, appearance time and assigned cab)
		$+\ a^R$	relevant action	(position and assigned passenger)
		$+\ e^R$	environment action	(weather)
		$\star\ T^R$	transition function	(appear rule)
		$-\ o^R$	output action	(position, destination and assigned cab)
		$\star\ U^R$	output function	(ride rule)
	Relevant System (2) (Competitive Company)	$\bullet\ x^R$	state	(position and assigned passenger)
		$+\ a^R$	relevant action	([Not Available])
		$+\ e^R$	environment action	(traffic jam)
		$\star\ T^R$	transition function	(move rule)
		$-\ o^R$	output action	(position and assigned passenger)
		$\star\ U^R$	output function	(take rule)
	Environment (Environment)	$\bullet\ x^V$	state	(states about weather and traffic jam)
		$\star\ T^V$	transition function	(weather and traffic translation)
		$-\ o^V$	output action	(weather and traffic jam)
		$\star\ U^V$	output function	(weather and traffic action rule)
Information	Controller (Dispatch Controller)	$+\ y$	observed value	(cabs states, passengers actions and environment actions)
		$\bullet\ \mathcal{M}$	observed record	(observed record)
		$-\ v$	operation	(dispatching area)
		$+\ p$	operation parameter	(weight coefficients)
		$\star\ G$	state restriction	([Not Available])
		$\star\ H$	operation restriction	(area permitted business)
		$\star\ C$	operation function	(priority calculation)
		$-\ Z$	evaluated value	(total of assignment)
		$\star\ F$	evaluation function	(sum of assignment)
Supervisor	Supervisor (Supervisor)	$+\ r$	result	(total of assignment)
		$\bullet\ \mathcal{R}$	result record	(assignment record)
		$-\ c$	order	(weight coefficients)
		$\star\ o$	order function	(GBML)
		$\bullet\ \mathcal{O}$	order record	(weight coefficient record)

† Words in parentheses are corresponding components or elements of Dispatching Cruising Taxi Problems.
‡ Header symbols mean $+$:input, $-$:output, $\bullet$:state, and $\star$:other, e.g. dynamics.

because taxi business has relied on individual experience of drivers. The taxi companies need systems which assist less-experienced drivers for finding passengers. There have been studies based on statistical models [4), 5], in contrast, few studies have been carried out to analyze Dispatching Cruising Taxi Problem agent-models.

3.2 Approach

Positions of passengers, targets of taxis, are unobservable in these problems. Thus, taxis are dispatched regions. This paper suggests an optimization framework for taxi dispatching by the rule-set which calculates target areas with observable state variables. The corresponds of the proposed model and Dispatching Cruising Taxi Problems have been described in Table 1.

Definition of Space: A region $\mathcal{A}$ is intended in a problem, and a cooperative company is permitted business in a region $\mathcal{A}^B$. The region $\mathcal{A}$ is divided to regions $\mathcal{A}_a$ $(a = 1, \ldots, n^A)$, and each cab of the cooperative company is distributed to any region $\mathcal{A}_a$. The region $\mathcal{A}_a$ is called "Area a" below.

Priority Rule: A priority rule calculates priorities of each area for every cab. Each cab is dispatched to the area which is highest priority for the cab. The rule calculates priority which is a weighted summation of preparing equations.

Learning Method: In this paper, condition is defined as partially feature quantity space of whole areas, and action is defined as weight coefficients in the priority rule [6].

Elements: The elements of dispatched cruising taxi problems have been shown in Table 1.

4. COMPUTATIONAL EXAMPLE

Simulations which learn the rules and evaluate them are performed to verify the validity of proposed approach for distribution cruising taxi problems. The Rules learned by GBML are compared with a heuristic rule by evaluated values.

4.1 Setting

Simulator: The cooperative company's cabs go to its own dispatching area by the shortest distance, and move randomly in the dispatching areas after arrival. The competitive company uses a proportion rule, this rule dispatches proportional number cabs to the ratio of estimated number of passengers in each area to whole areas. The passengers appear only in permitted business region $\mathcal{A}^B$. The number of emerging passengers in whole areas is 3 per a time unit on average. The number of the areas is 100, and the number of the permitted business areas is 64.

Priority Rule: The priorities are calculated by weighted equations with properties as below. The areas have more priority, in which more passengers appear probably and more nearly. The areas have more priority, in which less cabs of the cooperative company, The areas have more priority, in which more passengers appear probably, The prior areas are decided at random. The weight of equations have been normalized in refer to results of preliminary experiment.

GBML: The number of individuals is 30, and the number of generations is 100. Each individual is simulated 2 times in each condition (I) $\sim$ (III), about random variables for passengers, weather and traffic jam, and the average of evaluated value are treated as a fitness of the individual. The conditions of passenger appearance are combination of 4 patterns. The patterns are illustrated in Fig. 3, and the combinations are (I): pattern 1 in the noon and pattern 2 in the night, (II): 3 and 3, (iii) 4 and 4 which are made by different seeds. GBML searches the weight coefficients from 0, ± 0.25, ± 0.5, ± 0.75 or ± 1. The feature quantity space is composed of variables defined as, deviation of the number of cabs in each area, estimated number of appearing passengers in whole areas, ratio of cabs carrying no passengers, time slot. The space is divided by small or large of each variable.

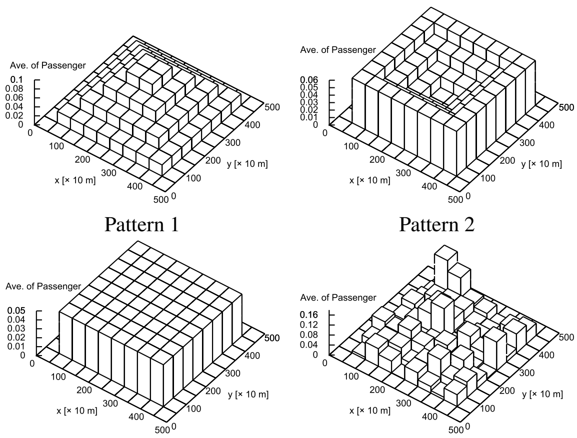

Fig. 3 Examples of appearance patterns of the passengers

Example Problem

The simulations for learning are performed in 2 cases; (a) the cooperative company has 10 cabs, and the competitive company has 20 cabs, (b) the cooperative company has 80 cabs, and the competitive company has 160 cabs, and each case are learned 3 trials. Moreover, the best solutions of each trail and the proportion rule are evaluated 4 times in each condition (I) $\sim$ (III) and all conditions.

4.2 Result

The learning process of the best solution in every trial are plotted in Fig. 4. The result for evaluation the acquired rule and the proportion rule is plotted in Fig. 5.

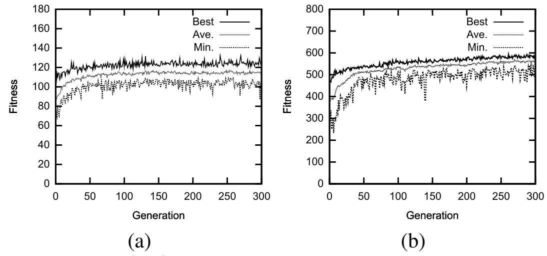

Fig. 4 Learning processes of GBML

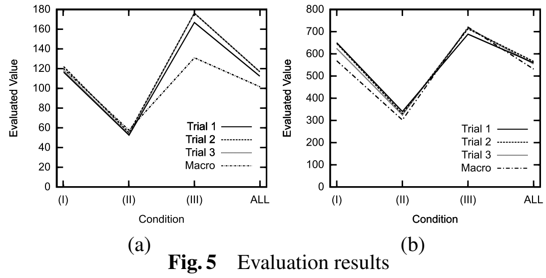

Fig. 5 Evaluation results

From the figures, the acquired rules are valid as well as the proportion rule in all conditions, and better than substantially in particular situations.

5. CONCLUSION

The optimization problems and incompleteness on observation are described. By using them, the optimization model with incomplete information is defined. In addition, in order to confirm utility of the proposed model, it applies Dispatching Cruising Taxi Problems. As the result, the proposed model has potential to express the taxi problems, and to acquire the good solutions. The following are left for future study: to improve the optimization model focused on incompleteness of information, and to apply the proposed model to other problems for future works.

REFERENCES

[1] Ian Jordaan: Decisions under Uncertainty, Cambridge University Press (2005).

[2] Zhiang Lin: Environmental determination or organizational design: An exploration of organizational decision making under environmental uncertainty, Simulation Modeling Practice and Theory, 14, 438–453 (2006).

[3] Enrique Alba, Christian Blum, Pedro Asasi et al.: Optimization Techniques for Solving Complex Problems, Wiley (2009).

[4] Wong K.I., Wong S.C. and Yang H.: Modeling urban taxi services in congested road networks with elastic demand, Transportaion Research Part B, 35, 819–842 (2001).

[5] Yu Jianxin, Zhou Xiaomin and Zhao Hongyu: Design and Implementation of Taxi Calling and Dispatching System based on GPS Mobile Phone A Research for LBS Application Teaching Case, Proceedings of 2009 4th International Conference on Computer Science & Education, 1163–1169 (2009).

[6] Sakakibara K., Tamaki H., Murao H. et al.: Genetics-Based Machine Learning Approach for Flexible Shop Scheduling Problems : Rule Acquisition for Real-time Scheduling (in Japanese), Journal of Information Processing Society of Japan, 45, 2, 91–98 (2004).