情報処理学会研究報告
IPSJ SIG Technical Report

Vol.2009-DBS-148 No.11
Vol.2009-FI-95 No.11
2009/7/28

データベースによるグラフとオートマトンのモデル化

宮永照二[†1,†2]

リレーショナルデータベースの提供する表の構造は、頂点とエッジからなるグラフの構造をシンプルにモデル化する。最近のデータベースシステムは、データの保存や抽出だけでなく、いくつかの組込み関数により、極めて多彩なタスクをこなすことが可能である。

このようなリレーショナルデータベースではグラフ構造をモデル化し、パスやサイクルなどの特徴的な構造を抽出でき、さらにオートマトンをモデル化し、クエリーによりオートマトンの遷移を実行できる。

本稿では、オートマトンにより表現される計算モデルの構成法を説明し、その応用例を示す。

Modeling of Graph and Automaton in Database

Shoji Miyanaga[†1,†2]

Table scheme that relational database provides can model the structure of graph which consists of vertices and edges. Recent database system can not only save or extract data but also process external diverse tasks. Such database system can model the structure of graph and extract the characteristic structures of model such like path or cycle, and furthermore can model the automaton and execute the transition of the automaton by calling query. In this paper, author explain the computation model constructions which are expressed by automaton and show their application.

1. Introduction

Currently, variously formed data are, which are used in both organization and personal, saved and reused. Complex structured data also can be saved into XML database, and their usage are expanded increasingly. On the other hand, attentioning to database specification, increase of disk spaces and memory capability and performance of CPU, LAN and CACHE enable system specification very high in both time and space. Furthermore, for database system's various embedded functions are provided so that its data customization is enabled by query execution.

By the way, it is known that various data and structure can be represented by graph. Table structure that relational database provides can model graph structure consisting of vertices and edges. Recent database systems can not only save and extract data but also process extreme diverse tasks by some embedded functions. Modeling graph structure, characteristic structure such like paths and cycles can be extracted, furthermore, can model automaton and execute transition by query, and equipped property as program execution platform.

In this paper, we explains system construction represented by automaton in relational database, and show its application. Then this paper is organized as follows. In chapter 2, we explains how to model graph and automaton in relational database. In chapter 3, some application are presented. In chapter 4, we discussed by our proposal in this paper. In chapter 5, related work is explained.

2. Modeling of Graph and Automaton

2.1 Digraph

Digraph is formalized like below.

$G = < V, E >$

Herein, V is a set of vertices, E is a set of directed edge. Now, we define the vertices and edges of graph G like below.

$V = (1, 2, 3, 4)$

$E = (< 1, 2 >, < 2, 3 >, < 3, 4 >, < 4, 1 >)$

Herein, this graph can be expressed in relational database by tables described below.

[†1] 大阪大学
Osaka University
[†2] 産総研
National Institute of Advanced Industrial Science and Technology

表 1 Vertex Table

vertexId	vertexLabel
1	v1
2	v2
3	v3
4	v4

表 2 Arc Table

arcId	startVertex	endVertex	arcLabel
1	1	2	e1
2	2	3	e2
3	3	4	e3
4	4	1	e4

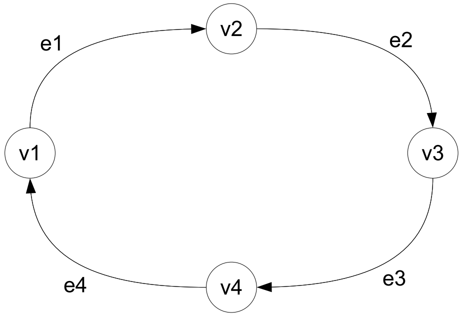

図 1 Digraph

2.2 Deterministic Finite Automaton (DFA))

Formalized definition of Deterministic Finite Automaton (DFA)) is described below.

$DFA\ A =< Q, \Sigma, \delta, q_0, F >$

Herein, Q is a set of states of automaton, Σ is a set of alphabets, δ is transition function, q_0 is initial states, F is a set of accept states. δ is defined like below.

$\delta : Q \times \Sigma \to Q$

δ is a function from a pair of current state and input alphabet to a state to be transited. An Example of table specification, which define this automaton, for transitions is described below.

表 3 State Table of Automaton

state	stateMean	stateLabel
0	init	q_0
1	normal	q_1
2	normal	q_2
3	normal	q_3
4	accept	q_4

表 4 State Transition of DFA

transId	startState	endState	alphabet
0	0	1	a
1	1	2	a
2	2	1	a—b
3	1	3	b
4	3	2	a
5	3	4	b

In this case, deterministic automaton is expressed like following state diagram.

2.3 Execution of Transition of DFA

Management information such like current state, semantics of current state, and input alphabet array are need to execute transition of deterministic automaton. Therefore, we suppose current state like below.

Now if we execute transition of automaton, whose id=1, on reading alphabet SQL like below is executed.

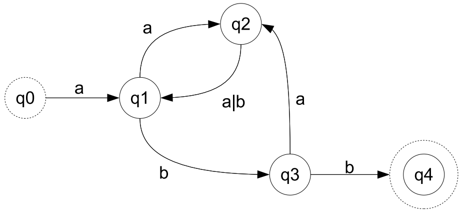

図 2 Example of DFA

表 5 State Management of Automaton

id	state	stateMean	alphabets
1	q_0	*init*	*aaabb*
2	q_1	*normal*	*abaabb*

```
select T1.dfaId as dfaId,
       T2.endState as state,
       SUBSTRING(T1.alphabet, 2, len(T1.alphabet)−1)
         as alphabet
from dbo.T_DFA as T1
     inner join dbo.T_DFA_Transition as T2
on T1.state=T2.startState
and T2.alphabet=SUBSTRING(T1.alphabet, 1, 1);
```

This query is one example which is described in Microsoft SQL Server 2008[7]. There are some way to rewrite the automaton information by using the outputs of above query, and one of outputs is to execute insert statement to add new entry like below.

```
insert into T_DFA
select T1.dfaId as dfaId,
       T2.endState as state,
       substring(T1.alphabet, 2, len(T1.alphabet)−1)
         as alphabet
from dbo.T_DFA as T1
     inner join dbo.T_DFA_Transition as T2
on T1.state=T2.startState
```

Transition execution is enabled to substitute original record by the record above query generates, therefore it is found that relational database can control automaton.

2.4 Extension to NFA

Non-deterministic Finite Automaton (NFA) is automaton whose transition state may not be determined uniquely by current state and input alphabet. An Example of such NFA is described below.

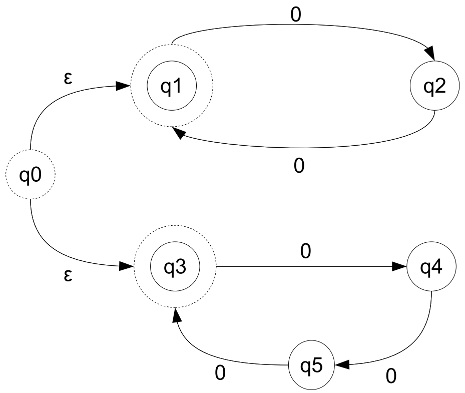

図 3 Non-deterministic Finite Automaton

In this case in state q_0 same character 1 given, it is not determined uniquely whether q_1 is next state or q_2 is next state. For automaton whose state to be transited, herein we show the transitive state randomly.

```
select T1.dfaId as dfaId,
       T2.endState as state,
       SUBSTRING(T1.alphabet, 2, len(T1.alphabet)−1)
         as alphabet
       RANDOM() as seq
from dbo.T_DFA as T1
```

```
      inner join dbo.T_DFA_Transition as T2
  on T1.state=T2.startState
  and T2.alphabet=SUBSTRING(T1.alphabet, 1, 1)
  order by seq
```

Herein RANDOM() is random number generator function. A set of transition states are rearranged by random numbers. we can express the non-decidability by picking one entry which is minimal (or maximal) from this set.

2.5 Extension to Pushdown Automaton

Pushdown Automaton (PDA) provides memory equipment by stack like below.

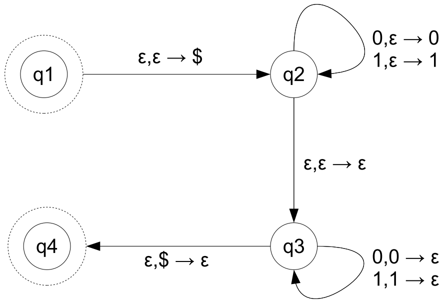

図 4 Pushdown Automaton

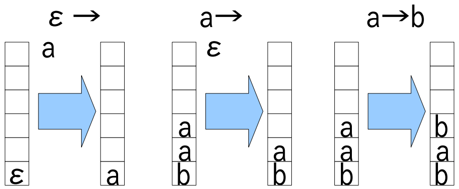

図 5 Stack Dealing

Formalized definition of PDA is like below.

$PDA\ A =< Q, \Sigma, \Gamma, \delta, q_0, F >$

Herein, Q is a set of states of PDA, Σ is a set of alphabet, Γ is a set of tape alphabets, q_0 is an initial state, and F is a set of accept states.

For extension to PDA, both state table and transition table should be changed because stack is rewriten.

表 6 State Mangement PDA

id	state	stateMean	alphabets	stack
1	q_0	*init*	*aaabb*	ϵ
2	q_1	*normal*	*abaabb*	ϵ

表 7 Transition Function of PDA

transId	startState	endState	alphabet	rtape	wtape
0	0	1	a	ϵ	ϵ
1	1	2	a	ϵ	a
2	2	1	a—b	a	ϵ
3	1	3	b	ϵ	b
4	3	2	a	b	a
5	3	4	b	b	b

Herein, letting T_DFA_Transition transition table of PDA, transited state of PDA is obtained by following query.

```
select T1.dfaId as dfaId,
       T2.endState as state,
       SUBSTRING(T1.alphabet, 2, len(T1.alphabet)-1)
         as alphabet
       T2.wtape + substring(T1.stack, 2, len(T1.stack)-1)
from dbo.T_DFA as T1 inner join dbo.T_DFA_Transition as T2
on T1.state=T2.startState
   and T2.alphabet=SUBSTRING(T1.alphabet, 1, 1)
   and T1.rtape=SUBSTRING(T1.stack, 1, 1)
```

Like this, by execution some queries, dealing with automaton with stack rewriting is enabled. By the way stack rewriting of PDA is performed like below.

2.6 Extension to Cellular Automaton

Cellular Automaton can express the system which consists of some automata which are deployed on grid like below. On the other hand, assigning indivisual automaton to each game character, such game can be implemented. Cellular Automaton Methods are widely used, define the structure which consists of many factors such like traffic simulator and environment simulator, and analyze phenomena.

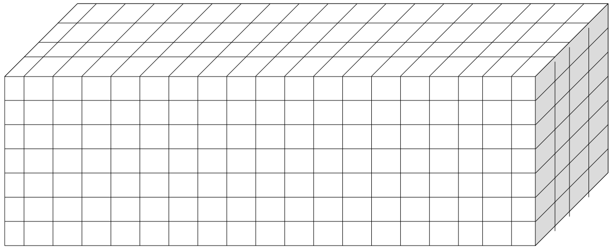

図 6 Cellular Automaton on Grid

As Explained in previous section, one DFA corresponds to distinct unique record. Defining some records corresponding to distinct automaton can execute some automata concurrently. Furthermore, on characteristic property, CA can process several different functions case by case. Precedingly described automaton can only deal with state transition, so the system in which some functions each of which corresponds to a function is hard to be modeled. However, in the database system in which some functions can be executed by evaluation of the formula representing the function, and in the programming environment Cellular Automaton can easily define the its activities corresponding to each state of CA. Now, we model the activities of Cellular Automaton like below.

In this diagram, CA is formalized like below.

$PDA\ A =< Q, V, \Sigma, \delta, q_0, F, g >$

Herein, Q is a set of states of CA, V is a set of cell variables, Σ is a set of input alphabets. q_0 is an initial state, $F = \{f\ ; f : V \rightarrow V\}$ is a set of function corresponding to each state, and $g : V \rightarrow Q$ is a function to determine the state to be transited.

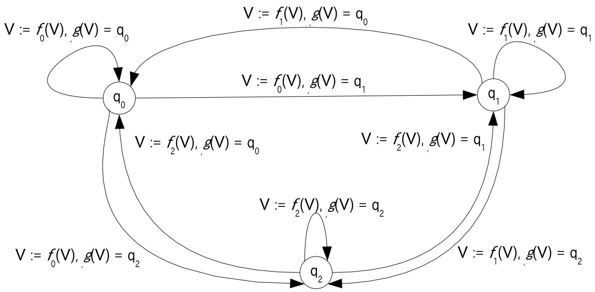

図 7 Transition of CA

表 8 Transition Function of CA

id	state	alphabet	func	transFunc
1	q_0	1	f_1	g
1	q_0	2	f_2	g

This record is interpreted like following. If DFA '1' is in state q_0, then, if alphabet '1' read, function f_1 is executed and current variables are updated and next state is determined by function g. Like this, activity rules of cells are provided by tables and can be described.

3. Application Examples

3.1 Web Application

In distributed environment, as centerized data management application example, Web Application is often picked-up. Web application consists of page management block, page display block and page transition mangement block. Page management displays pages to user and manages page contents, and save temporary pages untill commitment requested or timeout notified. Page display block customizes contents to tagged text which is enabled to be displayed in Web Browser. Page transition management block manages page states, and switches procedures by request from browser.

As Implementation example, there is JSP and Servlet which manage consistent object, such that object is general. In this case also by hold preceding database on background, it is enable to manage easily.

Now, consider the facility reservation system like following list.

(1) q_0:Login
(2) q_1:Facility List
(3) q_2:Facility Detail
(4) q_3:Reservation
(5) q_4:Confirmation
(6) q_5:Cancel

Screen transition of this system is expressed by state diagram.

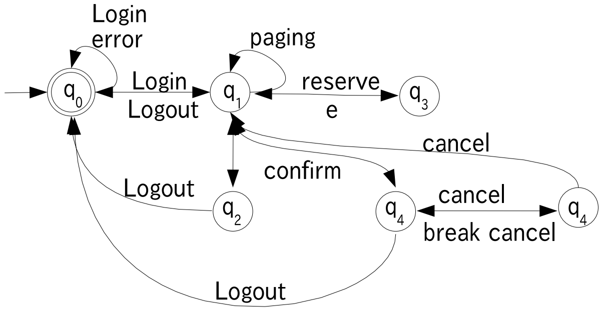

図 8 State Diagram of Facility Reservation

It is enable to assume input letters as event, and to describe and to manage state transition such that process is change in order to event. In this case, client must has only DFA ID as management information, so it is enable to manage easily.

3.2 Physical Phenomena represented by Evolutional Equation

Evolutional equation is used to analyze air polution and earch polution. physical phenomena of such system is enable to be analyzed by CA. By describing preceding function corresponding to each state of CA, it is enable to manage large systems.

4. Discussion

In this paper, we focus table structure of relational database and showed that it is enabled to define the digraph structure by table and to model classes of automata described below.

(1) Deterministic Automaton
(2) Non-deterministic Automaton
(3) Pushdown Automaton
(4) Cellular Automaton

Furthermore, it is found that by query execution and using database property as application execution environment it is enable to control above automata.

By this, it is confirmed that defining descrete system in relational database, it is enable to execute automaton.

By using databse system in which evaluation of equation (of function) is enabled to be describe by query, it is enable to describe and execute the complex system such that it does various actions case by case like agent.

5. Related Work

There are many research about management and analysis of database and management of graph structure in database. There is research by Ishino and Takeda[?] about automaton in database.

6. Acknowledgement

We appriciate to Prof. Koji Torii and Prof. Masatsugu Kidode of Nara Advanced Institute of Science and Technology.

参 考 文 献

1) M Sipser: *Introduction to the Theory of Computation*, Thomson Course Technology, ISBN978-0-534-95097-2, (2006).
2) John E H, R Motowani, J D Wolfman, A Nozaki: Automaton, *Linguistic Theory, Computation Theory 2 (2nd ed)*, Science, (2006).
3) M Huth, M Ryan: *Logic in Computer Science: Modelling and Reasoning About Systems*, Cambridge, (2004).
4) R Diestel: *Graph Theory 3rd ed*, Springer, ISBN978-3-540-26183-4, (2004).
5) *Microsoft SQL Server 2008 Express*, download enable from http://www.microsoft.com/.
6) A Ishino, M Takeda: *A Proposal for XQuery Processor with Deterministic Automaton and Path Pruning*, The Database Society Japan Letters, 4(4) ,p17-20, (2006).