# Extensions of Dynamic Information Systems in State Prediction Problems: the First Study

**Krzysztof Pancerz**
University of Information Technology and Management in Rzeszów, Poland
College of Management and Public Administration in Zamość, Poland
kpancerz@wsiz.rzeszow.pl

## Abstract

The paper is devoted to the application of extensions of dynamic information systems to state prediction problems. A dynamic information system can describe states of processes observed in a given system and transitions among them. If we extend a given dynamic information system by adding some new transitions among states which have not been observed yet, then we are interested in degrees of consistency of added transitions with the knowledge about transitions included in the original dynamic information system. Such information can be helpful in predicting possibility of appearing in the future transitions among states in the examined system. In the paper, we propose some approach how to compute a possibility factor of appearing new transitions among states.

**Keywords:** Dynamic information systems, Prediction.

## 1 Introduction

Information systems can be used to represent the knowledge of the behavior of concurrent systems [4]. In this approach, an information system represented by a data table includes the knowledge of the global states of a given concurrent system $CS$. The columns of the table are labeled with names of attributes (treated as processes of $CS$). Each row labeled with an object (treated as a global state of $CS$) includes a record of attribute values (treated as local states of processes). In a general case, a concurrent system is a system consisting of some processes, whose local states can coexist together and they are partly independent. For example, we can treat systems consisting of economic processes, financial processes, biological processes, genetic processes, meteorological processes, etc. as concurrent systems. Dynamic information systems were proposed by Z. Suraj in 1998 [6] to represent the knowledge of states of concurrent systems and transitions between them. Transitions between states were described by binary transition relations. In this paper, we extend a notion of dynamic information systems to the so-called multistage dynamic information systems. These systems enable us to represent multistage transitions among states (observed sequences of states called also episodes). Therefore, transitions among states are described by polyadic transition relations. To represent such relations we propose multistage decision transition systems. We are especially interested in extensions (consistent and partially consistent) of multistage decision transition systems. A partially consistent extension of a given multistage decision transition system consists of new transitions among states which are totally consistent or consistent only to a certain degree (partially consistent) with the knowledge included in the original multistage decision transition system. The degree of con-

L. Magdalena, M. Ojeda-Aciego, J.L. Verdegay (eds): Proceedings of IPMU'08, pp. 101–108
Torremolinos (Málaga), June 22–27, 2008

sistency can be between 0 and 1, 0 for the total inconsistency and 1 for the total consistency. We assume that the knowledge included in multistage decision transition systems is expressed by transition rules, which are minimal decision rules understood from the rough set point of view. We use the helpful theorem given in [1] in order to compute a degree of consistency of a given episode from any extension of a given multistage decision transition system. This theorem enables us to determine which transitions (episodes) in the original multistage decision transition system generate transition rules which are not satisfied by the tested episode from the extension. It is worth noting, that if we use that theorem, then we do not calculate any transition rules in a multistage decision transition system. This is an important property from the computational complexity point of view, especially, if we have high dimensional data (for example, in genetics).

The rest of the paper is organized as follows. In Section 2, a brief review of the basic concepts concerning information systems is given. In Section 3, basic definitions concerning multistage dynamic information systems are presented. Section 4 presents how to compute a degree of consistency of a given episode from any extension of a given multistage decision transition system. Section 5 gives an illustrative example. Finally, Section 6 consists of some conclusions.

## 2 Information System Rudiments

First, we recall the basic concepts of rough set theory concerning information systems (cf. [3], [5], [2]).

An *information system* is a pair $S = (U, A)$, where $U$ is a set of *objects*, $A$ is a set of *attributes*, i.e., $a : U \to V_a$ for $a \in A$, where $V_a$ is called a value set of $a$. A *decision system* is a pair $DS = (U, A)$, where $A = C \cup D$, $C \cap D = \emptyset$, and $C$ is a set of *condition attributes*, $D$ is a set of *decision attributes*. Any information (decision) system can be represented as a data table, whose columns are labeled with attributes, rows are labeled with objects, and entries of the table are attribute values.

We associate a formal language $L(S)$ with every information system $S = (U, A)$. Formulas of $L(S)$ are built from atomic formulas in the form $(a, v)$, where $a \in A$ and $v \in V_a$, by means of propositional connectives: negation ($\neg$), disjunction ($\vee$), conjunction ($\wedge$), implication ($\Rightarrow$) and equivalence ($\Leftrightarrow$) in the standard way. The fact that object $u \in U$ satisfies formula $\phi$ of $L(S)$ (see [3]) will be denoted by $u \models \phi$. If $\phi$ is a formula of $L(S)$, then the set $|\phi|_S = \{u \in U : u \models \phi\}$ is called the meaning of formula $\phi$ in $S$.

Let $S = (U, A)$ be an information system and $B \subseteq A$. The formula of $L(S)$ containing only atomic formulas in the form $(a, v)$, where $a \in B$, $v \in V_a$, will be denoted by $\phi|_B$.

A rule in the information system $S$ is a formula of the form $\phi \Rightarrow \psi$, where $\phi$ and $\psi$ are referred to as the predecessor and the successor of a rule, respectively. The rule $\phi \Rightarrow \psi$ is true in $S$ if $|\phi|_S \subseteq |\psi|_S$. In our approach, we consider rules in the form $\phi \Rightarrow \psi$, where $\phi$ is a conjunction of atomic formulas of $L(S)$ and $\psi$ is an atomic formula of $L(S)$. A rule is called minimal in $S$ if and only if removing any atomic formula from $\phi$ results in the rule being not true in $S$. The set of all minimal rules true in $S$ will be denoted by $Rul(S)$. We assume that $Rul(S)$ includes only such minimal rules true in $S$ which are also realizable in $S$, i.e., $|\phi \wedge \psi|_S \neq \emptyset$ for each $(\phi \Rightarrow \psi) \in Rul(S)$.

Let $DS = (U, C \cup D)$ be a decision system. A *decision rule* in $S$ is a formula of $L(DS)$ in the form $\phi|_C \Rightarrow \psi|_D$. $\phi|_C$ and $\psi|_D$ are referred to as *condition* and *decision* parts of the rule, respectively. In our approach, we consider decision rules such that $\phi$ is a conjunction of atomic formulas of $L(DS)$ and $\psi$ is an atomic formula of $L(DS)$. Each decision rule specifies a decision that should be taken when conditions pointed out by condition attributes are satisfied.

In the approach presented in this paper, we assume that the knowledge included in a given information system $S$ (or a decision system $DS$) is expressed by means of all minimal (decision) rules, true and realizable in $S$ (or $DS$).

A crucial notion in the presented approach is an extension of an information system.

Let $S = (U, A)$ be an information system. An information system $S^* = (U^*, A^*)$ is called an *extension* of $S$ if and only if the following conditions are satisfied: (1) $U \subseteq U^*$, (2) $card(A) = card(A^*)$, (3) $\underset{a \in A}{\forall} \underset{a^* \in A^*}{\exists} V_{a^*} = V_a$ and $a^*(u) = a(u)$ for all $u \in U$.

Each extension $S^*$ of a given information system $S$ includes the same number of attributes and only such objects whose attribute values appeared in the original table representing $S$. Moreover, the data table representing $S$ is a part of the data table representing $S^*$, i.e., all objects which appear in $S$, also appear in $S^*$. According to requirements (2) and (3), for the rest of the paper, the sets $A$ and $A^*$ will be marked with the same letter $A$.

Analogously, we define an extension of a decision information system $DS$.

## 3 Multistage Dynamic Information Systems (MDISs)

In general, a description of concurrent systems by means of information systems does not cover their dynamic behavior, i.e., an information system includes only the knowledge of global states observed in a given concurrent system. In [6], dynamic information systems have been proposed for a description of concurrent systems. A dynamic information system additionally includes information about transitions between global states observed in a given concurrent system. So, the dynamics is expressed by a transition relation defined in a dynamic information system and the term of a dynamic information system should be understood in this sense. Here, we give some crucial notions concerning dynamic information systems.

**Definition 3.1** *A transition system is a pair $TS = (U, T)$, where $U$ is a nonempty set of states and $T \subseteq U \times U$ is a transition relation.*

It is easy to see that in Definition 3.1, a transition relation is a binary relation over the set $U$ of states.

**Definition 3.2** *A dynamic information system is a tuple $DIS = (U, A, T)$, where $S = (U, A)$ is an information system called the underlying system of $DIS$ and $TS = (U, T)$ is a transition system.*

The underlying system includes global states of a given concurrent system whereas a transition system describes transitions between these global states.

Now, we extend a notion of dynamic information systems to the so-called multistage dynamic information systems (in short, MDISs). If we are interested in sequences of changes of global states, then we should represent such changes by means of polyadic relations over the sets of global states. Therefore, we propose to use the polyadic transition relation in the definition of a dynamic information system. Appropriate definitions are given below.

**Definition 3.3** *A multistage transition system is a pair $MTS = (U, T)$, where $U$ is a nonempty set of states and $T \subseteq U^k$ is a polyadic transition relation, where $k > 2$.*

**Definition 3.4** *A multistage dynamic information system is a tuple $MDIS = (U, A, T)$, where $S = (U, A)$ is an information system called the underlying system of $MDIS$ and $MTS = (U, T)$ is a multistage transition system.*

Each element of a multistage transition relation $T$ in a multistage dynamic information system $MDIS = (U, A, T)$ is a sequence of global states (from the set $U$) which can be referred to as an episode.

**Definition 3.5** *Let $MDIS = (U, A, T)$ be a multistage dynamic information system, where $T \subseteq U^k$. Each element $(u^1, u^2, \ldots, u^k) \in T$, where $u^1, u^2, \ldots, u^k \in U$, is called an episode in $MDIS$.*

A dynamic information system can be presented by means of data tables representing information systems in the Pawlak's sense (see [2]). In this case, each dynamic information system $DIS$ is depicted by means of two data tables. The first data table represents an underlying system $S$ of $DIS$ that is, in fact, an information system. The second one represents a decision system that is further referred to as a decision transition system. This table represents transitions determined by a transition relation. Analogously, we can use a suitable data table to represent a multistage transition system. Such a table will represent the so-called multistage decision transition system.

**Definition 3.6** *Let $MTS = (U, T)$ be a multistage transition system. A multistage decision transition system is a pair $MDTS = (U_T, A^1 \cup A^2 \cup \ldots \cup A^k)$, where each $t \in U_T$ corresponds exactly to one element of the polyadic transition relation $T$ whereas attributes from the set $A^k$ determine global states of the $k$-th domain of $T$.*

Each object in a multistage decision transition system represents one episode in a given multistage dynamic information system.

If $k$ is fixed, we can talk about a $k$-adic transition relation, a $k$-stage transition system and a $k$-stage dynamic information system.

For a given multistage decision transition system, we can consider its elementary decision transition subsystems defined as follows.

**Definition 3.7** *An elementary decision transition subsystem of a multistage decision transition system $MDTS = (U_T, A^1 \cup A^2 \cup \ldots \cup A^k)$ is a decision transition system $DTS(i, i+1) = (U_T, A^i \cup A^{i+1})$, where: $i \in \{1, ..., k-1\}$.*

In an elementary decision transition subsystem, we can consider some rules called, in short, elementary transition rules.

**Definition 3.8** *Let $DTS(i, i+1) = (U_T, A^i \cup A^{i+1})$ be an elementary decision transition subsystem. An elementary transition rule in $DTS(i, i+1)$ is a formula of a formal language $L(DTS(i, i+1))$ in the form $\phi|_{A^i} \Rightarrow \psi|_{A^{i+1}}$, where $\phi|_{A^i}$ and $\psi|_{A^{i+1}}$ are formulas of $L(DTS(i, i+1))$ restricted to the sets of attributes $A^i$ and $A^{i+1}$, respectively.*

It is easy to see that elementary transition rules are, in fact, decision rules. In our approach, we will be interested in elementary transition rules in the form of $\phi|_{A^i} \Rightarrow \psi|_{A^{i+1}}$, where $\phi|_{A^i}$ is a conjunction of atomic formulas of $L(DTS(i, i+1))$ and $\psi|_{A^{i+1}}$ is an atomic formula of $L(DTS(i, i+1))$. Moreover, the rules considered will be minimal, true and realizable in a given elementary decision transition subsystem. $\phi|_{A^i}$ and $\psi|_{A^{i+1}}$ are referred to as *condition* and *decision* parts of a given elementary transition rule, respectively.

**Example 3.1** *As an example we take daily exchange rates between the Polish zloty and two currencies: the US dollar (marked with u) and the euro (marked with e). A data table consists of daily exchange rates for each business day. For simplicity, we consider only a selected fragment of the table consisting of seven consecutive business days. For the table, we build an information system in the following way. Attributes correspond to currencies, whereas objects correspond to consecutive business days. The meaning of values of attributes is the following:*

- *-1 denotes decreasing a given exchange rate in relation to the previous exchange rate,*
- *0 denotes remaining a given exchange rate on the same level in relation to the previous exchange rate,*
- *1 denotes increasing a given exchange rate in relation to the previous exchange rate.*

*An information system $S = (U, A)$ is shown in Table 1. Formally, we have: the set of objects $U = \{u_1, u_2, \ldots, u_7\}$, the set of attributes $A = \{u, e\}$, the sets of attribute values $V_u = \{-1, 1\}$, $V_e = \{-1, 0, 1\}$.*

*To represent episodes (transitions among states) as sequences of three consecutive global states we build a multistage decision transition system $MDTS$ shown in Table 2. We obtain five episodes $t_1, t_2, \ldots, t_5$. We can say that attributes from the set $A^1$ determine global states in the time instant $\tau$, attributes*

Table 1: An information system $S$.

| $U/A$ | $u$ | $e$ |
|---|---|---|
| $u_1$ | -1 | -1 |
| $u_2$ | -1 | -1 |
| $u_3$ | 1 | 1 |
| $u_4$ | 1 | -1 |
| $u_5$ | -1 | 1 |
| $u_6$ | -1 | 0 |
| $u_7$ | 1 | 0 |

*from the set $A^2$ determine global states in the time instant $\tau + 1$ and attributes from the set $A^3$ determine global states in the time instant $\tau + 2$.*

Table 2: A multistage decision transition system $MDTS$.

| $U_T/A^1 \cup A^2 \cup A^3$ | $u^1$ | $e^1$ | $u^2$ | $e^2$ | $u^3$ | $e^3$ |
|---|---|---|---|---|---|---|
| $t_1$ | -1 | -1 | -1 | -1 | 1 | 1 |
| $t_2$ | -1 | -1 | 1 | 1 | 1 | -1 |
| $t_3$ | 1 | 1 | 1 | -1 | -1 | 1 |
| $t_4$ | 1 | -1 | -1 | 1 | -1 | 0 |
| $t_5$ | -1 | 1 | -1 | 0 | 1 | 0 |

*In the multistage decision transition system $MDTS = (U_T, A^1 \cup A^2 \cup A^3)$, we can distinguish two elementary decision transition subsystems: $DTS(1,2) = (U_T, A^1 \cup A^2)$ shown in Table 3 and $DTS(2,3) = (U_T, A^2 \cup A^3)$ shown in Table 4.*

Table 3: An elementary decision transition subsystem $DTS(1,2)$ of $MDTS$.

| $U_T/A^1 \cup A^2$ | $u^1$ | $e^1$ | $u^2$ | $e^2$ |
|---|---|---|---|---|
| $t_1$ | -1 | -1 | -1 | -1 |
| $t_2$ | -1 | -1 | 1 | 1 |
| $t_3$ | 1 | 1 | 1 | -1 |
| $t_4$ | 1 | -1 | -1 | 1 |
| $t_5$ | -1 | 1 | -1 | 0 |

*In the elementary decision transition subsystem $DTS(1,2)$, we have, for example, the following elementary transition rules: $(u^1, -1) \wedge (e^1, -1) \Rightarrow (u^2, -1)$, $(u^1, -1) \wedge (e^1, -1) \Rightarrow (e^2, -1)$ which are minimal, true and realizable. In the elementary decision transition subsystem $DTS(2,3)$, we have, for example, the following elementary transition rules:*

Table 4: An elementary decision transition subsystem $DTS(2,3)$ of $MDTS$.

| $U_T/A^2 \cup A^3$ | $u^2$ | $e^2$ | $u^3$ | $e^3$ |
|---|---|---|---|---|
| $t_1$ | -1 | -1 | 1 | 1 |
| $t_2$ | 1 | 1 | 1 | -1 |
| $t_3$ | 1 | -1 | -1 | 1 |
| $t_4$ | -1 | 1 | -1 | 0 |
| $t_5$ | -1 | 0 | 1 | 0 |

$(e^2, 0) \Rightarrow (e^3, 0)$, $(u^2, -1) \wedge (e^2, -1) \Rightarrow (e^3, 1)$ *which are minimal, true and realizable.*

## 4 Some Issues on Extensions of MDISs

The extensions of dynamic information systems have been considered in [7], [2]. In this paper, we focus only on the extensions of multistage decision transition systems. Analogously to definition of extensions of information systems, we define an extension of a multistage decision transition system. So, any nontrivial extension of a given multistage decision transition system $MDTS = (U_T, A^1 \cup A^2 \cup \ldots \cup A^k)$ includes new episodes such that for each episode $t^*$ we have $a(t^*) \in V_a$ for each $a \in (A^1 \cup A^2 \cup \ldots \cup A^k)$.

In this section, we are interested in computing a degree of consistency (called a consistency factor) of a given episode from any extension of a given multistage decision transition system $MDTS$ with the knowledge included in $MDTS$. We give an efficient algorithm for computing consistency factors (Algorithm 2). This algorithm has a polynomial time complexity. An approach proposed here does not involve computing any rules from an original decision system $DS$. At the beginning, we present a significant theorem given in [1] allowing us to determine whether the new object added to the original decision system $DS$ satisfies all minimal decision rules, true and realizable in $DS$, without computing such rules. Here, the theorem taken from [1] is formulated for decision systems and expressed by means of the formalism used in the paper.

**Theorem 4.1** *Let $DS = (U, C \cup D)$ be a decision system, $DS^* = (U^*, C \cup D)$ be its extension, $Rul(DS)$ be a set of all minimal decision rules true and realizable in $DS$ and $u^* \in U^*$. For each $u \in U$ let $M_u = \{a \in C : a(u^*) = a(u)\}$ and $P_u^d = \{d(u') : u' \in U$ and $\forall_{a' \in M_u} a'(u') = a'(u)\}$ for each $d \in D$. The object $u^*$ satisfies all rules from $Rul(DS)$ if and only if for any $u \in U$ and $d \in D$ one of the following requirements is satisfied:*

1. *$card(P_u^d) \geq 2$,*
2. *$card(P_u^d) = 1$ and $d(u^*) = d(u)$.*

One can find the proof of Theorem 4.1 in [1], [9].

In order to compute a consistency factor of a given episode $t^*$ from any extension of a given multistage decision transition system $MDTS = (U_T, A^1 \cup A^2 \cup \ldots \cup A^k)$ we create a family **DTS** of elementary decision transition subsystems, i.e., $\mathbf{DTS} = \{DTS(i, i+1) = (U_T, A^i \cup A^{i+1})\}_{i=1,\ldots,k-1}$. Next, the consistency factor $\xi_{DTS(i,i+1)}(t^*)$ of the episode $t^*$ with the knowledge included in $DTS(i, i+1)$ is computed for each subsystem $DTS(i, i+1)$ from the family **DTS**. Finaly, the consistency factor $\xi_{MDTS}(t^*)$ of the episode $t^*$ with the knowledge included in $MDTS$ is calculated as (see Algorithm 1):

$$\xi_{MDTS}(t^*) = \prod_{i=1}^{k-1} \xi_{DTS(i,i+1)}(t^*)$$

The consistency factor $\xi_{DTS(i,i+1)}(t^*)$ of the episode $t^*$ with the knowledge included in $DTS(i, i+1)$ is defined in the same way as the consistency factor $\xi_{DS}(u^*)$ of the object $u^* \in U$ with the knowledge included in a decision system $DS = (U, C \cup D)$.

**Definition 4.1** *Let $DS = (U, C \cup D)$ be a decision system, $DS^* = (U^*, C \cup D)$ its extension and $u^* \in U^*$. A consistency factor $\xi_{DS}(u^*)$ of the object $u^* \in U^*$ with the knowledge included in a decision system $DS = (U, C \cup D)$ is defined as*

$$\xi_{DS}(u^*) = 1 - \frac{card(\widetilde{U})}{card(U)},$$

*where $\widetilde{U}$ is a set of objects from $U$ generating minimal decision rules true and realizable in $DS$, which are not satisfied by the object $u^* \in U^*$.*

It is easy to see that if a new object $u^*$ from the extension $DS^*$ of a decision system $DS$ satisfies each minimal decision rule true and realizable in $DS$, then $\widetilde{U} = \emptyset$. Hence, $\xi_{DS}(u^*) = 1$, i.e., we can say that the object $u^*$ is totally consistent (or in short, consistent) with the knowledge included in $DS$ and expressed by means of all minimal decision rules true and realizable in $DS$. On the other hand, if each object $u$ in a decision system $DS$ generates at least one rule which is not satisfied by the object $u^*$, then $\widetilde{U} = U$. Hence, $\xi_{DS}(u^*) = 0$, i.e., we can say that the object $u^*$ is totally inconsistent with the knowledge included in $DS$ and expressed by means of all minimal decision rules true and realizable in $DS$.

Determining a consistency factor for each episode of any extension of a given multistage decision transition system $MDTS$ we can talk about the consistent or partially consistent extension of $MDTS$.

**Definition 4.2** *Let $MDTS = (U_T, A^1 \cup A^2 \cup \ldots \cup A^k)$ be a multistage decision transition system and $MDTS^* = (U_T^*, A^1 \cup A^2 \cup \ldots \cup A^k)$ be its extension. $MDTS^*$ is called a consistent extension of $MDTS$ if and only if $\xi_{MDTS}(t^*) = 1$ for all episodes $t^* \in U_T^*$. Otherwise, i.e., if there exists an episode $t^* \in U_T^*$ such that $\xi_{MDTS}(t^*) < 1$, then $MDTS^*$ is called a partially consistent extension of $MDTS$.*

## 5 State Prediction with Extensions of MDISs

We can apply computing a consistency factor of a given episode from the extension of a given multistage decision transition system $MDTS$ to predicting future states which can appear in the system described by $MDTS$. We can assume that episodes with a greater consistency factor should appear more often in the future. An example given in this sec-

**Algorithm 1**: Algorithm for computing a consistency factor of an episode belonging to the extension of a given multistage decision transition system

**Input** : A multistage decision transition system $MDTS = (U_T, A^1 \cup A^2 \cup \ldots \cup A^k)$, an episode $t^*$ from any extension of $MDTS$.
**Output**: A consistency factor $\xi_{MDTS}(t^*)$ of the episode $t^*$ with the knowledge included in $MDTS$.

$\xi_{MDTS}(t^*) \longleftarrow 1$;
**for** *each elementary decision transition subsystem* $DTS(i, i+1) = (U_T, A^i \cup A^{i+1})$ *of* $MDTS$ **do**
  Compute $\xi_{DTS(i,i+1)}(t^*)$ using Algorithm 2 (treat $DTS(i, i+1)$ as a decision system and $t^*$ as an object in a decision system);
  $\xi_{MDTS}(t^*) \longleftarrow \xi_{MDTS}(t^*) \cdot \xi_{DTS(i,i+1)}(t^*)$;
**end**

**Algorithm 2**: Algorithm for efficient computing a consistency factor of an object belonging to the extension of a decision system

**Input** : A decision system $DS = (U, C \cup D)$, an object $u^*$ belonging to the extension of $DS$.
**Output**: A consistency factor $\xi_{DS}(u^*)$ of the object $u^*$ with the knowledge included in $DS$.

$\widetilde{U} \longleftarrow \emptyset$;
**for** *each* $u \in U$ **do**
  **for** *each* $a \in C$ **do**
    **if** $a(u) \neq a(u^*)$ **then**
      $a(u) \longleftarrow *$;
    **end**
  **end**
**end**
Remove each object $u \in U$ such that $\underset{a \in C}{\forall} a(u) = *$;
**for** *each* $u \in U$ **do**
  $M_u \longleftarrow \{a \in C : a(u) \neq *\}$;
  **for** *each* $d \in D$ **do**
    $P_u^d \longleftarrow \{d(u') : u' \in U$ and $\underset{a' \in M_u}{\forall} a'(u') = a'(u)\}$;
    **if** $card(P_u^d) = 1$ *and* $d(u^*) \neq d(u)$ **then**
      $\widetilde{U} \longleftarrow \widetilde{U} \cup \{u\}$;
      **break**;
    **end**
  **end**
**end**
$\xi_{DS}(u^*) \longleftarrow 1 - \frac{card(\widetilde{U})}{card(U)}$;

tion shows how to compute a consistency factor for a given episode using Algorithms 1 and 2 presented in Section 4.

**Example 5.1** *Let us consider a multistage decision transition system* $MDTS$ *from Example 3.1. Suppose we are given a new episode (multistage transition) presented in Table 5. We are going to determine a degree of the possibility of appearance of this episode in our system.*

Table 5: A new episode.

| $U_T/A^1 \cup A^2 \cup A^3$ | $u^1$ | $e^1$ | $u^2$ | $e^2$ | $u^3$ | $e^3$ |
|---|---|---|---|---|---|---|
| $t^*$ | -1 | -1 | -1 | 0 | 1 | 1 |

*The sets* $P_u^d$ *for the first elementary decision transition subsystem* $DTS(1, 2) = (U_T, A^1 \cup A^2)$ *of* $MDTS$ *are presented in Table 6 whereas the sets* $P_u^d$ *for the second elementary decision transition subsystem* $DTS(2, 3) = (U_T, A^2 \cup A^3)$ *of* $MDTS$ *are presented in Table 7.*

Table 6: The sets $P_u^d$ for the subsystem $DTS(1, 2) = (U_T, A^1 \cup A^2)$.

| $U_T/A^1 \cup A^2$ | $u^1$ | $e^1$ | $P_u^{u^2}$ | $P_u^{e^2}$ |
|---|---|---|---|---|
| $t_1$ | -1 | -1 | $\{-1, 1\}$ | $\{-1, 1\}$ |
| $t_2$ | -1 | -1 | $\{-1, 1\}$ | $\{-1, 1\}$ |
| $t_4$ | * | -1 | $\{-1, 1\}$ | $\{-1, 0, 1\}$ |
| $t_5$ | -1 | * | $\{-1, 1\}$ | $\{-1, 0, 1\}$ |

Table 7: The sets $P_u^d$ for the subsystem $DTS(2, 3) = (U_T, A^2 \cup A^3)$.

| $U_T/A^2 \cup A^3$ | $u^2$ | $e^2$ | $P_u^{u^3}$ | $P_u^{e^3}$ |
|---|---|---|---|---|
| $t_1$ | -1 | * | $\{-1, 1\}$ | $\{0, 1\}$ |
| $t_4$ | -1 | * | $\{-1, 1\}$ | $\{0, 1\}$ |
| $t_5$ | -1 | 0 | $\{1\}$ | $\{0\}$ |

*According to Theorem 4.1 we have the episode* $t^*$ *which satisfies all minimal decision rules true and realizable in the elementary decision transition subsystem* $DTS(1, 2)$ *because for each set* $P_u^d$ *we have that* $card(P_u^d) \geq 2$. *Hence,* $\xi_{DTS(1,2)}(t^*) = 1$, *i.e., the episode* $t^*$ *is consistent to the degree 1 (or consistent in short) with the knowledge included in the elementary decision transition subsystem* $DTS(1, 2)$. *In case of the elementary decision transition subsystem* $DTS(2, 3)$, *for some sets* $P_u^d$ *we have that* $card(P_u^d) = 1$ *and decision attribute values are different. So, the episode* $t^*$ *does not satisfy all minimal decision rules true and realizable in* $DTS(2, 3)$. *The set* $\widetilde{U}$ *of episodes from* $DTS(2, 3)$ *generating rules not satisfied by* $t^*$ *consists of the episode* $t_5$. *Therefore,* $\xi_{DTS(2,3)}(t^*) = 0.8$. *Finally, we obtain the consistency factor* $\xi_{MDTS}(t^*)$ *of* *the episode* $t^*$ *with the knowledge included in* $MDTS$ *which is:*

$$\xi_{MDTS}(t^*) = \xi_{DTS(1,2)}(t^*) \cdot \xi_{DTS(2,3)}(t^*) = 0.8.$$

*According to our approach we can say that the episode* $t^*$ *is possible to appear to the degree* 0.8 *with respect to the knowledge included in the original multistage dynamic information system* $MDIS$.

## 6 Conclusions

In this paper, we proposed an efficient method of computing consistency factors of new episodes added to multistage decision transition systems. These factors can be useful to predicting degrees of possibilities of appearing in the future episodes in the examined systems. In our approach, we assumed that all states appearing in new episodes have only known values from the original multistage decision transition systems. In the future work, we will also consider a more general case, i.e., when processes of examined systems can have new states which have not been observed yet. Another important task for further research is to propose other ways of knowledge representation, and what follows, other ways of computing consistency factors. For the sake of a polynomial time complexity of presented algorithms, the proposed approach can be used in the case of high dimensional data. Especially, such data arise in genetics. It is also necessary to examine our approach on real-life data.

**Acknowledgements**

This paper has been partially supported by the grant from the University of Information Technology and Management in Rzeszów, Poland.

## References

[1] M. Moshkov, A. Skowron, Z. Suraj (2006). On Testing Membership to Maximal Consistent Extensions of Information Systems. In *Proceedings of the conference RSCTC'2006*, Kobe, Japan, November 2006, LNAI, volume 4259, pages 85-90, Springer-Verlag, Berlin Heidelberg.

[2] K. Pancerz, Z. Suraj (2008). Rough Sets for Discovering Concurrent System Models from Data Tables. In: *Rough Computing. Theories, Technologies and Applications*, pages 239-268, Information Science Reference, Hershey.

[3] Z. Pawlak (1991). *Rough Sets - Theoretical Aspects of Reasoning About Data.* Kluwer Academic Publishers, Dordrecht.

[4] Z. Pawlak (1992). Concurrent Versus Sequential: the Rough Sets Perspective. In *Bulletin of the EATCS*, number 48, pages 178-190.

[5] Z. Pawlak (2004) Some Issues on Rough Sets. In *Transactions on Rough Sets I*, LNCS, volume 3100, pages 1-58, Springer-Verlag, Berlin Heidelberg.

[6] Z. Suraj (1998). The Synthesis Problem of Concurrent Systems Specified by Dynamic Information Systems. In *Rough Sets in Knowledge Discovery 2*, Studies in Fuzziness and Soft Computing, volume 19, pages 418-448, Physica-Verlag, Berlin.

[7] Z. Suraj, K. Pancerz (2005). Some Remarks on Computing Consistent Extensions of Dynamic Information Systems. In *Proceedings of the conference ISDA'2005*, pages 420-425, Wroclaw, Poland, September 2005.

[8] Z. Suraj, K. Pancerz (2006). A New Method for Computing Partially Consistent Extensions of Information Systems: A Rough Set Approach. In *Proceedings of the conference IPMU'2006*, volume III, pages 2618-2625, Paris, France, July 2006.

[9] Z. Suraj, K. Pancerz (2007). Towards Efficient Computing Consistent and Partially Consistent Extensions of Information Systems. In *Fundamenta Informaticae*, volume 79(3-4), pages 553-566, IOS Press, Amsterdam.