

## A History-oriented Envisioning Method

Takashi Washio and Hiroshi Motoda  
Division of Intelligent Systems Science,  
The Institute of Scientific and Industrial Research, Osaka University  
8-1 Mihogaoka, Ibaraki, Osaka 567, Japan  
E-mail {washio,motoda}@sanken.osaka-u.ac.jp  
Phone: 81-6-879-8540, Fax : 81-6-879-8544

### Abstract

A novel and generic approach named as "history-oriented envisioning" is proposed to qualitatively envision all the possible and the sound situations focusing on our intended partial behaviors and actions of an objective system. Some basic notations called a "partial history" and a "partial situation" are introduced as the extensions of the conventional history and its situations representing qualitative, temporal behaviors and actions of the system. They provide the basic information of the intended partial behaviors and actions to the history-oriented envisioning. A major characteristic of the envisioning proposed here is its low complexity attained by the specified history of behaviors and actions in addition to the conventional scenario. Another important characteristic is its incremental structure enabling the iterative import of the predicted and/or observed information of the objective system. The feasibility of the proposed method is demonstrated by an example of controlling a steam generator of a plant. This method provides a basic measure to qualitatively estimate and plan system behaviors, and is considered to be applicable to wide engineering areas such as simulation, planning, design and diagnosis.

### 1 Introduction

One of the primary tasks of qualitative reasoning is the envisioning of system "behaviors" [de Kleer and Brown 1984; Forbus 1984, 1988; Kuipers 1984, 1986]. The "behaviors" are collection of possible situation transitions that result by system operations. The conventional framework of the envisioning consists of "attainable envisioning" and "total envisioning". The former derives the qualitatively possible situation transitions achievable from a specified initial state of a system. The latter derives all the possible situation transitions that may occur in the operations. The basic idea of these methods is to evaluate possible and sound behaviors of a system while maintaining a set of initially given background assumptions associated with the exogenous quantity states, views and processes of the system without imposing intentional changes of the assumptions.

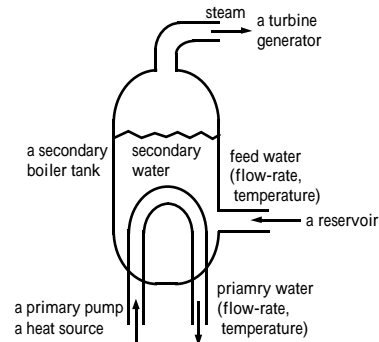
In contrast with this standard methodology, the authors are interested in the envisioning in case that some portions of the evolutionary system behaviors are specified in advance by our intention. Some work to introduce exogenously specified quantity, view and process transitions into the envisioning has been conducted [Forbus 1989; Drabble 1993; Iwasaki et al. 1993; Vescovi et al. 1993]. Forbus defined an "action" as an exogenous replacement of some background assumptions in a system scenario, and established "action-augmented envisioning" that enumerates all the possible transitions among combinations of quantity states, views, processes and actions. Drabble extended the notion of the actions to involve the exogenous specification of quantity states and to have qualitative time intervals. He also introduced a hierarchical sequence of actions to represent complex influences that are exogenously driven. Iwasaki and Vescovi proposed a language to specify intended functional behaviors in terms of causal transition rules. The latter two studies utilize in principle the repetitions of the attainable envisioning to search for their intended system behaviors. However, the difficulty of combinatorial explosion of the derived situations has been reported in the aforementioned envisioning methods, especially for the total and action-augmented envisioning, when the methods were adopted to practical scale applications [Caloud 1987; Forbus 1989; Forbus and Falkenhainer 1990, 1992; Amador et al. 1993]. For example, a self-explanatory simulation system "SimGen Mk2" to envision only local states of a system requires 4 hours to compile a simulator for a model of 9 containers and 12 pipes [Forbus and Falkenhainer 1992]. The main cause of this difficulty arises from the vast number of possible states envisioned, e.g. almost 1012 states, even for such a simple system.

An efficient remedy to this difficulty is to restrict the scope of the envisioning within the partial behaviors and actions intentionally specified following our interests, observations or the objectives of application tasks. For example, consider to control a steam generator in a power plant depicted in Fig. 1. It has a primary water tube (p-tube) passing through a secondary boiler tank (s-tank). Highly pressurized hot water is supplied from a primary heat source by a pump. When the temperature of the primary water (p-water) is higher than the boiling point of the secondary water (s-water) in the low pressure tank, the heat flow from the primary to the secondary side can boil the secondary water. To compensate for the decrease of amount of the secondary water due to the escape of the steam (s-steam) to a turbine generator, the extra water feed (f-water) to the tank through a feed pipe (f-pipe) is required. An appropriate flow control of f-water is crucial for an efficient and safe operation of the power plant. If we apply the conventional envisioning methods to plan the control strategies, the envisioning system must address all the internal states of this component together with all the combinations of states of boundary quantities such as inlet temperatures of p-water, f-water and flow rate of p-water. This induces large ambiguity and complexity of the solution. On the other hand, by allowing only the state trajectories satisfying our observation that "the temperature of p-water is increasing," and our control objectives that "boiling of s-water should occur," for instance, the ambiguity and the complexity of the resultant envisionment become considerably reduced without loss of utility for our planned task to figure out any strategies to start up the steam generation. As the information on task objectives and system boundaries is available in many engineering applications, this approach can be widely used to enhance the applicability of envisioning methods. Many works on simulation, planning, diagnosis and design in the field of qualitative reasoning utilize the envisioning to obtain the information associated with specific system behaviors [DeCoste 1990, 1993; Drabble 1993; Forbus 1986; Forbus and Falkenhainer 1990, 1992; Ishida and Eshelman 1988; Iwasaki 1993; Pearce 1988; Umeda et al. 1991; Yannou 1993]. Their efficiency may also be increased by this approach.

The work presented here is to propose a novel and generic envisioning method focused on specific partial behaviors and actions of a system, called "history-oriented envisioning". The feasibility of the proposed method is demonstrated by an example of controlling a steam generator in the latter half of this paper.

## 2 Partial Situation and Partial History

We define first an efficient representation of the partial behaviors and actions that we intentionally specify. The fundamental structure of temporal behaviors and actions has been discussed in detail in the past work [Hayes 1979; Forbus 1984, 1989; Williams 1984; Dean and McDermott 1987]. Hayes and Forbus defined a sequence of changes of objects in a scenario as a "history" consisting of "situations". A situation of a history denotes a piece of a history at a particular time, and is either of "episodes" or "events". Events always last only for an instant, while episodes usually occur over a time interval. Each episode has a start and an end which are events that serve as its boundaries. Both of an event and an episode can involve the descriptions of quantity states, views, processes, actions, their relations and their transitions at some time (or a time interval). An assertion representing one of such descriptions is called a "token" [Dean and McDermott 1987]. Each token states a primitive fact in an event or an episode such as "amount of water in a pot is 1kg." or "boiling of water occurred."



**Fig. 1.** A steam generator in a power plant for electricity generation.

Based on these definitions, some novel ideas on the history are introduced as follows.

**Definition 1:** A "partial event" is a set of some tokens involved in an event of a history.

A "partial episode" is a set of some tokens involved in an episode of a history.

**Definition 2:** A "partial situation" of a history is either of a partial event and a partial episode.

The notation of a partial situation consists of a list of individuals that must exist, a list of quantity values and relations indicating the objects' states, a list of views, a list of processes and a list of actions. The detailed descriptions of the contents in each list follow the notations in QP-theory [Forbus 1984]. T-operators are used to say that a particular token is true at some time, and M-operators represent the measured value of a quantity at some time. An extra notation represented as a predicate "Status(x, y)" has been newly introduced in our description. The symbol x is one of a view, a process and an action. The symbol y is one of "active", "inactive", "activated" and "inactivated". For instance, Status(Process\_A, active) means that Process\_A keeps its active status during a partial situation. Status(Action\_B, inactivated) says Action\_B is finished during a partial situation.

Two of the partial situations for an example of catching a ball falling through a flame depicted in fig.2 are as follows.

Partial Situation Heat-Flow-to-Ball-Active(?time)

Individuals: ball a ball  
flame a flame

Quantities: (T A[temperature-of(ball)] < A[temperature-of(flame)] ?time) (1)  
(M A[temperature-of(ball)] ?time) = T1  
(M Ds[temperature-of(ball)] ?time) = 1

Views:

Processes: (T Status(Heat-Flow(flame, ball, flame-ball),  
Active) ?time)

Actions:

Partial Situation Catching-Ball-under-Flame(?time)

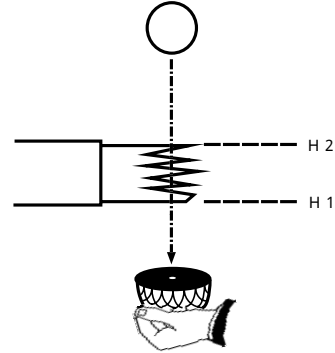
Individuals: ball a ball  
flame a flame  
basket a basket (2)

Quantities: (M A[position-of(ball)] ?time) = (-, H1)  
(M Ds[position-of(ball)] ?time) = -1

Views: (T Status(Contained-Stuff(ball, basket),  
Activated) ?time)

Processes:

Actions: (T Status(Catch-In(ball, basket),  
Activated) ?time)



**Fig.2.** Catching a ball falling through a flame.

The term "?time" of the T-operator and M-operator represents the temporal specification of the partial situation, and its value is specified by the description of partial history as explained later. The former partial situation represents that the objects; a ball and a flame exist at a particular time. The temperature of the ball and the sign of its derivative are T1 and positive respectively. The temperature of the ball is lower than that of the flame. The process of heat flow is taking place at the same time. The latter means that the objects; a ball, a flame and a basket exist, and the ball falling through the flame is caught and settled in the basket at some time. Some lists and their contents can be left unspecified in a partial situation. For instance, the position of the ball are not given in the former partial situation, but the quantity must be given to specify a unique state of the ball. A distinct partial situation is "No-Specification(?time)" in which all lists are empty. This partial situation is used to represent that any behaviors and actions are unspecified at some time.

The contents in the lists of individuals and quantities of a partial situation are used as the

assumptions for the history-oriented envisioning. For instance, the assertion of "ball a ball" in the individual lists is an assumption specified by the partial situations in the above examples. On the other hand, the views, the processes and the actions in their lists do not represent their assumptions directly. The views and the processes in the scenario and the domain model of the QP-theory have the information of "Individuals", "Preconditions" and "Quantity Conditions" [Forbus 1984]. Also, the actions have the part of "Individuals" [Forbus 1989]. These are their assumptions for them to be active. Accordingly, the unifications of the contents in the view, the process and the action lists to the scenario and the domain model in the envisioning system provide their assumptions for the partial situation. For instance, the unification of "Heat-Flow(flame, ball, flame-ball)" in the process lists of the partial situation (1) to the following domain model (3) of the heat-flow process instantiates the contents in the "Individuals", "Preconditions" and "Quantity Conditions" of the process as the corresponding assumptions.

Process Heat-Flow(?src, ?dst, ?path)

Individuals:	?src an object, Has-Quantity(?src, heat) ?dst an object, Has-Quantity(?dst, heat) ?path a Heat-Path, Heat-Connection(?path, ?src, ?dst)	
Preconditions:	Heat-Aligned(?path)	(3)
Quantity Conditions:	A[temperature(?src)]>A[temperature(?dst)]	
Relations:	Quantity(flow-rate) flow-rate = (temperature(?src)-temperature(?dst))	
Influences:	I-(heat(?dst), A[flow-rate]) I+(heat(?src), A[flow-rate])	

Another important assumption in a partial situation is the duration of ?time. Its specification controls the generation of the limit hypotheses in the envisioning process as explained later. The history-oriented envisioning utilizes all of the assumptions described here for a scenario involving the partial situation.

The definition of a "partial history" is given based on the partial situations.

*Definition 3:* A "partial history" of a history is a set of partial situations of the history which time intervals and instants are totally ordered in time domain.

A partial history has a list of the T-operators to say that a particular partial situation is true at some time. It also has a list of time constraints. An example of a partial history for the ball is shown here.

Partial History Initial-and-Final-Ball

Partial Situations:	(T Initial-Position-of-Ball(I0) I0) (T Position-Decreasing-of-Ball-above-Flame(I1) I1) (T Heat-Flow-to-Ball-Active(I2) I2) (T No-Specification(I3) I3) (T Position-Decreasing-of-Ball-under-Flame(I4) I4) (T Catching-Ball-under-Flame(I5) I5)	(4)
Time Constraints:	(start(I0)=end(I0)), (end(I0)=start(I1)), (start(I1)<end(I1)), (end(I1)=start(I2)), (start(I2) end(I2)), (end(I2)=start(I3)), (start(I3)<end(I3)), (end(I3)=start(I4)), (start(I4)<end(I4)), (end(I4)=start(I5)), (start(I5)=end(I5))	

the partial situations before No-Specification(I3) specifies the partial behaviors of a ball beginning with its initial position until it touches the flame, while those after No-Specification(I3) describes the partial behaviors and actions associated with the ball under the flame until it is caught in a basket. Hence, this partial history specifies two clusters of partial situations mutually apart in time domain.

The list of time constraints specifies the temporal characteristic of each partial situations, and follows the conditions indicated below with respect to its duration.

$$\begin{array}{ll}
\text{start}(\text{?time})=\text{end}(\text{?time}) & \text{?time is an instant,} \\
\text{start}(\text{?time})<\text{end}(\text{?time}) & \text{?time is an interval,} \\
\text{start}(\text{?time}) \quad \text{end}(\text{?time}) & \text{The duration of ?time is unspecified.}
\end{array} \tag{5}$$

For example, (T Initial-Position-of-Ball(I0) I0) and (T Catching-Ball-under-Flame(I5) I5) hold only at an instant. Thus they have the identical starting and ending times, and must be partial events. The partial situation, (T Heat-Flow-to-Ball-Active(I2) I2), is specified as either of a partial event and a partial episode, because its duration is not given in this example.

The value of ?time of each partial situation specified in the partial history is substituted to the variable ?time in the representation of the respective partial situation. For example, the values of I2 and I5 are substituted to the variables of partial situation (1) and (2), respectively. Then, the value is used to check the temporal characteristic of the partial situation and control the envisioning process explained later by following the rules indicated below.

#### Condition on Limit Hypotheses:

A partial situation involves some limit hypotheses.

?time is an instant, i.e. the partial situation is a partial event. (6)

?time is an interval, i.e. the partial situation is a partial episode.

A partial situation does not involve any limit hypotheses.

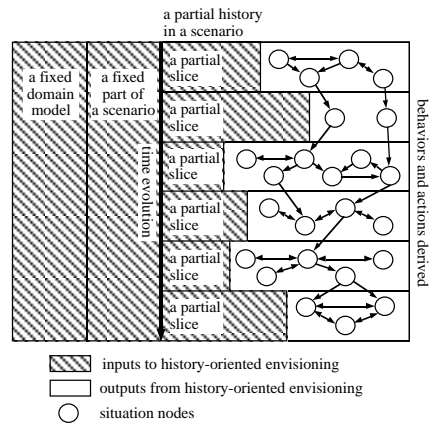
These conditions states that any partial situation involving some limit hypotheses should be a partial event. However, a partial event does not necessarily involve limit hypothesis, because some tokens not belonging to the partial event within the event may be some limit hypotheses. On the other hand, any partial episode should not involve any limit hypothesis by definition.

A partial history given to the history-oriented envisioning is part of the scenario for the envisioning. Some partial histories may not be realized in the possible histories of the scenario. For example, the heat flow from the flame to the ball does not take place before the ball touches the flame . When such a partial history is specified, the history-oriented envisioning halts at its intermediate step, and does not generate any envisionment consistent with the partial history.

### 3 History-oriented Envisioning

#### 3.1 Overview of History-oriented Envisioning

The outline of the history-oriented envisioning is depicted in Fig.3. The vertical direction from the top to the bottom of the box stands for the time evolution of the behaviors and actions of an objective system. The horizontal axis represents the spectrum of the assumptions in the envisioning process. The shadowed area is the input information to the history-oriented envisioning, while the white part is its output. The domain model and a part of the scenario for the objective system are fixed over the entire time evolution in the envisioning. The conventional envisioning enumerates the situation nodes grounded on all the possible and sound combinations of the open assumptions associated with the system. On the other hand, the history-oriented envisioning imports the given specifications on some part of the assumptions for each time interval and instant in the form of the partial situation. It derives situation nodes that are allowed within all the possible and sound combinations of the remaining open assumptions following the



**Fig.3.** The outline of history-oriented envisioning.

order of the specified time interval or instant. Accordingly, the history-oriented envisioning focuses on only the situations of the objective system that match the specified partial behaviors and actions. The number of the derived situation nodes and the computation time are significantly reduced due to the low ambiguity of the conditions that are exponential to the number of the open assumptions.

### 3.2 Algorithm

A partial history for the history-oriented envisioning must be compiled in advance to derive all the assumptions associated with its partial situations. The algorithm to compile the partial histories is depicted in Fig.4. This derives the set of assumptions,  $\Psi_i$ , for every partial situation,  $\text{PS}(i)$  ( $i=1, \dots, n$ ), and its time constraints,  $\text{TC}(i)$ , in the partial history by unifying their views, processes and actions to the scenario and the domain model in the envisioning system. The series of the partial situations and their time constraints in a partial history is sequentially compiled. (Step 3) and (step 4) collect the assumptions of a partial situation. (Step 3) obtains the assumptions explicitly represented in the individuals lists, the quantities lists and the time constraints of the partial situation, while (step 4) derives implicit ones of the views, processes and actions not directly represented in the partial situation. (Step 2) and (step 5) check the violation of Condition on Limit Hypotheses in the given partial situations, and quit the compiling if any violations are detected. (Step 2) checks any explicit limit hypotheses appearing in a partial episode, e.g., (T Status(Catch-In(ball, basket), Activated) Interval). On the other hand, (step 5) checks any implicit limit hypotheses appearing in the assumptions of a partial episode unified in (step 4), e.g.,  $[A[\text{temperature}(\text{ball})]=$

- (step 1)  $i \leftarrow 1$ .
- (step 2) Choose the  $i$ th partial slice  $\text{PS}(i)$  and the time constraints  $\text{TC}(i)$  on  $\text{PS}(i)$ .  
If any explicit limit hypotheses appear in any lists of the individuals, the quantities, the views, the processes and the actions in  $\text{PS}(i)$ , check Condition on Limit Hypotheses. If it is violated, then stop.
- (step 3) Let  $\Psi_i$  be a set of the contents in  $\text{TC}(i)$ , the individuals lists and the quantities lists of  $\text{PS}(i)$ .
- (step 4) Unify the views, the processes and the actions in their lists to the scenario and the domain model in the envisioning system, and let  $\Psi_{su}$  be a set of the contents in the individuals, the preconditions and the quantity conditions of the unified predicates.  
 $\Psi_i \leftarrow \Psi_i \cup \Psi_{su}$ .
- (step 5) If any implicit limit hypotheses appear in  $\Psi_i$ , check Condition on Limit Hypotheses. If it is violated, then stop. If  $i < n$ , then go to (step 2), else end.

**Fig.4.** An algorithm of compiling a partial history.

- (step 1)  $i \leftarrow 1$ . Let the set of assumptions  $\Pi_i$  be  $\Pi_f \cup \Psi_i$ . Perform a total (or action-augmented) envisioning under  $\Pi_i$ , and let  $Q$  be a set  $\{q(i,j) \mid q(i,j) \text{ is } j\text{th situation node generated in the envisioning, and } j=1, \dots, m_i\}$ . If  $Q$  is null, then stop.
- (step 2)  $R \leftarrow \{ \}$ .  
For  $j=1$  to  $m_i$  {  
  - (step 2.1) Let the set of assumptions  $\Pi_{i,j}$  be  $\Pi_f \cup \text{Initial}(q(i,j))$ . Perform one step attainable (or attainable action-augmented) envisioning under  $\Pi_{i,j}$ .
  - (step 2.2) Filter only the situation nodes which satisfies all conditions specified in  $\Psi_{i+1}$ .  
Let  $R_f$  be a set of the filtered situation nodes, and  $R \leftarrow R \cup R_f$ . }
- If  $R$  is null, then stop.
- (step 3) Remove every situation node from  $Q$  which is not reachable to any  $r(j) \in R$  ( $j=1, \dots, m_r$ ).
- (step 4)  $i \leftarrow i+1$ .  $S \leftarrow \{ \}$ .  
For  $j=1$  to  $m_r$  {  
  - Let the set of assumptions  $\Pi_{i,j}$  be  $\Pi_f \cup \Psi_i$ .
  - Initial( $r(j)$ ). Perform an attainable (or attainable action-augmented) envisioning under  $\Pi_{i,j}$ .
  - Let  $S_f$  be a set of the situation nodes generated in the envisioning, and  $S \leftarrow S \cup S_f$ . }
- (step 5)  $S$  is represented as  $\{q(i,j) \mid j=1, \dots, m_i\}$ .  $Q \leftarrow Q \cup S$ .  
If  $i < n$ , then go to (step 2), else end.

**Fig. 5.** An algorithm of history-oriented envisioning.



100)&(Ds[temperature(ball)]=1) in the quantity conditions of a process.

Once, all the assumptions are obtained, they are applied to the algorithm of the history-oriented envisioning represented in Fig.5. It is noted that this incrementally accepts the assumption sets of partial situations in a partial history in accordance with their total order described in the time constraints of the history. Pf is the fixed portion of the background assumptions for a scenario over the envisioning. Pf corresponds to the conventional scenario excluding the partial history.

(Step 1) is to enumerate all the possible situations for the first partial situation. The total envisioning (or action-augmented envisioning, if possible actions must be taken into account.) under the conditions of Pf and Ps1 is required at this step because any preceding situations have not been specified. If the first partial situation is not consistent with the Pf, then no solutions are obtained, and the process halts.

(Step 2) is to identify all the situations for the next partial situation which are directly caused by the current situations. In (step 2.1), all the possible one step transitions from a current situation to the next are figured out. The one step attainable (or attainable action-augmented) envisioning is an ordinal attainable (or attainable action-augmented) envisioning under a given initial condition,  $q(i,j)$ , but it is limited to one situation transition. The notation,  $\text{Initial}(q(i,j))$ , expresses that  $q(i,j)$  is a given initial condition for the envisioning. (Step 2.2) filters out any inconsistent situations with the next partial situation. When the next partial situation is the No-Specification(?time), all the situations obtained in (step 2.1) is passed. On the contrary, when any current situations can not transit to any consistent situations with the specifications of the next partial situation, the process halts.

(Step 3) is a sort of retrospective reasoning, while the other steps are perspective. Q contains all the possible histories from the first to the current partial situation. Because the results of the filtering in (step 2) limits the possible transitions from the given partial situations, some histories in Q may not be causatively connected to the next partial situation. This step eliminates such dead end histories in Q.

(step 4) enumerates all possible situations in the new partial situation causatively occurred from the previous partial situation based on the attainable (or attainable action-augmented) envisioning. Each envisioning starts from a situation figured out in (step 2), and a list of all possible situations evolved for this partial situation is obtained.

(step 5) simply accumulates the situations for the new partial situation in Q. An envisionment following the given partial history is rested in Q, when all partial situations in the partial history have been processed.

The heaviest computational load in this algorithm is at (step 4). The load strongly depends on the number of its initial states R generated in (step 2), and the number mostly depends on the efficiency of the situation filtering in (step 2.2) associated with the preceeding partial situations. Hence, the computational load will be substantially reduced when many specifications are included in each partial situations. The loads of the other steps are not very significant. The total (or action-augmented) envisioning in (step 1), whose algorithm is essentially efficient, is performed only once, and its processing speed is accelerated by specifying the first partial situation. (Step 2) that performs only one step reasoning for each situation transition is quite efficient. (Step 3) is merely a network search for which various efficient algorithms are available. The simplicity of (step 5) is trivial.

An advantage of this algorithm is that the conventional total and attainable envisioning [Forbus 1984, 1988, 1989] can be utilized as parts of its process while reducing the solutions and processing time based on the information in a partial history. A unique difference of the envisioning utilized here from the conventional one is the imposition of the following rules on the generation of situation nodes that are associated with Condition of Limit Hypotheses on the assumption of TC(i). They reject the situation nodes involving any limit hypotheses in a finite time interval.

$$\begin{aligned}
&(\text{start}(I_i)=\text{end}(I_i)) \& (\text{The assumptions of PS}(i) \text{ do not involve any limit hypotheses.}) \\
&\quad (\text{The assumptions generated for the situation node in the envisioning} \\
&\quad \text{must involve some limit hypotheses.}), \\
&(\text{start}(I_i)<\text{end}(I_i)) \\
&\quad (\text{The assumptions generated for the situation node in the envisioning} \\
&\quad \text{must not involve any limit hypotheses.}), \\
&\text{where } I_i \text{ is an instant or an interval for PS}(i).
\end{aligned} \tag{7}$$

Another advantage is its incremental structure to process a partial history. This enables on-line application where new partial situation information is imported step by step. Especially, when each partial situation is well specified in on-line use, real time processing will be possible. These features of the algorithm are expected to be highly profitable for the real time applications of control, planning, measurement interpretation and diagnosis.

### 3.3 Soundness and Complexity

The standard total (or action-augmented) envisioning is sound for all the possible system behaviors and actions under closed world assumptions the members of which are the only possible assumptions for the scenario [Forbus 1988]. The standard attainable (or attainable action-augmented) envisioning is also sound for its possible initial conditions under the closed world assumptions. Hence, each standard envisionment generated in (step 1), (step 2.1) and (step 4) in the algorithm depicted in fig.5 is sound for the given assumptions. The other steps; (step 2.2) and (step 3) reduce the generated nodes. (Step 2.2) is clearly sound because it just filters situation nodes that are consistent with the constraints required for the transitions from the current situation to the next just like the standard envisioning internally does. (Step 3) is also sound since it keeps track of all the histories which do not contradict the assumptions of any partial situations and the scenario's fixed part in the context of a given partial history. These observations support the soundness of the history-oriented envisioning conducted by the algorithm of Fig.5 under the closed world assumptions.

The complexity of an envisioning process sensitively depends on the number of unspecified assumptions for an envisionment [Forbus 1988, 1989]. Let  $P$  be the set of assumptions for a scenario, where its fixed portion is  $P_f \subseteq P$ . The set of unspecified assumptions for the standard envisioning is  $P_u = P - P_f$ , because it does not utilize any information of a partial history. If  $P_u$  consists of pairs of in and out of assumptions, i.e.,  $p$  and  $\neg p$ , the number of states could increase by  $O(2^{|P_u|-1})$ . On the contrary, each partial situation specifies some portion of  $P - P_f$  in the history-oriented envisioning. The part of unspecified assumptions in  $P$  with respect to the first partial situation is  $P_{u1} = P - P_f - P_{s1}$ . Hence, the complexity of the total envisioning in (step 1) of fig.5 is proportional to  $O(2^{|P_{u1}|-1})$ . The attainable envisioning through (step 2) to (step 4) is performed for the unspecified assumptions of  $P_{ui} = P - P_f - P_{si}$ , and its initial situations are limited to the preceding envisionment. Accordingly, its initial complexity could be less than  $O(2^{|P_{ui}|-1})$ . These lead to the maximum complexity of  $O(2^{|P_{u1}|-1}) + O(\sum_{i=2}^n 2^{|P_{ui}|-1})$  for the history-oriented envisioning. Since the number of partial situations in a partial history,  $n$ , is independent of the assumptions, and also each,  $|P_{ui}|$  is equal or less than  $|P_{u1}|$ , the complexity of the history-oriented envisioning can be quite small comparing with the standard.

## 4 An Example

The proposed history-oriented envisioning has been applied to the control of a steam generator depicted in Fig.1 commonly used in power plants for electricity generation, and its performance evaluated. At the beginning of its operation, the secondary water has not started boiling yet. We could qualitatively determine the future change of the primary water flow rate and its temperature based on the operational conditions of the heat source and the primary pump in the upper stream. The future change of the temperature of the secondary feed water is also qualitatively assumed



based on the information of its reservoir. The temperatures of p-water and f-water are assumed to increase monotonically, while the flow rate of p-water is assumed to decrease monotonically in the mean time, and three of them are assumed to reach an equilibrium at certain levels after some time. Our task is to plan all the sound control strategies of the secondary water that is fed to the tank to start boiling after the transient of the three boundary quantities is over.

A possible partial history corresponding to our mission can be written as shown in Fig.6. It specifies the intended behaviors of the steam generator together with the assumed changes of the boundary quantities such as the temperatures of p-water and f-water. The occurrence of boiling of the secondary water is intended at the final stage of the transients. Figure 7 represents two partial situations in the partial history. The former specifies the initial situation associated with the three boundary quantities, an endogenous quantity, i.e. temperature-of(s-water), and the intended processes. The latter specifies that the endogenous temperature-of(s-water) reaches its boiling point, and simultaneously the boiling process is activated when the three boundary quantities reach their goal levels maintaining the amount-of(s-water), the heat-flow and the fluid-flow.

The fixed portion of the scenario for this system and its partial history in Fig.6 has been compiled to the assumptions and applied to the algorithm of history-oriented envisioning. The algorithm has been implemented and a program specific to this type of examples has been developed and tested out on a SPRAC-10 machine. A more generic program is currently under development. Figure 8 depicts the resultant envisionment indicating all the possible and sound strategies to control the boiling under the given partial history. A total of 29 situations were found. The author have tried to derive the total envisionment of this steam generator without specifying any partial history for comparison. However, the solution was not obtained due to the limitation of the memory capacity of the current program. The possible situations in the total envisionment can be at least more than 6000 even for this simple system since it has 4 free boundary quantities. The planning of control strategies for process systems such as this example is highly expensive unless appropriate constraints are introduced.

## 5 Discussions and Related Works

One of the major characteristics of the history-oriented envisioning is the direct specifications of behaviors and actions to the envisioning process in addition to the conventional scenario. The envisioning focuses only on the specified situations and their histories, and derives situations including those with small computational load. Iwasaki and Vescovi introduced a language, CFRL, to specify intended functional behaviors [Iwasaki et al. 1993; Vescovi et al. 1993]. However, the CFRL just filters intended behaviors from the possible behaviors that result in the envisioning, and hence does not control the envisioning process directly. In contrast, the efficient behavior focusing capability of the history-oriented envisioning enhances the applicability of the envisioning theory to the problems of practical scale.

Another major characteristic is the explicit use of not only our intentional actions to eliminate unrequired or useless solutions for our reasoning tasks but also the information on the behaviors' history we intend on the objective system. Drabble developed a system, EXCALIBUR, for plan-

Partial History Boiling-Control	
Partial Slices:	(T Initial-State(I0) I0) (T Start-of-Transient(I1) I1) (T Monotonic-Transient(I2) I2) (T End-of-Transient-and- Start-of-Boiling(I3) I3) (T Final-State(I4) I4)
Time Constraints:	(start(I0)=end(I0)), (end(I0)=start(I1)), (start(I1)=end(I1)), (end(I1)=start(I2)), (start(I2)<end(I2)), (end(I2)=start(I3)), (start(I3)=end(I3)), (end(I3)=start(I4)), (start(I4)=end(I4))

**Fig. 6.** A partial history to control the boiling of secondary water.

ning and reasoning with process systems [Drabble 1993]. It utilizes some attainable envisioning processes, and can manage the actions changing continuous process quantities not only the ones causing discontinuous change of views and processes. It can also take a tree and hierarchical structure of action sequences, but the behavior that evolves in the process cannot be explicitly specified in the envisioning. As many applications such as planning, simulation and design in practical fields are usually seeking objective processes or intended behavior sequences that can be specified in advance, the history-oriented envisioning provides an efficient approach to the synthetic tasks.

The third important characteristic is the incremental structure of the history-oriented envisioning. Work on the measurement interpretation utilizes the total envisionments of the objective system to interpret the situation transition [Forbus 1986; DeCoste 1990, 1993]. However, the total envisioning is quite expensive for processes of practical scale. In contrast, the incremental feature of the history-oriented envisioning enables to take a sequence of behaviors and actions one by one in the on-line monitoring process. If there is a reasonable amount of information taken, its

Partial Slice Initial-State(?time)	
Individuals:	p-tube a pipe f-pipe a pipe s-tank a container p-water a contained liquid s-water a contained liquid f-water a contained liquid
Quantities:	(T A[temperature-of(p-water)] >A[temperature-of(f-water)] ?time) (M A[temperature-of(p-water)] ?time) = T <sub>pmin</sub> (M Ds[temperature-of(p-water)] ?time) = 0 (M A[temperature-of(f-water)] ?time) = T <sub>fmin</sub> (M Ds[temperature-of(f-water)] ?time) = 0 (T A[temperature-of(s-water)] ?time) <A[t-boil(s-water)] ?time) (M Ds[temperature-of(s-water)] ?time) = 0 (M A[flow-rate-of(p-water)] ?time) = F <sub>pmax</sub> (M Ds[flow-rate-of(p-water)] ?time) = 0
Views:	
Processes:	(T Status(Heat-flow(p-water, s-water, p-tube), Active) ?time) (T Status(Fluid-flow(f-water, s-water, f-pipe), Active) ?time) (T Status(Boiling(s-water, Heat-flow), Inactive) ?time)
Actions:	
Partial Slice End-of-Transient-and-Start-of-Boiling(?time)	
Individuals:	p-tube a pipe f-pipe a pipe s-tank a container p-water a contained liquid s-water a contained liquid f-water a contained liquid
Quantities:	(T A[temperature-of(p-water)] >A[temperature-of(f-water)] ?time) (M A[temperature-of(p-water)] ?time) = T <sub>pmax</sub> (M Ds[temperature-of(p-water)] ?time) = 1 (M A[temperature-of(f-water)] ?time) = T <sub>fmax</sub> (M Ds[temperature-of(f-water)] ?time) = 1 (T A[temperature-of(s-water)] ?time) =A[t-boil(s-water)] ?time) (M Ds[temperature-of(s-water)] ?time) = 0 (M A[flow-rate-of(p-water)] ?time) = F <sub>pmin</sub> (M Ds[flow-rate-of(p-water)] ?time) = -1 (M A[amount-of(s-water)] ?time) = (0, M <sub>smax</sub> )
Views:	
Processes:	(T Status(Heat-flow(p-water, s-water, p-tube), Active) ?time) (T Status(Fluid-flow(f-water, s-water, f-pipe), Active) ?time) (T Status(Boiling(s-water, Heat-flow), Activated) ?time)
Actions:	

**Fig. 7.** An example of a partial slice for the control of the secondary water boiling.

attainable envisioning in each step will be quite cheap, thus making its real time use possible. This characteristic meets the practical needs of analytic tasks such as measurements interpretation, control and diagnosis.

## 6 Conclusion

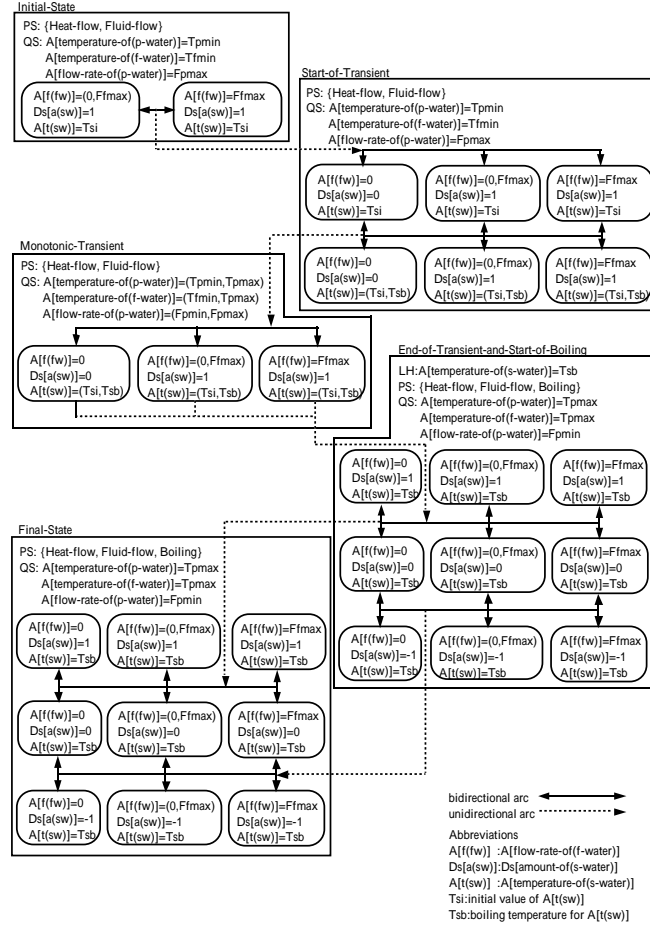
A history-oriented envisioning method has been proposed. It is characterized by its capability of receiving some intended behaviors of a system from outside. These behaviors are called "partial situations and a partial history". The method has been applied to an example of control strategy planning for a steam generator and its efficacy confirmed. The major characteristic of the history-oriented envisioning are summarized as follows.

- (1) The envisioning is sound. Its computational complexity is low and it is much more efficient than the conventional envisioning.
- (2) Envisioning is focused on a sequence of intended partial behaviors and actions.
- (3) Envisioning can be incremental to import the assumptions in an iterative manner.

The ideas presented here will promote a new progress of qualitative envisioning theory toward its application to more practical tasks of simulation, planning, design, measurements interpretation, control and diagnosis.

The future work that remains to be explored includes:

- (1) Extension of a partial history: At present the structure of a partial history is a sequence of partial situations although the sequence can be fragmented by the "No-Specification" partial situation. Its extension to a tree, graph and hierarchical structures or transition rules from a history as in the EXCALIBUR and the CFRL will enhance the usability of the history-oriented envisioning.
- (2) Seeking for a better algorithm: The current algorithm for the history-oriented envisioning is a first step to evaluate the basic performance. More efficient algorithm might be developed.



**Fig.8.** A situation transition diagram of a steam generator for a partial history.

- (3) Development of a computer program for general use: The domains of process systems that can be envisioned in the current program is quite limited. The authors are currently working on a more general program for history-oriented envisioning.

#### Acknowledgments

The authors wishes to express our thanks to Prof. Masaharu Kitamura in the Nuclear Engineering Department of Tohoku University for the useful discussions.

#### References

- [Amador et al. 1993] Franz G. Amador, Adam Finkelstein and Daniel S. Weld. Real-Time Self-Explanatory Simulation. In proceedings AAAI-93, pp.562-567, 1993.
- [Caloud 1987] Philippe Caloud. Toward Continuous Process Supervision. In proceedings IJCAI-87, pp. 1086-1089, 1987.
- [Dean and McDermott 1987] Thomas Dean and Drew V. McDermott. Temporal Database Management. Artificial Intelligence, 36(3):375-399, 1988.
- [DeCoste 1990] Dennis DeCoste. Dynamic Across-Time Measurement Interpretation. In proceeding AAAI-90, pp.373-379, 1990.
- [DeCoste 1993] Dennis DeCoste. Dynamic Across-Time Measurement Interpretation. Artificial Intelligence, 51(1-3):273-341, 1993.
- [de Kleer and Brown 1984] Johan de Kleer and John Seely Brown. A Qualitative Physics Based on Confluence. Artificial Intelligence, 24:7-83, 1984.
- [Drabble 1993] Brian Drabble. EXCALIBUR: A Program for Planning and Reasoning with Process. Artificial Intelligence, 62:1-40, 1993.
- [Forbus 1984] Kenneth D. Forbus. Qualitative Process Theory. Artificial Intelligence, 24:85-168, 1984.
- [Forbus 1986] Kenneth D. Forbus. Interpreting Measurement of Physical Systems. In proceeding AAAI-86, pp.113-117, 1986.
- [Forbus 1988] Kenneth D. Forbus. QPE: Using Assumption-based Truth Maintenance for Qualitative Simulation. Artificial Intelligence in Engineering, 3(4):200-215, 1988.
- [Forbus 1989] Kenneth D. Forbus. Introducing Actions into Qualitative Simulation. In proceedings IJCAI-87, pp.1273-1278, 1989.
- [Forbus and Falkenhainer 1990] Kenneth D. Forbus and Brian Falkenhainer. Self-explanatory Simulations: An Integration of Qualitative and Quantitative Knowledge. In proceedings AAAI-90, pp. 380-387 1990.
- [Forbus and Falkenhainer 1992] Kenneth D. Forbus and Brian Falkenhainer. Self-explanatory Simulations: Scaling Up to Large Models. In proceedings AAAI-92, pages 685-690, 1992.
- [Kuipers 1984] Benjamin Kuipers. Commonsense Reasoning about Causality: Deriving Behavior from Structure. Artificial Intelligence, 24:169-203, 1984.
- [Kuipers 1986] Benjamin Kuipers. Qualitative Simulation. Artificial Intelligence, 29:289-338, 1986.
- [Hayes 1979] P. J. Hayes. The naive physics manifesto, in: D. Michie (Ed.), Expert Systems in the Electric Age (Edinburgh University Press, Edinburgh, 1979).
- [Ishida and Eshelman 1988] Y. Ishida and L. Eshelman. AQUA: Integrating Model-Based Diagnosis and Syndrome-Based Diagnosis, cmu-cs-87-111, 1987.
- [Iwasaki et al. 1993] Yumi Iwasaki, Richard Fikes, Marcos Vescovi, and B. Chandrasekaran. How Things Are Intended to Work: Capturing Functional Knowledge in Device Design. In proceedings IJCAI-93, pp.1516-1522, 1993.
- [Pearce 1988] D. A. Pearce. The Induction of Fault Diagnosis Systems from Qualitative Models. In proceedings AAAI-88, pp. 353-357, 1988.
- [Umeda et al. 1991] Yasushi Umeda, Tetsuo Tomiyama, and Hiroyuki Yoshikawa. A Design Methodology for a Self-maintenance Machine. DE-Vol.31, Design Theory and Methodology - DTM'91 - edited by L.A. Stauffer, Book No. G00641, 1991.
- [Vescovi et al. 1993] Marcos Vescovi, Yumi Iwasaki, Richard Fikes, and B. Chandrasekaran. CFRL: A Language for Specifying the Causal Functionality of Engineered Devices. In proceedings AAAI-93, pp.626-633, 1993.
- [Williams 1984] Brian C. Williams. Qualitative Analysis of MOS Circuits. Artificial Intelligence, 24:281-346, 1984.
- [Williams 1986] Brian C. Williams. Doing time: putting qualitative reasoning on firmer ground, In proceedings AAAI-86, 1986.
- [Yannou 1993] Bernard Yannou. Qualitative Design with Envisionment. In working papers QR'93, pp.250-259. 1993.