Using Hindsight in Medical Decision Making

Thomas A. Russ[1]
Clinical Decision Making Group
MIT Laboratory for Computer Science
545 Technology Sq., Room 369
Cambridge, MA 02139
(617) 253-3533

Abstract

As the clinical picture of a patient evolves over time, more information becomes available. Certain procedures require time to perform, causing a delay between the time when the tests are ordered and when the results are available. Furthermore, as the patient's condition changes over time, serial measurements can be made. The availability of more data allows a more accurate assessment of the patient. Uncertainties, guesses or errors that were made early in the clinical course of patient care can also be identified and resolved when more information is available.

Reasoning with a stream of data that changes over time presents a challenge to the designers of expert systems. The use of hindsight in expert systems requires that appropriate attention be paid to the temporal relations of the data and that care is exercised in revising decisions. I present a data-dependency system, the Temporal Control Structure (TCS), designed to support reasoning with data changing over time and show how it can be used to implement reasoning by hindsight.

Keywords: Temporal Reasoning, Expert Systems, Artificial Intelligence.

1 Introduction

The trite observation that everyone has 20-20 hindsight recognizes that more information makes decision making easier. This is especially true about the consequences of actions undertaken. More information can lead one to revise observations, interpretations of previous data and to evaluate

[1]This research was supported by National Institutes of Health Grant No. R01 HL33041 from the National Heart, Lung, and Blood Institute, and No. R01 LM04493 from the National Library of Medicine.

or retract assumptions. In this paper, *hindsight* means using newly available information to revise decisions made earlier.

I describe the phenomenon of reasoning by hindsight, using an example drawn from cardiac patient care. I present an architecture for building expert systems that provides support for decision making based on data that changes with time. The system, called a Temporal Control Structure (TCS) takes over the responsibility for the scheduling of reasoning modules that depend on data that changes over time.

The TCS provides a framework in which decisions can be revised to take account of newly acquired information as well as information that changes with time and the evolving clinical course of a patient. The motivation for this control structure is research into expert systems for monitoring and decision making in cardiac intensive care and in the acute phase of diabetic ketoacidosis.

Reasoning by hindsight involves the reassessment of previous decisions as more information becomes available. The identification of errors, the discovery of violated assumptions, or simply the resolution of ambiguous findings becomes possible. It allows one to use response to therapy as diagnostic information. In some cases, the response itself can yield information bearing on the diagnosis of the patient's condition. Since the response of a patient to a particular treatment is modulated by the underlying disease process, an analysis of this response can shed light on that process.

For example, a diabetic in insulin shock (excess insulin) will have a dramatic improvement in his condition after the administration of sugar. Giving sugar is both a treatment and a source of confirmatory information. If there were no response, then further tests may be indicated to identify a different cause for the patient's symptoms. The lack of response to appropriate therapy can cast doubt on the accuracy of the diagnosis.

In the following sections I will describe the TCS, give an example of how it can be used in reasoning by hindsight and discuss the reasoning issues raised.

2 Organization of the TCS

The Temporal Control Structure divides a system into data, held in *variables*, and reasoning components, defined in *modules*, which operate on the variables. A module communicates with other modules via variables. The input and output variables describe the data dependency of the module. The declarations of data dependencies allows TCS to schedule the execution of modules in response to changes in data. The scheduler is responsible for maintaining an up-to-date description of the reasoning state. It also allows incorrect data to be corrected and is insensitive to the order in which data is made available to the system. This section give a short description of the TCS. A more complete account can be found in [20].

2.1 Temporal Control Structure Components

TCS uses a discrete model of time. Fuzzy ranges are not allowed. This is an efficiency measure to prevent branching in the reasoning at scheduling time.

Variables must be declared to TCS. There are two types of temporal variables that can be used, *points* and *intervals*. Points associate a single time value with each datum. Intervals associate a non-zero duration and have a begin and end time. There are no restrictions on the values that variables can take on. Intervals are constrained to have only a single value over each time period. This single value, however, can have a compound form such as a disjunction or a list as long as the alternatives are explicitly listed together. The only restriction on values is that an equality predicate (such as the Lisp `equal` function) must exist. The time(s) associated with variable values must be exactly specified.

In addition to the variables defined by the user of the TCS, there are three variables defined by the system itself: *now*, *past?* and *future?*, which contain information about the temporal state of the system. The importance of knowing the current time and distinguishing between reasoning in the past versus the future is discussed in Section 4. Essentially, actions can only be changed in the future, whereas interpretations can be changed at any time.

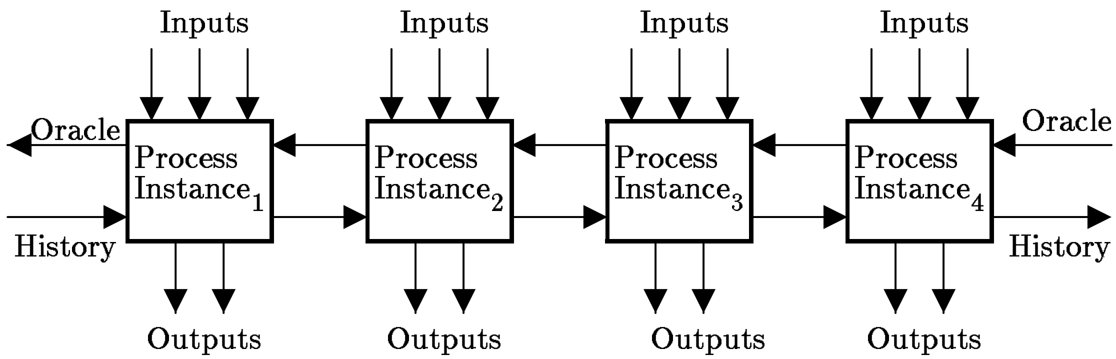


Figure 1: Schematic of a Chain of Processes

Reasoning is done by *modules*, which must declare their inputs and outputs to the system. They are procedures which calculate their outputs based on their inputs over some interval. The interval of action (in which a module is represented by a *process instance*) is chosen by TCS. It is guaranteed that each interval variable will have exactly one value throughout the interval over which a module is scheduled (See [19]). The number of such process instances for any given module will depend on the data available. As more information becomes available, more instances will be created. A module is analogous to a procedure definition in a conventional programming language and a process instance can be thought of as a procedure invocation. To assure complete updating, modules are restricted to using only the information that is declared to the system in calculating their outputs. Any deterministic computable function or procedure that obeys this constraint can be used to calculate the output.

Internal state information that is used in more than one process instance must also be declared. This internal state can take the form of information which is passed forward in time (*history* variables) or passed backward in time (*oracle* variables). It is the latter which allow reasoning by hindsight. The process instances that together form the temporal history of a particular module in execution communicate with each other through the use of the history and oracle mechanism. Figure 1 is a schematic view of process instances showing the relation between a series of process instances, their inputs, outputs and the history/oracle internal state variables.

2.2 Data Directed Updating

Whenever variables change, the modules which depend on the variables' value are scheduled for execution over the period of time of the change. If the outputs change, then the changes are propagated forward until the system quiesces.

In order to assure proper updating of the database, procedures are only allowed to affect the value of their output variables during the duration of the process instance that is executing. In order to affect other time intervals, the internal state must be used. To affect future decisions, information must be stored in history variables and passed forward for consideration by temporally later process instances. To change past decisions, the information must be passed back in time. From the point of view of the receiving process, this is information which is functionally equivalent to being able to see the future (i.e., make use of information which is available later in time). This is the same as using hindsight, for if accurate future knowledge is available, one would never make incorrect choices. This is the physical mechanism used by TCS to implement hindsight.

2.3 Relation to Other Work

Major work in AI has focused on defining relationships between differing time intervals and creating a calculus for the manipulation of these relations [1, 2, 3, 14, 15], the use of constraint propagation techniques to narrow ambiguous bounds on temporal statements [13, 17]. TCS does not examine these issues. Instead, it assumes that the time data is available and the extent of any periods of validity can be calculated exactly.

Medical applications have included using a time oriented database to search for causal relationships [4], the interpretation of clinical data using temporal models [11], and VM [9], a program for monitoring the transition between states in mechanical ventilator therapy. Of these programs, the most relevant is the work on VM. It evaluates data over time, including trends. VM's main drawback is the inability to revise conclusions when information does not arrive in a timely manner. It assumes that all data is either quickly available, or else irrelevant to the decision making process.

Tcs eliminates this shortcoming by adding the ability to revise past information and handle data that arrives out of chronological order. Tcs also provides a more flexible format for the specification of decision procedures than the rule-based form used in VM.

The TCS updating strategy, incremental update, differs from that used in the MYCIN program [22] (complete recalculation), but is similar to that used by the Digitalis Advisor [23]. When extending a program beyond the single consultation setting of MYCIN, the overhead of doing an incremental update is saved by only changing that part of the reasoning affected by the change in the data. It is similar to the work done in Truth Maintenance Systems (TMS) [5, 7, 8, 18]. A TMS does not include a temporal component, so it cannot accord time special handling. In particular, the temporal argument that is used to generate reasoning by hindsight cannot be made in those systems. Furthermore, they impose a restriction on the type of calculation that is done in combining information.[2] Dean and McDermott [6], have extended a TMS system to add time, but their Time Map Manager (TMM) still retains the predicate calculus representation and inference methods. Although the TMM allows the greater flexibility of inexact time periods, it is not logically complete and imposes restrictions on the extent of time periods in order to make the updating algorithms computationally tractable.

3 Example

The technical discussion of the computer implementation will be based on an abstraction of an actual case from cardiology. The chosen case shows the revision of diagnosis and the modification of therapy in response to evolving information about the patient's condition.

3.1 Clinical Example

Consider the following case of a woman presenting with a heart attack and ventricular premature beats (VPBs):

[2] This is not an inherent limitation, but the TMS is derived from predicate logic and tends to restrict the reasoning to logical connectives and implications rather than allowing arbitrary calculations.

> The patient was a 56 year old female with acute chest pain, ice cold hands, clammy skin, bibasilar rales, left s3 gallop, no murmurs, blood pressure of 80/50 by cuff, pO_2 of 64 (slightly low), pCO_2 of 36 (a bit low, reflecting hyperventilation), pH of 7.36, BUN of 19, serum creatinine 1.1 and K 4.9. The ECG showed multifocal VPBs, short runs of ventricular tachycardia of 3–8 beats at a rate of 130–160, ST elevated in V1–5 (suggesting a fairly large anterior wall infarct), and no Q waves. She was treated with dopamine and lidocaine. She was excreting some urine but was oliguric ($<$ 500cc/day). After some hours a Swan-line was inserted, showing a PA pressure of 50/30 and a wedge of 29, confirming the left ventricular failure.
>
> There was limited response to the lidocaine or dopamine after a day. The blood pressure only went up to 90/50 and her hands remained ice cold and the S3 gallop and bibasilar rales persisted. The arrhythmias improved, but multifocal VPBs and short runs of ventricular tachycardia still persisted. An arterial line was put in and the blood pressure was 200/120. [12, p. 34]

This patient's reaction to her cardiac problem was to shut down blood flow to the extremities to such a degree that the blood pressure reading obtained by using a cuff on the arm was no longer representative of the true blood pressure in the core of the body. This violates the basic (unstated) assumption of blood pressure measurement: the pressure in the upper arm is an accurate indicator of the blood pressure in the aorta. Nevertheless, the measurement results were not outside the range of plausible measurements in a heart attack victim. An important ramification of this fact is that simple data consistency checking cannot detect this mistake. It is only apparent that something is amiss over the *course of the next day*.

In this case, the initial treatment was correct for someone in cardiogenic shock (a low blood pressure state). The problem first became apparent when the expectations of therapeutic response were violated. The expected reaction to the dopamine would be a rise in blood pressure, increase in urine flow and an improvement in the heart failure. These effects did not occur. At this point it was necessary to reassess all the available information and the decisions based upon that information.

Based on the initial data, the therapy decision was correct. The inconsistent reaction to therapy forces one to reconsider the validity of the input data, the assumptions underlying that evaluation, or the therapy decision. Since the therapy decision was correct, based on information available at the time, the focus must be on the data evaluation. The equivalence between the

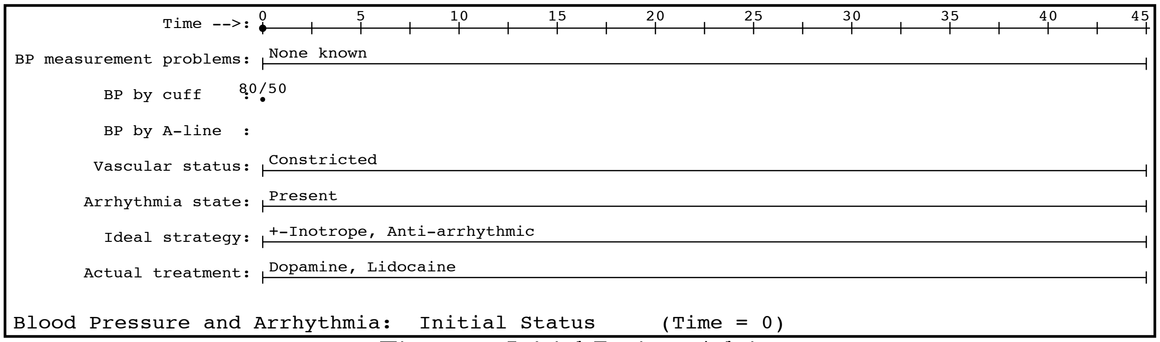


Figure 2: Initial Patient Advice

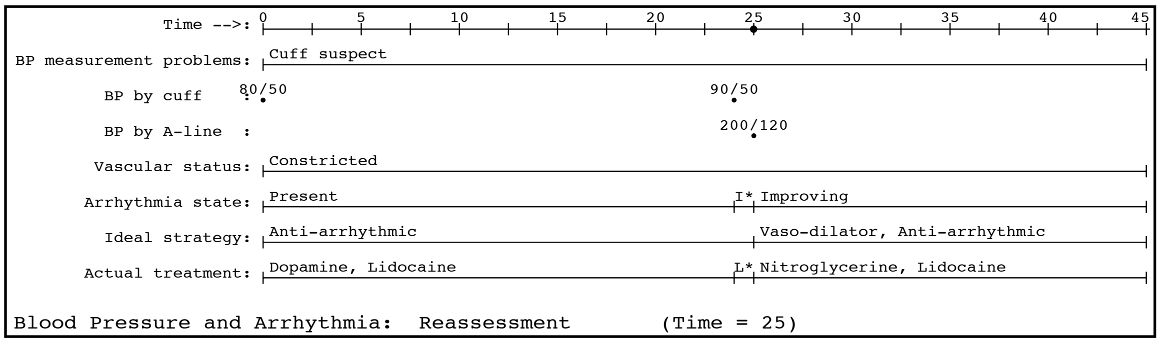
measurement of the blood pressure at the arm and the underlying datum of interest, central arterial pressure, was no longer present. This involves a reconsideration of the data evaluation. The revised opinion, benefiting from hindsight, is that a drug to vasodilate the patient should have been used instead of one to make the heart beat more strongly.

3.2 Program Results

To demonstrate the use of the TCS, a simplified version of the cardiac management decision above was programmed. Selected portions of the program output after the initial information (Figure 2) and after the hindsight was completed (Figure 3) are shown. For this example, the time scale uses units of hours.

The program takes the clinical observations and test data as its input and abstracts this to a description of the state of the patient. The initial decision uses the low blood pressure measurement (80/50), the constricted vascular status (from cold, clammy skin), and the presence of arrhythmias to suggest the use of a positive inotrope and an anti-arrhythmic agent. This abstract strategy is refined into the concrete recommendation of dopamine and lidocaine (Figure 2). The transformation of point data to intervals in the TCS is discussed in greater detail in [16].

As more information becomes available (at time 24 and 25), the assessment is reconsidered. The arrhythmia remains a problem, but since it is improving, the program concludes that the choice of lidocaine is correct and should be continued. This is reflected in the retention of the



(The "I←" and "L←" stand for "Improving" and "Lidocaine." There was not enough room on the graph for the full labels.)

Figure 3: Revised Patient Advice

anti-arrhythmic part of the ideal therapy strategy. Since the lack of blood pressure response is not consistent with the expected effects of dopamine, this part of the case analysis needs to be reexamined. The lack of response, combined with the vasoconstriction makes the cuff method of blood pressure measurement suspect, which is detected by a module monitoring the progress of therapy. Without a reliable blood pressure, the justification for the positive inotrope is missing, so it is removed from the ideal strategy.[3] The concrete treatment, however, can only be changed in the future, so dopamine remains on the treatment list for the first 24 hours (Figure 3). Once the arterial line is inserted and a reading obtained, a vasodilator is indicated to reduce the central blood pressure from its very high level of 200/120. This is reflected in the concrete suggestion that nitroglycerine be added.

In the implementation of this decision, the module used to evaluate the data to arrive at a treatment strategy considers the current values of the blood pressure, the arrhythmia state and the vascular status of the patient, as well as any known problems with drugs or blood pressure. The module that determines if there are any drug or blood pressure problems considers the treatment and current (input) and past (history) values of the clinical parameters. It also uses the history

[3] It would also be possible to implement a less radical strategy by suggesting the use of an arterial line before the therapy itself was changed. This is a change in the function used by the decision module and doesn't affect the demonstration of the action of TCS.

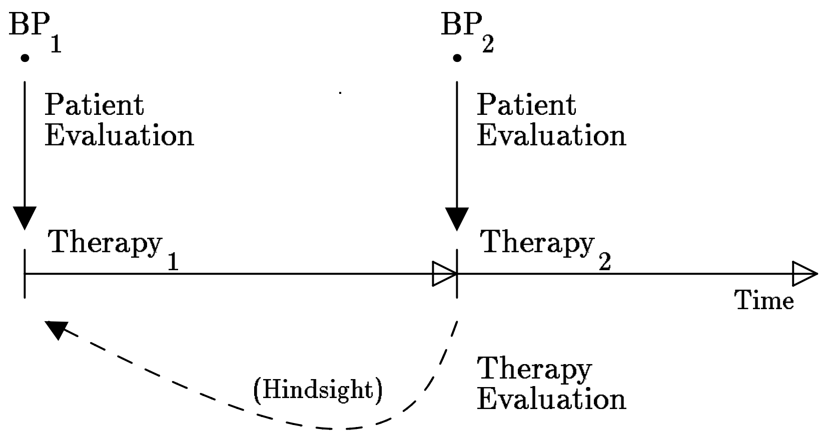


Figure 4: Temporal Aspect of Hindsight

and oracle facility to make the conclusions about the blood pressure difficulties available to earlier and later time periods. The detection of the problem at time 24 is therefore available for use in reconsidering the treatment strategy at time 0.

The program used to generate the output shown here consisted of 26 variables (of which only seven are shown in the figures) and 14 reasoning modules. The initial evaluation executed 38 process instances for the 14 modules. The two stages of the revision at times 24 and 25 combined executed 85 process instances.

4 Discussion

Hindsight is inherently a temporal process. It involves using data available at one time to evaluate decisions made earlier. The temporal aspect of reasoning by hindsight is illustrated in Figure 4. The initial advice for Therapy_1 is based on the evaluation of the first blood pressure reading BP_1. After some time has passed, another reading is considered (BP_2). This second reading is used in two ways:

1. To evaluate the efficacy of the initial intervention Therapy_1 as well as the process (Patient

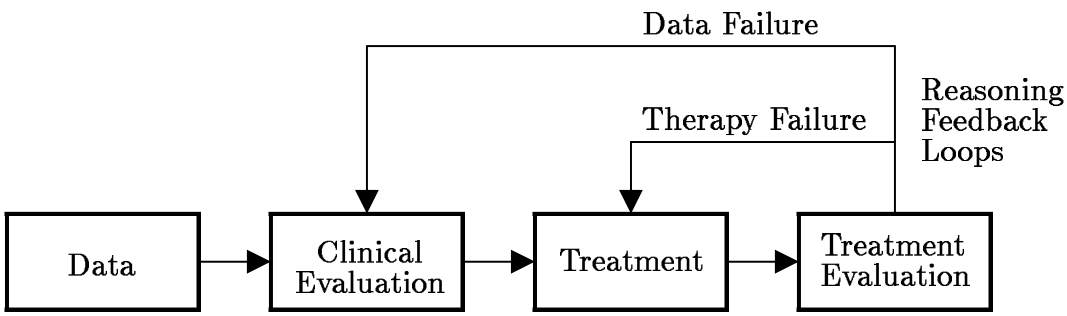


Figure 5: Evaluation Feedback Loops

Evaluation) that led to the choice of that therapy. This review can either confirm the correctness of the initial decision, suggest a modification of the therapy, change it completely or be neutral (i.e., not express an opinion).

2. To plan future therapeutic interventions (Therapy$_2$). The future plans will implement either the results of the review of therapy from 1, above or else involve some other changes, perhaps the progression to a new state of the therapy.

In the following sections I examine conceptual issues raised by the use of feedback as well as technical considerations needed for the proper implementation of this reasoning.

4.1 Evaluation Feedback

The use of new information for the evaluation of a treatment can detect two different types of failure. One is the failure in the choice of therapy. This could be due to an error in the reasoning which led to the choice of therapy, or it could be due to inherent uncertainty. An example of the latter would be the presence of arrhythmias that are lidocaine resistant. In the above example, if the lidocaine had proved incapable of improving the arrhythmia, the program would conclude that the arrhythmia is resistant to lidocaine. However, since the anti-arrhythmic strategy would still be correct, only the implementation of the strategy need be changed. An appropriate alternate drug such as procainamide would be suggested.

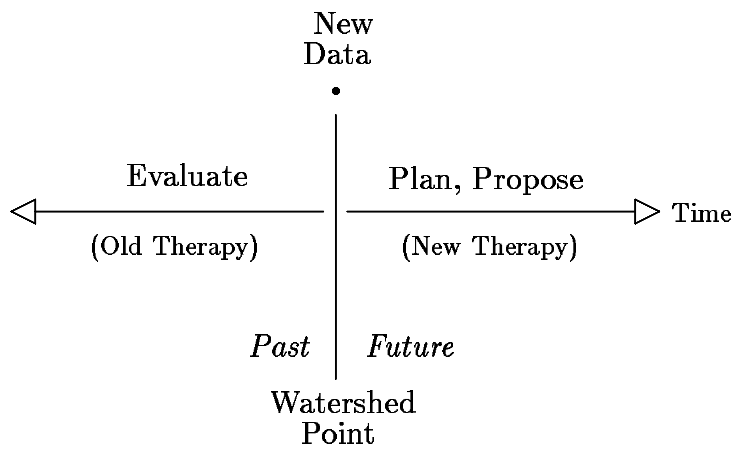


Figure 6: Use of Data at Watershed Points

A second form of failure that could be detected by hindsight could be an error in the data collection process or in the interpretation of the data in a particular patient's context. The example from Section 3 is an error in the data collection due to a violated assumption. An example inappropriate data interpretation could occur if a normally hypertensive patient presented with a blood pressure of 115/75. In most patients this would be considered in the normal range. For a hypertensive individual, however, this should be considered low and a cause for hypotension sought. If the history of hypertension became available only after the start of treatment, then the reinterpretation of the blood pressure readings would be an example of hindsight.

Both of these types of feedback in the reasoning are shown in Figure 5.

4.2 Watershed Points

Data that arrives can be used for two distinct purposes. It can evaluate therapy that has already been carried out, and it can serve as the basis for new or modified therapy to be given in the future. This split in function is along the past-future line.

The need to maintain the past and future distinction requires some technical care in the interpretation of point data that arrives. Since actions that depend on that data can only take place after it arrives, data can provide a justification for plans and proposed actions beginning with the time of arrival. At the same time, the data can be used to evaluate the past sequence of actions. In this manner the time that the data arrive forms a *watershed* in the type of reasoning. Since it can be an endpoint of a reasoning interval, the time of the datum naturally serves as a dividing point between these two functions. This is illustrated in Figure 6. Any information that arrives can be used for the evaluation of actions preceding the time of the data, since by the temporal ordering of cause and effect, only those actions could have had an influence on the value. The effect of actions occurring after the time of the data sample cannot be evaluated based on that particular value.

Similarly, the data can be used as a justification for taking only those actions which begin after the sample has been received. Because of the temporal order on cause and effect, it would make no sense to use the datum for the evaluation of the effects of future actions. It can, however, serve as a reason for wanting to perform those actions. The data can only serve as a cause for the mental desire to perform an action. They can not be a justification as a result of an action.

4.3 Past–Future Distinction

When considering the way reasoning and actions interact in an advice-giving system, one must maintain a separation between reasoning about future events and reasoning about past events. In the future, one can freely change both the advice and the actions that follow from the advice. In the past one can, through hindsight, change the advice—deciding "what *should* have been done"—but actions must remain unchanged, reflecting what was actually done.

In a program, this can be accomplished by maintaining separate variables for the advice (the ideal treatment and strategy) and the actual interventions (the concrete treatment and strategy). This is combined with a system maintained variable indicating whether the reasoning is in the past or the future. If reasoning in the past, no changes are allowed to the concrete choices. One could accomplish the same end by having the concrete actions be entered from outside the system. This would also be a confirmation of what was actually done, since the clinical staff is not forced to follow the advice of the computer.

In addition to being a logical nicety, maintaining this distinction is crucial to the performance of reasoning by hindsight. The importance lies in the interaction between changing items in the past (through hindsight) and the dependency directed updating system. Without taking care to make this distinction, one can be led into a circular argument of the following form:

1. We have a default assumption $\mathcal{A}$.
2. We believe datum $\mathcal{D}$, based on assumption $\mathcal{A}$.
3. $\mathcal{D}$ indicates that the proper therapy is $\mathcal{T}$.
4. Treatment $\mathcal{T}$ leads to response $\mathcal{R}$, when $\mathcal{D}$ (using assumption $\mathcal{A}$) is present.
5. When we later discover that $\mathcal{R}$ did not occur following $\mathcal{T}$, we can conclude that assumption $\mathcal{A}$ is invalid and should be retracted.
6. Since $\mathcal{D}$ depends on $\mathcal{A}$, $\mathcal{D}$ is retracted.
7. Since $\mathcal{D}$ is retracted, we have no longer have a reason for doing $\mathcal{T}$, so it is removed.
8. Without $\mathcal{T}$, the absence of the response $\mathcal{R}$ alone is not sufficient grounds for disbelieving $\mathcal{A}$.
9. Therefore, we can make $\mathcal{A}$ as the default assumption and the cycle begins again at step 1.

We get into trouble at step 7, where an attempt is made to undo a past action. If this step is disallowed, then the circularity is broken and the reasoning chain remains valid, without the infinite loop. $\mathcal{T}$ must be removed from the list of actions that we wanted to perform (in the ideal world), while remaining on the list of actions that were performed (in the real world).

This potential circularity requires that we keep those items of the history invariant which cannot be changed retroactively. Since we do not have real oracles, data is only available at the current time or from past times. Hindsight cannot undo physical actions. They must remain, not only for philosophical reasons, but also because of their logical necessity in support of the hindsight argument.

Histories can also be used to implement *blackout periods*, so that information that arrives immediately after a therapeutic step has been instituted does not cause the reasoning that was just done to become invalidated. What this means is that each change in therapy must be given an appropriate period of time in which to act before it becomes liable to criticism for failure. For example, one would not want to change the dose or type of diuretic if urine output did not rise within 5 minutes of administration. The only difference is that the variable performs a control function rather than a data function. In this way the TCS permits temporal considerations to control the chain of reasoning as well as to react to data input.

5 Conclusion

I have identified reasoning by hindsight as a useful mechanism to exploit the additional information that a temporal sequence of events can give to the evaluation of past decisions. Reasoning that moves backwards and forwards along the temporal axis requires that dependencies be maintained and the grounds for beliefs and actions be recorded.

I described an architecture that can accomplish this task by making minimal assumptions about the form of the reasoning that is to take place. The only requirement is that it be possible to compute "sameness" on the values of variables that enter into a decision procedure. This is a weak restriction on the values that can be represented. The only restriction on the reasoning procedures themselves is that they have no hidden state variables and that they be deterministic. This allows a user of the shell that I have designed to specify whatever decision procedure he wishes. Common examples are if-then rules of the sort indicated here, Bayesian updating of probabilities or the calculation of values using a mathematical model. Model-based calculations are particularly interesting because of the ability to revise the output of the models as underlying assumptions change.

Some mathematical models that have been developed, for example by Jelliffe [10], are adaptive in the sense that information provided by testing the predicted variable can be used to refine and customize the model parameters. By making such data available through the history/oracle mechanism, the outputs of those models can be changed and the changes propagated through the reasoning network. For example, assume one has a model to predict serum drug concentration following therapy (a pharmacokinetic model). A decision to change anti-arrhythmic drugs based on insufficient response might be modified if laboratory tests indicated that a pharmacokinetic model's general population parameters predicted a higher serum level than was present in the patient being treated. This ability to revise the basis of the model and update the conclusions that depend on it makes the system robust enough to deal with the uncertain and inaccurate data of real world clinical situations.

Implementation Note

TCS is implemented using Symbolics Common Lisp and runs under release 7.2 of the Symbolics system. The graphs in Figures 2 and 3 were produced by a program running under TCS version 26.0, implementing a simplified decision procedure for the example case presented in Section 3. The decision program presented in this paper was created to illustrate the phenomenon of reasoning by hindsight and is not intended to be a complete expert system for cardiac intensive care. Copies of the code for the procedure and for the TCS are available from the author.

Acknowledgments

I would like to thank Dr. Michael Klein from Boston University–University Hospital for providing the example case used in this paper. Dr. Michael Hagen from Tufts New England Medical Center and Dr. Reid Rubsamen from Massachusetts General Hospital gave advice about the medical content of the example. In spite of their assistance, any errors in the medicine are my own. I would also like to extend thanks to William Long who has been an invaluable sounding board for ideas relating to the design of TCS.

6 References

[1] James F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, November 1983.

[2] James F. Allen. Towards a general theory of action and time. *Artificial Intelligence*, 23:123–154, 1984.

[3] James F. Allen and Patrick J. Hayes. A common-sense theory of time. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 528–531, 1985.

[4] Robert L. Blum. *Discovery and Representation of Causal Relationships from a Large Time-Oriented Clinical Database: The* RX *Project*, volume 19 of *Lecture Notes in Medical Informatics*. Springer-Verlag, 1982.

[5] Johan de Kleer. An assumption-based TMS. *Artificial Intelligence*, 28:127–162, 1986.

[6] Thomas L. Dean and Drew V. McDermott. Temporal data base management. *Artificial Intelligence*, 32:1–55, 1987.

[7] Jon Doyle. A truth maintenance system. *Artificial Intelligence*, 12(2):231–272, 1979.

[8] Jon Doyle. A model for deliberation, action, and introspection. AI-TR 581, Massachusetts Institute of Technology, Artificial Intelligence Laboratory, 545 Technology Square, Cambridge, MA, 02139, 1980.

[9] Lawrence Marvin Fagan. VM*: Representing Time-Dependent Relations in a Medical Setting*. PhD thesis, Stanford University, June 1980.

[10] Roger W. Jelliffe. Clinical applications of pharmacokinetics and adaptive control. *IEEE Transactions on Biomedical Engineering*, BME-34(8):624–632, August 1987.

[11] Michael Gary Kahn. *Model-Based Interpretation of Time-Ordered Medical Data*. PhD thesis, University of California, San Francisco, December 1988.

[12] Michael D. Klein. Development of a ventricular arrhythmia management advisor. National Institutes of Health Research Grant Application, June 1983.

[13] Isaac S. Kohane. Temporal reasoning in medical expert systems. In Salamon et al. [21], pages 170–174.

[14] Peter Ladkin. Primitives and units for time specification. In *Proceedings of the National Conference on Artificial Intelligence*, pages 354–359. American Association for Artificial Intelligence, 1986.

[15] Peter Ladkin. Time representation: A taxonomy of interval relations. In *Proceedings of the National Conference on Artificial Intelligence*, pages 360–366. American Association for Artificial Intelligence, 1986.

[16] William J. Long and Thomas A. Russ. A control structure for time dependent reasoning. In *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, pages 230–232, 1983.

[17] J. Malik and T. Binford. Reasoning in time and space. In *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, pages 343–345, 1983.

[18] David Allen McAllester. A three valued truth maintenance system. AIM 473, Massachusetts Institute of Technology, Artificial Intelligence Laboratory, 545 Technology Square, Cambridge, MA, 02139, 1978.

[19] Thomas A. Russ. A system for using time dependent data in patient management. In Salamon et al. [21], pages 165–169.

[20] Thomas A. Russ. Temporal control structure reference manual. TM 331, Massachusetts Institute of Technology, Laboratory for Computer Science, 545 Technology Square, Cambridge, MA, 02139, June 1987.

[21] R. Salamon, B. Blum, and M. Jørgensen, editors. *MEDINFO 86: Proceedings of the Fifth Conference on Medical Informatics*, Washington, October 1986. North-Holland.

[22] Edward H. Shortliffe. *MYCIN: Computer-based Medical Consultations*. American Elsevier, New York, 1976.

[23] William R. Swartout. Producing explanations and justifications of expert consulting programs. TR 251, Massachusetts Institute of Technology, Laboratory for Computer Science, 545 Technology Square, Cambridge, MA, 02139, January 1981.