COORDINATING ADVICE AND ACTUAL TREATMENT

Thomas A. Russ

Massachusetts Institute of Technology
Laboratory for Computer Science
545 Technology Square Room 411
Cambridge, MA 02139–1986
tar@lcs.mit.edu

Abstract

In patient management, the challenge facing a knowledge-based system includes tracking the changes in the patient's condition over time and responding with appropriate advice. In addition to monitoring the patient, it is also necessary to follow the course of the actual treatment that is being administered. Any changes to the therapy must start with an understanding of the current state of the treatment. Since expert systems will not have direct control of patient treatment, they must coordinate their advice offered with the treatment that is actually given. This paper describes how the current time, blackout periods and agendas can solve therapy coordination problems.

Introduction

Management differs from the single consultation model of expert systems in making multiple decisions over a period of time. This temporal component of the decision making presents several challenges. In this paper I will focus on three issues: Identifying the current treatment, allowing time for therapy to take effect, and coordinating the advice with the actual execution of treatment plans. These problems were encountered during the implementation of an expert system for treatment of diabetic ketoacidosis (DKA).

Choosing the Source of Information

During the course of therapy planning and evaluation, it is important to keep track of the patient's treatment. Information about the treatment can come from one of three sources—advice from the program, clinical orders and records of actual treatment. At different times each of these information sources must be used in reasoning about therapeutic recommendations.

Advice that the expert system generates is easily accessible to the reasoning program, but is limited in usefulness because the advice will not always be followed. When reasoning about the future, once can expect that the program's advice will be followed. Using this assumption, complete care plans can be generated. It is also the only source of information about future events.

When reasoning about the past, however, the expert system is unable to change what actually happened. It must base its reasoning on the real clinical actions. The best source of information about treatment is the record of the fluid and drug administration given by the clinical staff. This is the most definitive record and is always used when available. Unfortunately, this information is not always immediately available. For example, the exact fluid infused via an intravenous line can only be determined after someone checks the infusion bottle to determine how much fluid was actually used. Since a one liter bottle flowing at 250ml/hr will take four hours to empty, there can be a substantial delay until definitive data are available.

To cover this intermediate time period, the system can use the information contained in the medical orders. This describes the intended treatment, but it can differ in timing and exact amount from what is actually done. For example, an infusion order might call for normal saline at 250ml/hr whereas the actual infusion rate may turn out to be 220ml/hr. Since the definitive results take longer to be available, a temporal order is imposed on the desirability of the input source. This order is illustrated in Figure 1

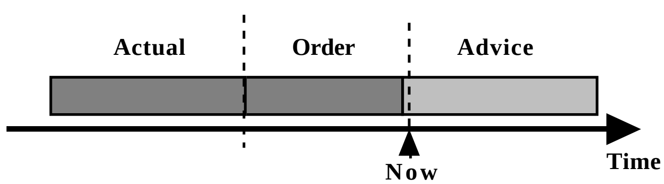

Figure 1. The sources of data in temporal order.

Allowing Therapy Time to Act

A further temporal consideration is the need to allow therapies time to act before assessing their effectiveness. Depending on the type of preparation, subcutaneous insulin injections have an onset of action from 15 minutes to 8 hours [4, p. 321]. A laboratory blood sugar measurement that leads one to give a supplemental insulin injection cannot be evaluated until enough time has passed for the drug to have an effect. High blood sugar seen in the interim should not cause the program to recommend another injection until sufficient time has elapsed to make this judgement.

Coordinating Therapeutic Advice and Actual Treatment

The third complication involves the delay between the posting of advice and the implementation of the advice by clinical personnel. For a variety of reasons, advice produced by an expert system may not be acted upon immediately. For example, more urgent chores may require the attention of the health care providers. It is important, then, for the expert system to retain the advice in some form until it is acted upon. At the same time, it must recognize when actions that have been recommended have been carried out.

The work reported here has been supported by National Institutes of Health grant R01 HL 33041 from the National Heart, Lung, and Blood Institute.

This need to retain information for later action also appears when external events trigger the application of therapy. In treating DKA, supplements of potassium and bicarbonate are often recommended [4], and are given in the form of additions to intravenous bottles. Such additions can only be practically given at times when a new bottle is started. Since the emptying of bottles is often not coordinated with the availability of laboratory test results guiding treatment, the results of the recommendation must be retained until the appropriate triggering events occurs.

Implementing the Solutions

I have implemented solutions to these problems. The specific solutions are built on a substrate that automatically handles the scheduling of reasoning in response to data availability.

Maintenance of a Temporal Database

The programming system is called the Temporal Control Structure (TCS). It is a temporal data dependency manager described in greater detail in [6]. For the purposes of this paper, it is sufficient to know that the system maintains a temporal database and automatically updates decisions that are based on information that changes.

Data can be stored either as point events or as intervals. When an expert system is programmed, the data dependencies of all of the decisions are declared. By tracing the dependency structure, the TCS can assure the complete propagation of information as it arrives and changes. By relieving the programmer of the burden of explicitly calling for the recalculation of affected decisions, the effort involved is reduced and the reliability of the system is enhanced.

Choosing the Data Source

A simple rule for the choice of data source is shown in Figure 2. This code fragment defines a rule which takes the three data sources as input, along with a system-defined variable that indicates whether the reasoning is in the future or not. The value of the treatment is set to be the same as the advice in the future, and either the actual therapy or the ordered therapy in the past. As time advances and more information becomes available, the value of the treatment variable will change and the effects will be automatically propagated. The setup of the environment for rule execution and the handling of values over time is further described in [5]. Since the form of this rule closely matches the TCS model of temporal data extent, the code for the implementation is quite simple.

```
(defrule (actual order advice future?) (treatment)
   (setq treatment (cond  (future? advice)
                          ((known? actual) actual)
                          (t order))))
```

Figure 2. Program for Data Selection.

Allowing Therapy Time to Act

Coordination via blackout periods is relatively easy to arrange, because preemption of advice is well ordered temporally. Since we know about the preempting action at the time the advice would be generated, we can use that information to suppress inappropriate advice. Blackout periods are shown in the line labeled “Inhibit” in Figure 3. The actual therapy establishes a period in which advice generation in response to the incoming data is inhibited. (This does not preclude the data from being used in other analysis in a different part of the expert system.) This is an instance of a more general phenomenon that involves assigning persistence to point events used in reasoning. A more detailed description of the use of TCS for this type of reasoning can be found in [3].

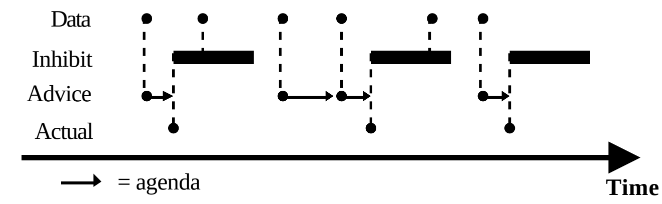

Figure 3. Immediate agenda with blackout periods.

Coordinating Therapeutic Advice and Actual Treatment

Figure 3 also shows the temporal extent of agenda items. Advice generated in response to the data in recorded on an agenda that lasts until the advice is carried out or superseded by different advice in response to newer information. Generating the advice as an immediate consequence of the data values and then recording it separately on an agenda provides a modular separation between the temporal relationships used in this reasoning. The direct cause and effect between the data and the advice allows a simple propagation of new data as well as corrections to previously entered data. The responsibility for maintaining the extent of the advice over time and recognizing when actions have been carried out is left to the attention of a specialized agenda maintenance module.

The idea of an pending agenda can be extended to handle the case of items awaiting a particular trigger. The example from DKA involves the choice of potassium and bicarbonate supplements to the infusion fluids. In this instance, the agenda describes the advice for the fluid composition at the next trigger point, that is, when the bottle becomes empty and a new bottle is started. This is illustrated in Figure 4. Rather than being direct advice, the infusion agenda is triggered to produce advice at a time when it is practical to act on it. This advice can then be added to an agenda of the first type, since the clinical staff may not be able to start a new bottle when the old one is empty.

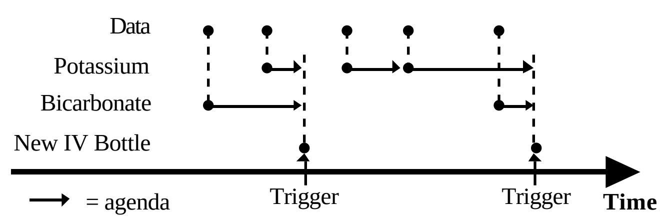

Figure 4. Agenda with external trigger.

Conclusion

The use of the Temporal Control Structure allows the flexible implementation of reasoning systems that manipulate data that arrives during the course of the consultation. The dependency-directed updating guarantees the complete propagation of new information and changes in conclusions throughout the system while relieving the programmer of the necessity of doing these chores himself.

The system differs from other medical monitoring systems by the use of an extensive system for updating the information and handling data that does not arrive in chronological order. Fagan's Ventilation Manager [1], for example, operated in a domain in which all data arrived quickly and thus did not require a complicated updating scheme. Kahn's used a multiple-model approach to controlling cancer therapy [2] was an outpatient monitoring system and did not need to handle data arriving out of order.

In an inpatient setting, it is important to be able to revise the conclusions while in the middle of handling the case. I described three techniques that solve problems associated with the data management and coordination of the machine-generated advice and the actions of the clinical staff. Data source selection and blackout times are easily implemented in the TCS formalism. The use of the agenda mechanism provides a powerful abstraction that simplifies the task of constructing medical monitoring systems.

Acknowledgements

Thanks to my medical collaborator Klemens Meyer and his predecessor Michael Hagen from the Tufts New England Medical Center. They were invaluable resources for the medical content of the expert system. Any mistakes in the medicine in this paper are my own.

Implementation Note

TCS is currently implemented in CommonLisp and Flavors on Symbolics Lisp Machines. A preliminary version using CommonLisp and CLOS has undergone preliminary testing using Macintosh Allegro Common Lisp on Macintosh IIs and Lucid CommonLisp on Sun workstations.

References

[1] L. M. Fagan, VM: Representing Time-Dependent Relations in a Medical Setting, PhD thesis, Stanford University, June 1980.

[2] M. G. Kahn, Model-Based Interpretation of Time-Ordered Medical Data, PhD thesis, University of California, San Francisco, December 1988.

[3] W. J. Long and T. A. Russ, "A Control Structure for Time Dependent Reasoning," in Proceedings of the Eighth International Conference on Artificial Intelligence, 1983, pp. 230–232.

[4] M. J. Orland and R. J. Saltman, eds., Manual of Medical Therapeutics, 25th edition, Boston: Little, Brown and Company, 1986.

[5] T. A. Russ, "A System for Using Time Dependent Data in Patient Management," in MEDINFO 86: Proceedings of the Fifth Conference on Medical Informatics, 1986, pp. 165–169.

[6] T. A. Russ, Temporal Control Structure Reference Manual, Massachusetts Institute of Technology, Laboratory for Computer Science Technical Manual, MIT-LCS-TM-331, June 1987.