Artificial Intelligence in Medicine 3 (1991) 313–323
Elsevier

Automatically analyzing a steadily beating ventricle's iterative behavior over time*

Alexander Yeh

MITRE Corporation, Burlington Road, Bedford, MA 01730, USA

Received March 1991
Revised July 1991

Abstract

Yeh, A., Automatically analyzing a steadily beating ventricle's iterative behavior over time, Artificial Intelligence in Medicine 3 (1991) 313–323.

The full analysis of complicated dynamic systems has yet to be automated. Fortunately, a steady-state analysis is often still quite useful. This paper describes AIS, a program written to analyse (sub)systems that at steady-state are iterating a fixed sequence of actions. Such systems are too complicated to directly describe using something like electrical circuit analogies. An example of such a system is the ventricle of the heart. AIS takes in a system description in terms of its detailed behavior during the time scale of a single sequence iteration (such as a single heart beat), and outputs a system description in terms of the system's net behavior during the time scale of many iterations. Single iteration descriptions are often easier for people to give, but average behavior descriptions are often more useful.

Keywords. Steady-state system; temporal reasoning; heart.

1. Introduction

The ability to automatically analyze dynamic systems would help with analyzing and diagnosing problems in the human circulatory and other similar systems. At present, this ability is quite limited for systems as complicated as the circulatory system, or even a part of such a system like a ventricle of the heart.

Often, just being able to analyze a system's steady-state behavior is useful. In addition, in many cases, while examining a steady-state, one is only interested in system variables that maintain a steady value or versions of variables (the average or extreme values, or the rate of change, etc.) that are steady. With this narrowing of the scope, one can build a model to analyze by finding the simultaneous time-invariant relationships that hold between the variables and versions of variables that maintain steady values.

One way to model the simultaneous relationships is to use direct current electrical circuit analogies. In the case of the circulatory system, an example of such a relationship would be

* This paper reports on research performed at the Clinical Decision Making Group (CDMG) of the MIT Laboratory for Computer Science. The research was supported by the National Institute of Health through grant R01-LM04493 from the National Library of Medicine, and grant R01-HL33041 from the National Heart, Lung, and Blood Institute. Various members of CMDG have aided this research, especially Jon Doyle, William Long, Elisha Sacks, and Peter Szolovits. Also the anonymous referees are acknowledged for their constructive criticisms of an earlier version of the paper.

[pressure drop] = [resistance] · [flow], where [pressure drop] and [flow] are both averages over several heart beats at steady state. Unfortunately, some relationships are too complex to be modeled this way. For example, in the circulatory system it is hard to directly describe the relationship between a ventricle's average input and output blood pressures, heart rate, and pressure versus volume curves to the average rate of blood flow through that ventricle.

Fortunately, many such hard to model relationships concern subsystems that like the ventricle have the following two properties:

1. they carry out a series of actions or transformations that have the following 'constancy' (invariant property over time): the series always repeats the same sequence of actions in the same order and each occurrence of a particular action always changes the parameters by the same amounts, and

2. the subsystems iterate the sequence at such a rate that the other parts of the system respond only to the behavior of such subsystems averaged over many iterations.

With such subsystems, an easier method of describing what happens to them may be to give the sequence and the relationships that hold in each part of the sequence. With the ventricle example, let P and V be the ventricle's pressure and volume respectively, Bi be the amount of blood that has moved into the ventricle, Bo be the amount that has moved out, Pi be the pressure of the entering blood, Po be the pressure of the exiting blood. $Vd[Pi]$ (a function of Pi) be the amount of blood in the ventricle when it is relaxed, and $Vs[Po, HR]$ be the amount of blood in the ventricle when it is squeezing as hard as it can. HR is the rate at which the ventricle beats. Then part of such a description may be as follows: The sequence is that

1. The ventricle squeezes the blood in it without releasing any: V, Bi and Bo stay the same. P changes to a value of Po.

2. The squeezing continues with blood exiting out the ventricle's output: P and Bi stay the same. V changes to a value of $Vs[Po, HR]$. Bo increases by the opposite of the change in V's values.

3. The ventricle relaxes: V, Bi and Bo stay the same. P changes to a value of Pi.

4. The relaxation continues with blood entering via the ventricle's input: P and Bo stay the same. V changes to a value of $Vd[Pi]$. Bi increases by the change in V's value.

While such a description is easier to give, it alone does not complete the model. To complete the model, one needs to derive from that description the time-invariant parameter relationships at steady-state. In the case of the ventricle, two of the relationships are (dX/dt is the *average* rate of change in parameter X):

$$d(Bi)/dt = d(Bo)/dt = HR \cdot (Vd[Pi] - Vs[Po, HR]) ,$$

where the symbols are as described above. This paper describes AIS (short for Analyzer of Iterated Sequences), a program to automatically perform such derivations, taking the easier to give 'single iteration' description and converting it to the more useful 'average behavior' description. AIS also performs sensitivity analysis on how such relationships would be different if various constants had different values. This analysis can be useful both in getting a feel for how the relationships behave and also in giving something to compare against experiments done on the system of interest.

An alternative way to analyze a repetitive subsystem is to characterize all the forces in the system and the parameters that they affect. Then, analyze the description with a method that combines qualitative simulation with cycle detection [2]. For complicated systems, these methods predict many possible sequences of actions besides the actual sequence. *Aggregation* [16] and *computer analysis* [17, 18] are useful after isolating the actual sequence.

Another way to perform such an analysis is to

1. describe the system with a single set of differential equations that is always applicable, and then

2. use the program described in [12] to analyze the set.

However, coming up with such a description for a complicated system is quite difficult. In contrast, the input for both the qualitative simulation approaches and AIS can have many sets of simple equations along with the conditions to determine when a particular set is applicable.

Examples of this difficulty can be found in some attempts to model the ventricles using differential equations [9, 6]. In [9], instead of using a single set of equations that is always applicable, the authors use one set of auxiliary variables and equations for modeling a ventricle's contraction and another set for relaxation. In [6], a ventricle's contraction and relaxation are modeled by elastance/capacitance versus time graphs rather than differential equations. With enough additional auxiliary variables, functions like step functions,[1] and additional equations, we could probably model ventricular contraction and relaxation using a single set of differential equations. But the resulting model will be large, and hard to derive, to comprehend and to reason about.

The rest of this paper is as follows: The next section describes AIS. Following this is a section on AIS running on the ventricle example, a section on possible future work, and a summary. Other examples (a ventricle with mitral stenosis and a steam engine) of using AIS can be found in [19], as well as more details on AIS itself and the example given in this paper.

2. Description of AIS

This section describes AIS, the program used to analyze a steadily beating ventricle. In order, the subsections describe the input for AIS and AIS's output.

2.1 Input

An input description consists of three parts: the parameters which describe the system state, static conditions on those parameters, and the sequence of transformations (actions) that gets iterated. The description only gives what happens in the sequence, not how or why it occurs or repeats.

Parameters are divided by the model-builder into four types. The first three types are classified by how a parameter behaves as the sequence of actions is iterated:

1. *Constant parameters* do not change in value at all during the iterations.

2. *Periodic parameters* change in value, but the sequence of values repeats exactly with each iteration.

3. *Accumulating parameters* monotonically increase or decrease in value with each sequence iteration.

[1] For a step function $u(t)$, $u(t) = 1$ when $g \geq 0$ and $u(t) = 0$ otherwise. This function is often used to combine expressions that are valid under different conditions.

In general, parameters are represented by symbols. The constant parameter type also includes numbers and arbitrary functions of expressions of constant parameters such as $f\left[x+3, g[5]\right]$, where x is a constant. The fourth parameter 'type' has only one parameter: the rate at which the sequence of actions is iterated. At present, the rate must be expressed as a constant parameter that is a symbol or number.

The second part of the input is a set of static conditions between constant parameters. These conditions are inequalities between numbers and expressions made up of constant parameters. The expressions can have algebraic and the more common transcendental functions. Also permissible are (partial) derivatives of constant parameters which are arbitrary functions.[2] The inequalities can be either definitions that are always true or conditions that are required for the given sequence of actions to iterate. An example of a definition is to say that some volume is at least 0. An example of a necessary condition is to say that for a normal sequence of actions in the heart, the input pressure is less than the output pressure.[3]

The last part of the input gives the sequence of actions (transformations) that is iterated. The sequence is partitioned into *phases* so that each part of a sequence is put into exactly one phase and each part where different actions are occurring is put in a separate phase. What is desired is that all the important and possible extreme parameter values appear at the end of some phase. The specific requirements are that the phases must be chosen so that

1. every part of a sequence (including all the parts with parameter value changes) is put in exactly one phase, and

2. during each phase, every parameter is either monotonically non-decreasing or non-increasing in value.

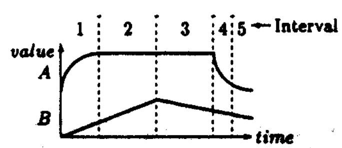

Fig. 1. Examples of possible intervals for phases.

Beyond these two requirements, a model-builder is free to divide a sequence into as few or many phases as desired. As an example, see *Fig. 1*, where the values of the parameters A and B versus time (for one iteration) are given. A model-builder may put each of the five marked intervals into a separate phase. An alternative is to have two phases, with intervals 1 and 2 in one phase and intervals 3 through 5 in the other. Other groupings of the intervals are also possible, as is dividing an interval over more than one phase. One constraint on the grouping is that if two intervals of an iteration belong to one phase, then so do all the intervals in between those two (intervals 1 and 5 count as being adjacent). Another constraint is that intervals 2 and 3 have to be in different phases because parameter B is increasing in interval 2 and decreasing in 3. For a similar reason, intervals 1 and 5 have to be in different phases.

A model-builder may violate the two requirements if the consequences are judged to be negligible.

[2]The derivative of a constant parameter here makes sense and may need to be described because: (1) the function itself is not constant, only the arguments are; and (2) one may need to describe how an argument's value being different would affect the function's 'output'.

[3]Otherwise, all the heart valves will open, letting blood flow freely through the heart.

For each phase, the input description needs to supply an expression for every parameter that changes in value during that phase. For a periodic parameter, the corresponding expression gives that parameter's value at the end of the phase.[4] For an accumulating parameter, the expression gives the change in that parameter's value each time that phase occurs. An expression may have algebraic and transcendental functions. The expression's arguments can consist of constant parameters, periodic parameters' values at the beginning or end of that phase, and/or accumulating parameters' change in values[5] each time that phase occurs.

The limitations on describing parameter changes are to assure that each occurrence of a phase alters the parameters by the same constant amount. Without some restrictions on how phases alter parameters, it will be hard to impossible for AIS to determine the effects of steadily iterating the sequence of actions. There are at least two interesting alternatives to having constant alterations. The first is a generalization of constant alterations. In the current version of AIS, a particular parameter changes by the same constant amount each time a particular phase occurs. In the generalization, what needs to stay constant will not be the amount of change, but rather the *change* in the amount of change (or an even higher order of change). The second is having the alterations form a converging series [15, Ch. 18]. Neither of these alternatives has been needed so far to model a 'steadily running' device.

It is sometimes difficult to provide expressions for the periodic parameter values at the end of a phase. For example, one might not be able to explicitly give the pressure at any point in a water pipe circuit. Unfortunately, if one provides only changes to the periodic parameter values, finding their actual values during the sequence would be impossible or hard, involving symbolically solving simultaneous (nonlinear) equations. With only changes in their value solved for, periodic parameters would be just like accumulating parameters that have a zero net change on each sequence iteration.

Each phase also has a list of the conditions that either are true by definition or need to be true for the phase to occur as stated. The conditions are inequalities between expressions and numbers. Note that the definitions of phase expressions and conditions are slightly different from the definitions given earlier for static conditions between constant parameters.

AIS makes the 'closed world' assumption that all changes are mentioned. So if some phase's description does not mention a new value for a parameter α, α is assumed not to change in value during that phase.

Here is an example of an input description for a phase. Let X_b stand for parameter X's value at the beginning of a phase, X_e for the value at the end, and X_c for X's change in value when the phase occurs. Furthermore, let a be an accumulating parameter, q and r be periodic parameters, and c be a constant parameter. The sample phase description is:

$$(5 \leq q_e)\ , \quad q_e = (c + a_c)\ , \quad a_c = (q_b \cdot r)\ .$$

Whenever this phase occurs: r's value is constant, q is at least 5 at the phase's end, a changes by the product of q's value at the phase's beginning and r's value during the phase, and q ends with c's value plus the change in a's value.

2.2 Output

AIS takes the input equations, solves them and checks for inconsistencies (using the Bounder system [11]). Bounder is also used to find the numeric bounds mentioned below.

[4] Due to the requirements on choosing phases, a periodic parameter's value at a phase's beginning and the preceding phase's end is the same. And because the sequence iterates, the last phase in the sequence is also considered to 'precede' the first phase.

[5] Only the change in value can be referred to because it stays the same from one iteration of the sequence to the next. The actual value changes with each iteration of the sequence.

Then, to derive the average rate of change in an accumulating parameter a, AIS locates the change in that parameter's value (a_c) during each phase of a sequence, adds all those changes together, and then multiplies the sum by the rate of cycle repetition. Then AIS finds numeric bounds on this rate. As an example of deriving a rate of change, let A be an accumulating parameter and R be the rate of sequence repetition. Furthermore, let two phases in this sequence alter A's value. One phase has $A_c = C$ and the other has $A_c = K$, where C and K are constant parameters. Then the average rate of change in A is $dA/dt = R \cdot (C + K)$.

After deriving an average rate for a, AIS can observe how that rate would be different if any one constant symbol or function were different. For each symbol, AIS takes the first two (symbolic) derivatives of the rate with respect to that symbol, obtains numeric bounds on those derivatives, and tries to determine which phases helped to increase or decrease each derivative. Each constant symbol is considered to be independent of all other symbols.

At present, AIS also tries to plot a 'qualitative' graph of the rate versus each constant symbol. The first derivative described above provides slope information and the second provides convexity information. AIS makes the assumption that the curve for the rate versus each constant is smooth (differentiable). If the second derivative can be both more or less than zero, AIS gives up. Otherwise, depending on how the second derivative is bounded by zero and on how the first derivative's bounds relate to zero, AIS determines which of the following shapes the curve may possibly have: ╲, —, ╱, ╰, ◡, ╯, ╭, ◠ and/or ╮. For example, if the first derivative is < 0 and the second is $= 0$ (such as when the rate is $-3x$ and the symbol is x), then the curve shape is ╲. If, however, the first derivative has no bounds while the second is < 0, then the possible shapes are ╭, ◠ or ╮. If the second derivative can be both more than and less than zero, no inferences can be made about the curve shape.

In the future, the QS system [12] can be used to perform the plotting. The advantage of QS is that it can detect complications like discontinuities and sketch curves with such complications. However, before using QS, one needs to extend it to handle functions for which derivative and smoothness information exists, but where the exact analytic form is unknown. Such functions are often used in system descriptions.

Besides deriving the effects of symbols having different values on a rate, AIS also derives the effects of functions having different values. One cannot take a derivative with respect to a function. But if one wants to observe how rates would be different if function f were larger in value, one can substitute $f(x) + e(x)$ for every occurrence of $f(x)$ in the rate (making the side assumption that $\forall x$: $[e(x) > 0]$), symbolically subtract the original rate from this altered rate, and bound the difference. If the difference is greater than 0, then if f were larger, the rate would be also, and so on.

3. The ventricle example

This section describes the current version of AIS running on a model of the beating of the part of the human heart called the left ventricle.[6] The ventricle (shown schematically in *Fig. 2*) is a chamber with two one-way valves: one valve lets in blood from the lungs at a pressure of Pi, and the other valve lets out blood going to the rest of the body at a pressure of Po. The chamber consists of muscle which can either relax or contract. When relaxed (*diastole*), the ventricle's volume (V) versus pressure (P) curve ($Vd[P]$) is roughly

[6]The description is based on various texts and articles [10, 14] [3, Ch. 13: Mechanisms of Cardiac Contraction and Relaxation] and makes many assumptions. One assumption is that blood is an incompressible fluid without inertia.

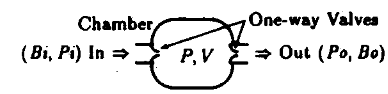

Fig. 2. Left ventricle.

as shown in *Fig. 3a* (the P and V axes are interchanged from their usual positions). When contracted (*systole*), the V versus P curve ($Vs[P, HR]$) is roughly as shown in *Fig. 3b*. The symbol HR appears because with Vs, V decreases as the rate at which the ventricle contracts and relaxes increases. This rate is known as the heart rate (HR). *Fig. 3c* shows with a dashed line the V versus P path that ventricle takes as it contracts and relaxes once (a beat sequence):

1. The ventricle contracts, but no blood moves. So, V stays the same while P increases to Po. Move from a to b in the diagram.
2. The ventricle continues contracting, but now, blood is ejected out the output valve. P stays the same while V decreases to $Vs[Po, HR]$. Move from b to c.
3. The ventricle now starts to relax and the blood movement stops. V becomes constant as P decreases to Pi. Go from c to d.
4. The ventricle continues relaxation, but now blood enters from the input valve. P stays the same while V increases to $Vd[Pi]$. Go from d back to a.

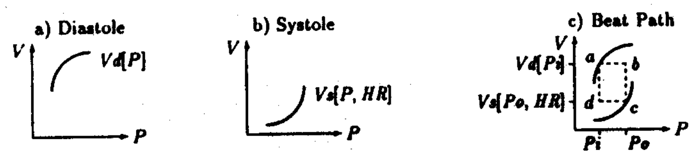

Fig. 3. Curves for a normal left ventricle.

The input to AIS has the following: The symbol HR gives the rate at which the ventricle beat sequence repeats. The constants are Pi, Po, $Vd[Pi]$ and $Vs[Po, HR]$.[7] The periodic parameters are P and V. The accumulating parameters are the amount of work done by the blood in moving through the ventricle (W), and the amount of blood that has gone into the ventricle (Bi) and out of the ventricle (Bo). The static conditions on the constants are:

$$Pi < Po, \quad Vd[Pi] > Vs[Po, HR], \quad 0 \leq Vs[Po, HR], \quad 0 \leq Vd[Pi],$$

$$0 > \partial[Po, HR])/\partial(HR), \quad 0 < \partial(Vs[Po, HR])/\partial(Po),$$

$$0 < \partial^2(Vs[Po, HR])/\partial(Po)^2, \quad 0 < d(Vd[Pi])/d(Pi), \quad 0 > d^2(Vd[Pi])/d(Pi)^2 .$$

The first two conditions ($Pi < Po$ and $Vd[Pi] > Vs[Po, HR]$) set up the proper operating conditions for a ventricle to pump blood. The rest of the conditions describe the shape of $Vd[Pi]$ and $Vs[Po, HR]$.

There are four phases in the sequence. Each phase has a name, condition(s), and equation(s) for value changes. In order, the phases are (as before, π_b and π_e stand for the periodic

[7] Pi and Po are assumed to be constant during the ventricle beats. These assumptions then force $Vd[Pi]$ and $Vs[Po, HR]$ to be also constant during the beats.

parameter π's value at the beginning and end of the phase respectively, and α_c stands for the accumulating parameter α's change in value during the phase):

1. Isovolumetric contraction: $0 \leq V$, $P_e = Po$.
2. Ejection: $0 \leq V_b$, $0 \leq V_e$, $V_e = Vs[Po, HR]$, $W_c = -P \cdot Bo_c$, $Bo_c = V_b - V_e$.
3. Isovolumetric relaxation: $0 \leq V$, $P_e = Pi$.
4. Filling: $0 \leq V_b$, $0 \leq V_e$, $V_e = Vd[Pi]$, $W_c = P \cdot Bi_c$, $Bi_c = V_e - V_b$.

After solving these equations and checking them for consistency, AIS discovers the following average rates of change for the accumulating parameters and bounds on those rates:

$$\begin{aligned} dW/dt &= HR \cdot \Big(\big(Pi \cdot (Vd[Pi] - Vs[Po, HR])\big) + \\ &\qquad \big(-Po \cdot (Vd[Pi] - Vs[Po, HR])\big)\Big) \\ d(Bi)/dt &= d(Bo)/dt = HR \cdot (Vd[Pi] - Vs[Po, HR]) > 0\,. \end{aligned} \tag{1}$$

The accumulating parameter rates were derived by summing all the changes in an accumulating parameter's value that occur in a sequence and then multiplying the sum by the rate of sequence iteration. For example, the accumulating parameter W changes in value during the *ejection* ($W_c = -Po \cdot (Vd[Pi] - Vs[Po, HR])$) and *filling* ($W_c = Pi \cdot (Vd[Pi] - Vs[Po, HR])$) phases. Sum these two changes together and multiply by HR, the rate of iteration, to get the above equation for dW/dt.

AIS's bounding is not always the tightest. For example, one can show that $dW/dt < 0$ by noting that

$$dW/dt = HR \cdot (Vd[Pi] - Vs[Po, HR]) \cdot (Pi - Po)\,,$$

which is a product of one negative and two positive values, but the bounding mechanism misses this.

After finding the rates, AIS derives and bounds the first two derivatives of those rates with respect to each constant symbol, and tries to give the shape of the curve of each rate versus each constant. For $d(Bi)/dt$, its first derivative with respect to HR is > 0, but no bounds are found for the second derivative. No curve shape is deduced. With respect to the constant Pi, the first derivative is > 0 but the second is < 0. Assuming smoothness, AIS deduces a shape for $d(Bi)/dt$ versus Pi. With respect to Po, both derivatives are < 0, so the curve has a shape. These results also apply to $d(Bo)/dt$.

For the rate dW/dt, the only bound AIS can derive is that this rate's second derivative with respect to either Pi or Po is > 0. So for dW/dt versus either Pi and Po, the possible curve shapes are , or .

As for the Vd and Vs functions, AIS deduces that if Vd were larger, both the $d(Bi)/dt$ and $d(Bo)/dt$ rates would be also. But with Vs, the opposite occurs.

As mentioned in the introduction, when modeling a circulatory system that has been averaged over many heart beats and is in steady-state, such as done in [5, 7, 14], most of the system's mechanics can be modeled with direct current electrical circuit analogies. Too complicated to be modeled this way are the relationships between the various parameters affecting the ventricles. Current modeling efforts either directly use empirically derived relationships (like [13]) or derive the needed equations by hand from an AIS-input-like description (done in [14]). AIS can perform the latter derivations automatically: Equation (1) found by AIS for $d(Bi)/dt$ provides the desired relationship for the left ventricle. The right ventricle is similar. Actually, to use Equation (1) numerically, one must have more specific Vs and Vd curves, such as specifying that $Vd[x] = \log x$.

Other than needing more specific curve shapes, the AIS $d(Bi)/dt$ equation is similar to the equations derived by others. The differences are caused by modeling with slightly different sets of assumptions and beliefs on what relationships exist and are important. Two major differences between AIS's results and some existing models [13, 14, 5] are that the latter have more specific relationships for the volume versus pressure curves than the former and that these more specific curves are also more linearized. In addition, in Sagawa's [13] and Sato et al.'s [14] ventricular models, the blood flow rate ($d(Bi)/dt = d(Bo)/dt$) is independent of the heart rate(HR), which is often quite inaccurate, especially during exercise or other times of increased venous return [3, p. 414] [10, p. 222]. Also, even when this independence is true (when a person is at rest), [10, p. 222, 294] attributes the constancy of $d(Bi)/dt$ as HR increases to a decline in Pi. So the independence arises from interactions between parts of the cardiovascular system (the interactions that cause Pi to decline as HR increases), not from the ventricle itself, as is implied by the two models.

4. Future directions

The descriptions in the last section describe some aspects of AIS that can be improved upon. For example, the normal ventricle example shows that having methods that find tighter bounds on mathematical expressions would help AIS make more conclusions. However, if a model has ambiguities to begin with, better bounding of mathematical expressions will not help clear up those ambiguities.

One might think that a symbolic math system like MACSYMA [8] would help a lot with finding tighter bounds, but it does not: such a system helps simplify math expressions, but does not place numeric or symbolic bounds on them. In fact, an early version of the bounding system in AIS used MACSYMA as a subroutine to do the expression simplifying.

At present, AIS handles behaviors where the change in parameter values in a phase is invariant over time: each repetition of a phase changes the parameters by the same constant amounts. A way to describe these behaviors is constant 'velocity': the change in an accumulating parameter (like distance) is constant. A way to extend AIS to handle other types of repetitive behavior is to look at the properties that are invariant over time in those other types of behavior while keeping in mind the limitations imposed by the present abilities (or lack of) in automatically symbolically solving possibly nonlinear and simultaneous equations. Two examples of other repetitive behavior and their invariants have been briefly mentioned in Subsection 2.1. Handling certain situations may require finding how an iterating system initially responds to a perturbation. To find this response, one might combine AIS with a qualitative simulation [2] of the processes involved. AIS can help to disambiguate what happens next when the qualitative simulation is unsure, and the simulation can show the possible actions just after a perturbation (when AIS's assumptions are temporarily violated).

AIS has improved things by taking an input that is easier for a user to provide than either a single set of always applicable differential equations or AIS's output. However, a further improvement would be to automatically derive from a physical model the expressions used for AIS's input. Some work in qualitative reasoning [2] has started in this direction, it is not ready for some device as complicated as a ventricle. If it were, qualitative simulation would be able to analyze devices like a ventricle, which as mentioned in the introduction, it cannot.

Often, modeling involves deciding whether or not to make certain simplifying assumptions that make it possible/easier to draw certain conclusions when the assumptions are true. To enable AIS to decide on which assumptions to make, one might add to AIS some of the work done in [1, 4].

A related matter is that one cannot input to AIS a situation where the phase equations are conditional on certain parameter values. Enabling AIS to handle such situation would mean that AIS will need to detect and deal with situations where different phase equations are active on different iterations of an action sequence.

5. Summary

The automatic general analysis of large dynamic systems is beyond the reach of current systems. Fortunately, one can often get by just analyzing a system at steady-state. Some of the modeling and analysis for such steady-state analysis can be done by using electrical circuit analogies. AIS is a program that handles modeling and analysis for systems and parts of systems that cannot be handled in this fashion, but that do have the invariant property of steadily repeating a fixed sequence of parameter value changes. AIS takes advantage of this invariant to easily make the necessary computations to convert a description of a steadily iterating device's detailed behavior over the time scale of a single iteration into a description of that device's 'average' or net behavior over several iterations. The former type of description is easier to give, while the latter type is more useful.

Despite the limitation in the type of analysis and system that it can handle, AIS can still analyze some non-trivial problems. Besides the normal ventricle example given in this paper, AIS has also handled examples of a ventricle with mitral stenosis and a steam engine. The presented example both shows some of AIS's abilities, as well as some of the present limits of those abilities.

This limitation makes AIS less general than some other work on automatically dynamic systems. In exchange for this lack of generality, AIS does not get lost trying to find the iterated sequence, nor is AIS limited to descriptions in the form of a single set of differential equations. The work on AIS furthers the ability to automatically analyze dynamic systems, a goal of much work in artificial intelligence.

References

[1] S. Addanki, R. Cremonini and J. Penberthy, Reasoning about assumptions in graphs of models, in: *Proc. 11th Internat. Joint Conf. on Artificial Intelligence* (1989) 1432–1438.

[2] D.G. Bobrow, ed., *Qualitative Reasoning about Physical Systems* (MIT Press, Cambridge, MA, 1985). Reprinted from *Artificial Intelligence* 24 (1984).

[3] E. Braunwald, ed., *Heart Disease*, 3rd edition (W.B. Saunders, Philadelphia, PA, 1988).

[4] B. Falkenhainer and K. Forbus, Setting up large-scale qualitative models, in: *Proc. Seventh Nat. Conf. on Artificial Intelligence* (Aug. 1988) 301–306.

[5] C.V. Greenway, Mechanisms and quantitative assessment of drug effects on cardiac output with a new model of the circulation, *Pharmacol. Rev.* 33 (4) (1982).

[6] V.-K. Lau and K. Sagawa, Model analysis of the contribution of atrial contraction to ventricular filling, *Ann. Biomed. Engrg.* 7 (1979) 167–201.

[7] W.J. Long, S. Naimi, M.G. Criscitiello and R. Jayes, Using a physiological model for prediction of therapy effects in heart disease, in: *Proc. Computers in Cardiology Conf.* IEEE (Oct. 1986).

[8] Mathlab Group, *Macsyma Reference Manual.* MIT Laboratory for Computer Science, Cambridge, MA 02139, USA, version 10 edition, January 1983.

[9] C.S. Peskin and C. Tu, Hemodynamics in congenital heart disease, *Comput. Biol. Med.* 16 (5) (1986) 331–359.

[10] J. Ross, Cardiovascular systems (sec. 2), in: *Best and Taylor's Physiological Basis of Medical Practice*, 11th ed. (Williams and Wilkins, Baltimore, MD, 1985).

[11] E.P. Sacks, Hierarchical reasoning about inequalities, in: *Proc. Nat. Conf. on Artificial Intelligence*, American Association for Artificial Intelligence (1987) 649–654.

[12] E.P. Sacks, Automatic qualitative analysis of ordinary differential equations using piecewise linear approximations, *Artificial Intelligence* 41 (3) (1990) 313–364.

[13] K. Sagawa, Analysis of the ventricular pumping capacity as a function of input and output pressure loads, in: E. Reeve and A. Guyton, eds., *Physical Bases of Circulatory Transport: Regulation and Exchange*, Chapter 9 (W.B. Saunders Co., Philadelphia, PA, 1967).

[14] T. Sato, A. Takeuchi, J. Yamagami, H. Yamamoto, S. Akiyama, K. Endou, M. Shirataka, N. Ikeda and H. Tsuruta, Computer assisted instruction for therapy of heart failure based on simulation of cardiovascular system, in: *MEDINFO 86: Proc. Fifth Conf. on Medical Informatics*, Washington (North-Holland, Amsterdam, 1986) 761–765.

[15] G. Thomas, Jr., *Calculus and Analytic Geometry*, 4th ed. (Addison-Wesley, Reading, MA, 1968).

[16] D.S. Weld, The use of aggregation in causal simulation, *Artificial Intelligence* 30 (1) (1986) 1–34.

[17] D.S. Weld, Comparative analysis, in: *Proc. Tenth Internat. Joint Conf. on Artificial Intelligence* (1987) 959–965.

[18] D.S. Weld, Exaggeration, in: *Proc. Nat. Conf. on Artificial Intelligence*, American Association for Artificial Intelligence (1988).

[19] A. Yeh, Automatic analysis of systems at steady-state: Handling iterative dynamic systems and parameter uncertainty, Ph.D. Thesis, Mass. Institute of Technology, Cambridge, MA 02139, USA, September 1990.