Finding the average rates of change in repetitive behavior

Alexander Yeh* MIT Laboratory for Computer Science
545 Technology Sq., NE43-413
Cambridge, MA 02139 USA
ay@zermatt.lcs.mit.edu

ABSTRACT

The repetitive behavior of a device or system can be described in two ways: a detailed description of one iteration of the behavior, or a summary description of the behavior over many repetitions. This paper describes an implemented program called AIS that transforms the first type of description into the second type. AIS deals only with behavior where each repetition changes parameters by the same amounts. At present, the summary consists of the symbolic average rates of change in parameter values and information on how those rates would be different if various constants and functions had been different. Unlike some other approaches, AIS does not require that a repeating behavior be described in terms of a set of differential equations. Two examples of running AIS are given: one concerns the human heart, the other a steam engine.

INTRODUCTION: I have implemented a program called AIS (short for Analyzer of Iterated Sequences) that when given a continuous state-description of a system and a sequence of actions or transformations on that state, symbolically finds some of the time-averaged effects of continually iterating that sequence. The specific effects found at present include 1) the symbolic average rate of change in parameters that have a net increase or decrease in value with each iteration, and 2) how those rates of change would be different with different values of various constants and functions (sensitivity analysis). The sequences handled by AIS are ones which have the following "constancy": the sequence always has the same actions in the same order and each occurrence of a particular action changes the parameters by the same amounts. An example of such an iterated action sequence is the one taken by a heart in going through a beat cycle at steady-state. Effects to be found include the average rate at which blood enters the heart and how increasing the pressure of that entering blood affects that rate.

A motivation for finding such effects is that while modeling some system, there may be some sub-system β which iterates a sequence of actions at such a fast rate that the rest of the system only responds to β's behavior averaged over many iterations. Then a steady-state model for the entire system would only need a description of β's averaged behavior; β can be modeled as constantly iterating the same sequence of parameter value changes. Examples of such sub-system and system combinations include 1) the heart and the human circulatory system, and 2) an engine and a car.

Some other approaches of finding the behaviors of a continually iterating sequence have combined qualitative simulation with cycle detection [1]. For complicated systems (such as the heart), these simulations predict many possible sequences of actions besides the actual sequence. If the actual sequence can be isolated, one can use *aggregation* [10] to find which parameters change as the sequence repeats and use *comparative analysis* [11] to find the effects of perturbing model constants.

Another approach [6] uses piecewise-linear approximations of differential equations. This approach requires that one describe a system with a single set of always applicable differential equations. Creating such a description may often be hard, such as when describing a human heart or a steam engine. 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.

The next section describes the form of input for AIS. Following this are sections on how AIS processes that input and on what AIS can output. Afterwards are sections that give examples of AIS running on a description of a heart and a steam engine, respectively. The paper ends with a summary.

*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.

AIS INPUT: An input description consists of three parts: the parameters which describe the system state, static conditions on those parameters, and the sequence of actions (transformations) that gets iterated. The description only has to try to describe what happens in a sequence of actions, not necessarily how or why that sequence 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 action sequence 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 new action sequence iteration.

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

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

The second part of the input are 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.[1] The inequalities can be either definitions that are always true or conditions that are required for the given action sequence to iterate. An example of a definition is to say that some volume is ≥ 0. An example of a necessary condition is to say that for a normal sequence of actions in the heart, (input pressure) $<$ (output pressure).[2]

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 possibly 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's value is either monotonically non-decreasing or non-increasing.

Beyond these two requirements, a model-builder is free to divide a sequence into as few or many phases as desired. Also, a model-builder might violate the above requirements if the violation's consequences are judged to be negligible.

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 π, the corresponding expression gives π's value at the end of the phase.[3] For an accumulating parameter α, the expression gives the change in α's value each time that phase occurs. An expression may have algebraic and the more common 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[4] 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 and has as what stays constant be the *change* in the amount changed (or an even higher order of change). The second is having the alterations form a converging series [9, 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.

[1] Such a condition only makes a statement of the derivative with respect to the symbol mentioned. For example, mentioning that $0 < d^2f(x)/dx^2$ says nothing about $d^2f(y)/dy^2$. 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".

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

[3] 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.

[4] 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.

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 ≥ 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 a's change in value.

AIS PRELIMINARY PROCESSING: Before producing output, AIS needs to solve the equations given in the phase description and to check for obvious inconsistencies between the equations and given conditions.

To solve the equations, AIS computes for each phase: the change in value for each accumulating parameter, and the beginning and end values for each periodic parameter. These values and changes are expressed in terms of constant parameters. The beginning value of each periodic parameter is taken from that parameter's value at the end of the previous phase.[5] The solver currently handles only simple substitutions of the solved for the unsolved. Complicated equations like quadratics are left unsolved.

As an example of equation solving, suppose the equations

$$V_E = Y, \quad A_C = (V_E - V_B), \quad W_C = (P \cdot A_C)$$

are given, where Y is a constant and P is a periodic parameter that does not change during the phase. Let AIS find $V_B = Z$ and $P = Pi$ by looking at the values of V_E and P_E in the previous phase (Z and Pi are constants). Then AIS derives $V_E = Y$, $P = Pi$, $A_C = Y - Z$, $W_C = Pi \cdot (Y - Z)$.

To check for obvious inconsistencies, AIS enters the solved equations, the assumption that the rate of sequence repetition is positive, and the conditions given in the input (with periodic and accumulation parameter values substituted by the appropriate expression of constants) into the Bounder system [6]. This system checks for consistency by deriving an upper and lower numeric bound for every constant parameter. An inconsistency is declared if some parameter's lower bound is greater than its upper bound. Bounder derives the bounds with the *bounds propagation* and *substitution* methods. The former method reasons over numeric bounds. The latter method will also perform substitutions of symbolic expressions for symbols. For example, if $c > d + 5$, then the latter can find a lower bound on $(c - d)$ of $c - (c - 5) = 5$. In addition to these methods, the Bounder system uses an algebraic simplifier. These methods are also used to perform the bounding and inequality testing needed in the steps described below to produce the output.

AIS OUTPUT: After performing the above equation solving and inconsistency checking, AIS can infer the following about continually repeating the input sequence: 1) the average rate of change in an accumulating parameter including numeric bounds on that rate and the relative contribution of each phase to that rate, and 2) how that rate would differ if a constant symbol or function had a different value (sensitivity analysis).

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. Next AIS finds numeric bounds on this rate. Then AIS tries to determine which phases helped to increase or decrease this rate by observing which phases have a_C values that are bounded above and/or below by zero. 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

[5] The reason is given in a previous footnote. Also mentioned there is the consideration of the last phase in the sequence as "preceding" the first.

other symbols. AIS performs the phase determination task by looking at the derivatives (with respect to the symbol) of each phase's contribution to the rate (the phase's a_C value multiplied by the sequence repetition rate) and observing which are bounded above and/or below by zero. Those phases with a derivative of a_C that is > 0 made a positive contribution to the derivative, etc.

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 rate versus constant function 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 1st derivative is < 0 and the 2nd is $= 0$ (such as when the rate is $-3x$ and the symbol is x), then the curve shape is . However, if the 2nd is instead > 0 (such as when the rate is $\exp[-x]$) then the shape is . If the 1st derivative has no bounds, but the 2nd is < 0, then the possible shapes are , or .

In the future, the QS system [6] will probably 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 QS can be used, it needs to be extended 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 > 0, then if f were larger, the rate would be also, and so on.

HEART EXAMPLE: This section describes the current version of AIS running on a model of the beating of the part the human heart called the left ventricle.[6]

[6]The description is based on various texts and articles [5, 8] and makes many assumptions. One assumption is that blood is an incompressible fluid without inertia.

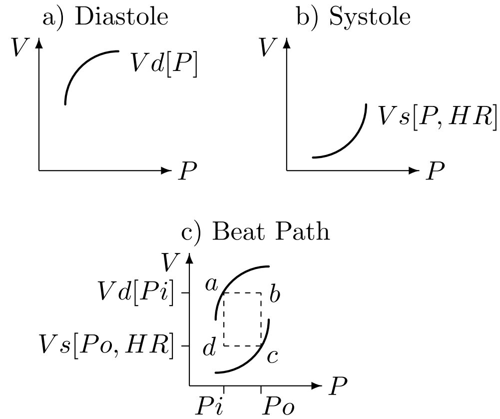

Figure 1: Curves for a Left Ventricle

The ventricle 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 volume (V) versus pressure (P) curve ($Vd[P]$) is roughly as shown in Figure 1a. When contracted (*systole*), the V versus P curve ($Vs[P, HR]$) is roughly as shown in Figure 1b. 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). Figure 1c 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.

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 entered the ventricle (Bi) and left the ventricle (Bo). The static conditions on the constants are:

[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.

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

Most of the conditions help 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:

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 phases' equations, AIS discovers the following average rates of change for the accumulating parameters and bounds on those rates:

$$\begin{aligned} d(W)/dt &= ((Pi \cdot (Vd[Pi] - Vs[Po, HR])) \\ &\quad + (-Po \cdot (Vd[Pi] - Vs[Po, HR]))) \cdot HR \\ d(Bi)/dt &= HR \cdot (Vd[Pi] - Vs[Po, HR]) > 0 \qquad (1) \end{aligned}$$

Also, $d(Bo)/dt = d(Bi)/dt$. One can show that $dW/dt < 0$, but the bounding mechanism cannot pick this up. In looking at the contributions of the phases to these rates, AIS discovers that the *ejection* phase is the only phase to affect $d(Bo)/dt$, making it as positive as it is. Similarly, the *filling* phase is the only phase to affect $d(Bi)/dt$. AIS can deduce that the *ejection* and *filling* phases are the ones that affect $d(W)/dt$, but cannot deduce how they affect $d(W)/dt$.

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 1st derivative with respect to HR is > 0, but no bounds are found for the 2nd derivative. No curve shape is deduced. With respect to the constant Pi, the 1st derivative is > 0 but the 2nd 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 $d(Bo)/dt$. As a check on the ventricle model, these rate shape results are compared to experimental results. The results for Pi and Po agree [7]. For HR, the AIS and experimental results are incomparable because the latter came from intact systems where changing HR can change Pi and Po.

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 .

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 if Vs were larger, these rates would be smaller. These results agree with the description in [8].

When modeling a circulatory system that has been averaged over many heart beats and is in a steady-state, such as done in [3, 8], most of the system's mechanics can be modeled by using direct current electrical circuit analogies (such as [pressure drop] = [resistance]·[flow]). Too complicated to be modeled this way is the part of the mechanics that relates the Pi, Po, HR, Vs, and Vd for each ventricle to the rate at which blood flows through that ventricle ($d(Bi)/dt = d(Bo)/dt$). Current modeling efforts either directly use empirically derived relationships (like [7]) or derive the needed equations by hand from an AIS-input-like description (done in [8]). 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 this relationship numerically, one must be more specific about the 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.

STEAM ENGINE EXAMPLE: This next example of running AIS concerns a simple steam engine (simplified version of the ones in [2]). This engine has one cylinder and a piston that slides back and forth along the inside of that cylinder. The piston also covers the main opening in the cylinder. The sequence of actions is that the piston slides further out in the cylinder and then back in. As piston slides out, the volume contained by the cylinder and piston combination (V) increases, moving from a low value of Vl to a high of Vh. Steam (at a pressure of Pi and a temperature of Ti) is let into the cylinder from $V = Vl$ to $V = Vex$. From $V = Vex$ to $V = Vh$, no steam is let in or out (steam in the cylinder expands adiabatically [4]). At $V = Vh$, a flywheel (connected to the piston via a crankshaft) pushes the piston back into the cylinder. As the piston slides back in, V decreases from a value of Vh back to Vl. From $V = Vh$ to $V = Vcp$, steam is let out of the cylinder via an exhaust port (at a pressure of Po). From $V = Vcp$ to $V = Vl$, no steam is let in or out (steam in the cylinder is compressed adiabatically). At $V = Vl$, the sequence repeats. The model makes many assumptions, including one that steam behaves almost like an ideal gas.

The parameters are: The symbol RPM (for revolutions per minute) gives the rate of sequence repetition. The constants are Pi, Ti, Po, Vl, Vex, Vcp, Vh, R and k. R is the constant in the ideal gas law $PV = nRT$, and kR is the molar specific heat of steam at constant volume [4]. The periodic parameters are V and the pressure inside the cylinder (P). The accumulating parameters are the amount of work done in driving the piston (W), the energy of all the steam entering the cylinder (Ei) and leaving the cylinder (Eo), and the amount of steam that has entered the cylinder (Ai).

Static conditions on the constants are:

$$0 < Po < Pi,\ 0 < Vl < Vex < Vh,\ Vl < Vcp < Vh,$$
$$0 < Ti,\ 0 < R,\ 3/2 \leq k$$

The sequence has six phases, each with a set of phase equations. After AIS solves these phase equations, it deduces the average rates of change for the accumulating parameters. For example,

$$\begin{aligned} d(Ei)/dt &= (k \cdot Vl \cdot (Pi - Po \cdot (Vcp/Vl)^{(1+1/k)}) \\ &\quad +Pi \cdot (Vex - Vl) \cdot (1 + k)) \cdot RPM \end{aligned}$$

AIS can determine which phases affect these rates, but cannot always determine how these phases affect the rates. A reason for this is that the bounding algorithms do not always find the tightest bounds on a given expression. This reason also causes AIS' inability to put numeric bounds on any of the rates.

In addition, this general shortcoming affects AIS' ability to bound the derivatives of the rates with respect to various constants. AIS cannot bound the 1st derivatives of any rate with respect to RPM, nor bound any of the 1st derivatives of the W rate. For the Ai, Ei and Eo rates, AIS can do better. For example, the 1st derivative of $d(Ei)/dt$ with respect to Pi is > 0 and the 2nd is $= 0$, so the $d(Ei)/dt$ versus Pi curve has a $\diagup$ shape.

SUMMARY: A program called AIS has been implemented and tested. It takes in a description of a sequence of actions and tries to find information associated with the symbolic average rate of change in various parameters.

Compared to some other work on automatically analyzing dynamic systems, AIS is limited in that it only analyzes systems which steadily repeat a fixed sequence of parameter value changes. In exchange for this limitation, 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. So work on AIS helps further the ability to automatically analyze dynamic systems, a goal of much work in artificial intelligence.

ACKNOWLEDGMENTS: Peter Szolovits and Jon Doyle helped in formally describing AIS' abilities. Elisha Sacks programmed and maintained the Bounder system. Members of the lab's CDMG group helped with modeling the heart and proof reading.

References

[1] D. G. Bobrow, ed.. *Qualitative Reasoning about Physical Systems.* MIT Press, 1985. Reprinted from *Artificial Intelligence*, vol. 24, 1984.

[2] Terrell Croft, editor. *Steam-Engine Principles and Practice.* McGraw-Hill Book Co., Inc., New York, 2nd edition, 1939. Revised by E. J. Tangerman.

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

[4] David Halliday and Robert Resnick. *Physics.* John Wiley and Sons, Inc., New York, 1960.

[5] J. Ross. Cardiovascular system. In *Best and Taylor's Physiological Basis of Medical Practice.* Williams & Wilkins, Baltimore, 11th ed., 1985.

[6] E. Sacks. Automatic qualitative analysis of ordinary differential equations using piecewise linear approximations. TR 416, MIT, Lab. for CS, 545 Tech. Sq., Cambridge, MA, 02139, 1988. Most of the material appears in *Artificial Intelligence*, 41(3), Jan. 1990.

[7] K. Sagawa. Analysis of the ventricular pumping capacity as a function of input and output pressure loads. In E. Reeve & A. Guyton, ed., *Physical Bases of Circulatory Transport: Regulation and Exchange.* W. B. Sanders Co., Philadephia, 1967.

[8] T. Sato, *et. al.* Computer assisted instruction for therapy of heart failure based on simulation of cardiovascular system. In *MEDINFO 86: Proceedings of the Fifth Conference on Medical Informatics*, p. 761–765, Washington, Oct. 1986. North-Holland.

[9] G. Thomas, Jr. *Calculus and Analytic Geometry, 4th edition.* Addison-Wesley Publishing Co., 1968.

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

[11] Daniel S. Weld. Theories of comparative analysis. AI-TR 1035, MIT, AI Lab., 545 Tech. Sq., Cambridge, MA, 02139, May 1988.