

COORDINATED SCIENCE LABORATORY

College of Engineering

**COMPUTER UNDERSTANDING
AND GENERALIZATION OF
SYMBOLIC MATHEMATICAL
CALCULATIONS**

**Jude W. Shavlik
Gerald F. DeJong**

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified		1b. RESTRICTIVE MARKINGS None			
2a. SECURITY CLASSIFICATION AUTHORITY N/A		3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited			
DECLASSIFICATION/DOWNGRADING SCHEDULE N/A					
4. PERFORMING ORGANIZATION REPORT NUMBER(S) UILU-ENG-86- 2219		5. MONITORING ORGANIZATION REPORT NUMBER(S) N/A			
6a. NAME OF PERFORMING ORGANIZATION Coordinated Science Lab University of Illinois		6b. OFFICE SYMBOL (If applicable) N/A	7a. NAME OF MONITORING ORGANIZATION National Science Foundation		
6c. ADDRESS (City, State and ZIP Code) 1101 W. Springfield Ave. Urbana, Illinois 61801		7b. ADDRESS (City, State and ZIP Code) 1800 G Street Washington, D.C. 20550			
8a. NAME OF FUNDING/SPONSORING ORGANIZATION National Science Foundation		8b. OFFICE SYMBOL (If applicable) N/A	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER NSF IST 85-11542		
8c. ADDRESS (City, State and ZIP Code) 1800 G Street Washington, D.C. 20550		10. SOURCE OF FUNDING NOS.			
		PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT NO.
11. TITLE (Include Security Classification) Computer Understanding Generalization of Symbolic Mathematical Calculations		N/A	N/A	N/A	N/A
12. PERSONAL AUTHOR(S) Rudolf W. Shavlik, Gerald F. DeJong					
13a. TYPE OF REPORT Technical		13b. TIME COVERED FROM _____ TO _____		14. DATE OF REPORT (Yr., Mo., Day) January 1986	
15. PAGE COUNT 18					
16. SUPPLEMENTARY NOTATION					
17. COSATI CODES		18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)			
FIELD	GROUP	SUB. GR.			
		machine learning, explanation-based learning, understanding mathematics, physics problem solving, computational systems			
19. ABSTRACT (Continue on reverse if necessary and identify by block number)					
<p>An artificial intelligence system that learns by observing its users perform symbolic mathematical problem solving is presented. This fully-implemented system is being evaluated as a problem solver in the domain of classical physics. Using its mathematical and physical knowledge, the system determines why a human-provided solution to a specific problem suffices to solve the problem, and then extends the solution technique to more general situations, thereby improving its own problem-solving performance. This research illustrates a need for symbolic mathematics systems to produce <i>explanations</i> of their problem-solving steps, as these explanations guide learning. Although physics problem solving is currently being investigated, the results obtained are relevant to other mathematically-based domains. This work also has implications for intelligent computer-aided instruction in domains of this type.</p>					
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT CLASSIFIED/UNLIMITED <input checked="" type="checkbox"/> SAME AS RPT. <input type="checkbox"/> DTIC USERS <input type="checkbox"/>			21. ABSTRACT SECURITY CLASSIFICATION Unclassified		
22a. NAME OF RESPONSIBLE INDIVIDUAL		22b. TELEPHONE NUMBER (Include Area Code)		22c. OFFICE SYMBOL None	

Technical Report
UILU-ENG-86-2219

COMPUTER UNDERSTANDING AND GENERALIZATION OF
SYMBOLIC MATHEMATICAL CALCULATIONS *
A Case Study in Physics Problem Solving

Jude W. Shavlik[†]
Gerald F. DeJong

Artificial Intelligence Research Group
Coordinated Science Laboratory
University of Illinois at Urbana-Champaign
Urbana, IL 61801

January 1986

ABSTRACT

This paper also appears on pages 148-153 of the *Proceedings of the 1986 Symposium on Symbolic and Algebraic Computation*, Waterloo, Ontario, CANADA, July 1986.

An artificial intelligence system that learns by observing its users perform symbolic mathematical problem solving is presented. This fully-implemented system is being evaluated as a problem solver in the domain of classical physics. Using its mathematical and physical knowledge, the system determines *why* a human-provided solution to a specific problem suffices to solve the problem, and then extends the solution technique to more general situations, thereby improving its own problem-solving performance. This research illustrates a need for symbolic mathematics systems to produce *explanations* of their problem-solving steps, as these explanations guide learning. Although physics problem solving is currently being investigated, the results obtained are relevant to other mathematically-based domains. This work also has implications for intelligent computer-aided instruction in domains of this type.

* This research was partially supported by the National Science Foundation under grant NSF IST 85 11542.

[†] University of Illinois Cognitive Science/Artificial Intelligence Fellow.

COMPUTER UNDERSTANDING AND GENERALIZATION OF SYMBOLIC MATHEMATICAL CALCULATIONS

1. INTRODUCTION

Symbolic mathematics systems, such as **MACSYMA** [Mathlab83], **MAPLE** [Geddes82], **REDUCE** [Hearn84], and **SMP** [Wolfram83], perform remarkable feats. Unfortunately these systems do not improve their performance with experience, automatically adapt to the idiosyncrasies of individual users, nor provide comprehensible explanations of their problem-solving steps. Largely this is because the bulk of their mathematical knowledge is implicitly encoded within their algorithms (see [Fateman85] for an example of this). We have designed and implemented an artificial intelligence system that learns by observing its users perform symbolic mathematical problem solving. We are evaluating our system as a problem-solver in the domain of classical physics. This is an elegant domain that stresses the use of complicated mathematics.

A central component of our system is a symbolic mathematics package. Its task is to provide the knowledge needed to make sense of the solutions provided to the system. Interestingly, no existing symbolic manipulation package is adequate. This is because the mathematics package must not only be capable of transforming one expression into another, but must leave a processing trace of the *effects* of its manipulations. A trace is needed so that our system can reason about the role of each of the solution steps. This aspect makes it unique among symbolic mathematics packages and places important constraints on how it can work.

Our system is capable of performing many of the mathematical manipulations expected of a college freshman who has encountered the calculus. By analyzing worked examples, it acquires concepts taught in a college-level introductory physics course: hence the name of the system, **Physics 101**. Newton's laws - which are provided to the system - suffice to solve all problems in

classical mechanics, but the general principles that are consequences of Newton's laws are interesting for their elegance as well as their ability to greatly simplify the solution process. The acquisition of one such concept, conservation of momentum, is used as an illustrative example throughout this paper.

Physics 101 is a schema-based problem solver. A schema (also called frame or script) [Chafe75, Minsky75, Schank77] is a data structure used to store the details of a problem-solving technique. When presented with a new problem, a schema-based problem solver like ours attempts to apply known techniques from its schema library. If no known schemata apply, the system is not able to solve the problem.

Explanation-based learning [DeJong81, DeJong86, Mitchell86], is a computer-based knowledge acquisition method that utilizes sophisticated domain representations. In this type of learning a computer generalizes a problem solution into a form that can be later used to solve conceptually similar problems. The generalization process is driven by the *explanation* of why the solution worked. The deep knowledge about the domain allows the explanation to be developed and then extended. We are applying this paradigm to the learning of classical physics.

In our system, the *explanation* is a sequence of applications of particular symbolic manipulation rules. Thus, if our system is to learn, it is not sufficient for the symbolic manipulation package to simply derive values for unknown quantities. Instead, it must generate and preserve the actual sequence of steps that result in the determination of unknowns. The explanation of each solution step must support reasoning about the step's validity and role in solving the over-all problem. **Physics 101** can then determine the weakest form of each rule application that contributes directly or indirectly to the solution, while continuing to preserve the over-all validity of the solution. The generalized rule sequence, all of its preconditions, and all of its effects are stored as a new schema. The new schema can then be used as a kind of macro-rule which may later be applied as a single problem-solving step. The resulting increase in efficiency brings previously insoluble classes of problems within the system's ability.

We envision incorporating an explanation-based learning system such as ours into systems that perform symbolic mathematical computations. In this vein, it can be viewed as a *learning apprentice* for domains based on mathematical calculation. Learning apprentices have been defined [Mitchell85] as

interactive knowledge-based consultants that directly assimilate new knowledge by observing and analyzing the problem-solving steps contributed by their users through their normal use of the system.

Since our system constructs detailed explanations, it can explain its answers to naive users, point out faulty human solution steps, and fill in the gaps in sketchy calculations. In addition, it improves its own problem-solving abilities with experience. For these reasons, this work also has implications for intelligent computer-aided instruction (ICAI) [Sleeman82]. Although we are currently working within the domain of physics, the results obtained are relevant to other mathematically-based domains.

2. SYSTEM OVERVIEW

Figure 1 presents an overview of the operation of the system. When the system cannot solve a problem, it requests a solution from its user. The solution provided must then be verified;

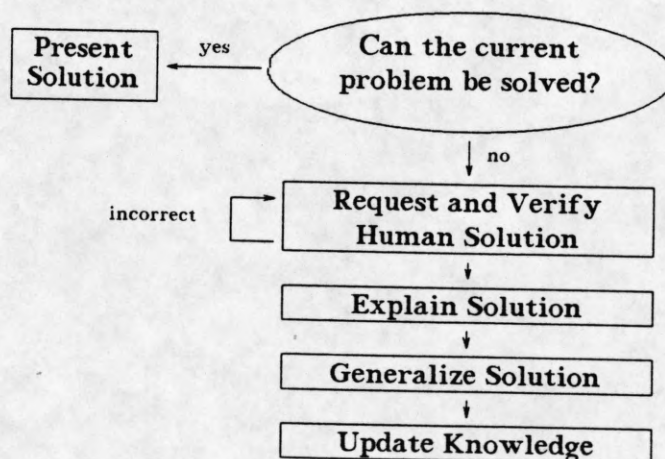


Figure 1. System Overview

additional details are requested when steps in the solution cannot be understood. We divide the process by which **Physics 101** understands an example into two phases. First, using its current knowledge about mathematics and physics, the system verifies that each solution step mathematically follows. It also infers missing steps. At the end of this phase the system knows that the user's solution solves the current problem, but it has no understanding of the global role of each step. During the second phase of understanding, the system determines these global roles. Understanding new formulae encountered in the solution is especially important. After this phase **Physics 101** has a firm understanding of how and why this solution solved the problem at hand. At this point it is able to profitably generalize any new principles that are used in the solution process, thereby increasing its knowledge of classical physics.

Physics 101 possesses a large number of mathematical problem-solving strategies. For example, it can symbolically integrate expressions, cancel variables, perform arithmetic, and replace terms by substituting known formulae. Figure 2 contains the initial physics formulae known to the system. These formulae are instantiated for each specific physical situation. Newton's second and third laws appear in figure 2. (Newton's first law is a special case of his second law.) The second law states that the net force on an object equals its mass times its acceleration. The net force is decomposed into two components: the external force and the internal force. External forces result from any external fields that act upon objects. Object I's internal force is the sum of the forces the other objects exert on object I. These *inter-object* forces are constrained by Newton's third law, which says that every action has an equal and opposite reaction.

An Illustrative Example

The current implementation of the model learns the physical concept of momentum conservation by analyzing, and then generalizing, a human's solution to a simple collision problem. The sample problem is shown in figure 3. In this one-dimensional problem there are two objects moving in free space, without the influence of any external forces. (Nothing is known about the

$$\forall_{obji} [\overrightarrow{\text{velocity}}_{obji}(t) = \frac{d}{dt} \overrightarrow{\text{position}}_{obji}(t)]$$

$$\forall_{obji} [\overrightarrow{\text{acceleration}}_{obji}(t) = \frac{d}{dt} \overrightarrow{\text{velocity}}_{obji}(t)]$$

$$\forall_{obji} [\overrightarrow{\text{force}}_{\text{net},obji}(t) = \text{mass}_{obji} * \overrightarrow{\text{acceleration}}_{obji}(t)]$$

$$\forall_{obji} [\overrightarrow{\text{force}}_{\text{net},obji}(t) = \overrightarrow{\text{force}}_{\text{external},obji}(t) + \overrightarrow{\text{force}}_{\text{internal},obji}(t)]$$

$$\forall_{obji} [\overrightarrow{\text{force}}_{\text{internal},obji}(t) = \sum_{\substack{objj = obj1 \\ objj \neq obji}}^{objN} \overrightarrow{\text{force}}_{objj,obji}(t)]$$

$$\forall_{obji} [\forall_{objj \neq obji} [\overrightarrow{\text{force}}_{objj,obji}(t) = - \overrightarrow{\text{force}}_{obji,objj}(t)]]$$

Figure 2. The Initial Formulae of the System

forces between the two objects. For example, besides their mutual gravitational attraction, there could be a long-range electrical interaction and a very complicated interaction during the collision.) In the initial state (**state A**) the first object is moving toward the second, which is stationary. Some time later (**state B**) the first object is recoiling from the resulting collision. The task is to determine the velocity of the second object after the collision.

First, the system unsuccessfully attempts to solve the problem using its initial knowledge. It cannot solve this problem, though, as the force exerted on object 2 by object 1 must be integrated and this force is not known. At this point the system requests a solution from its user. The solution provided can be seen in figure 4. Without explicitly stating it, the human problem solver takes advantage of the principle of conservation of momentum, as the momentum ($mass \times velocity$) of the world at two different times is equated. After that, various algebraic manipulations lead to the answer.

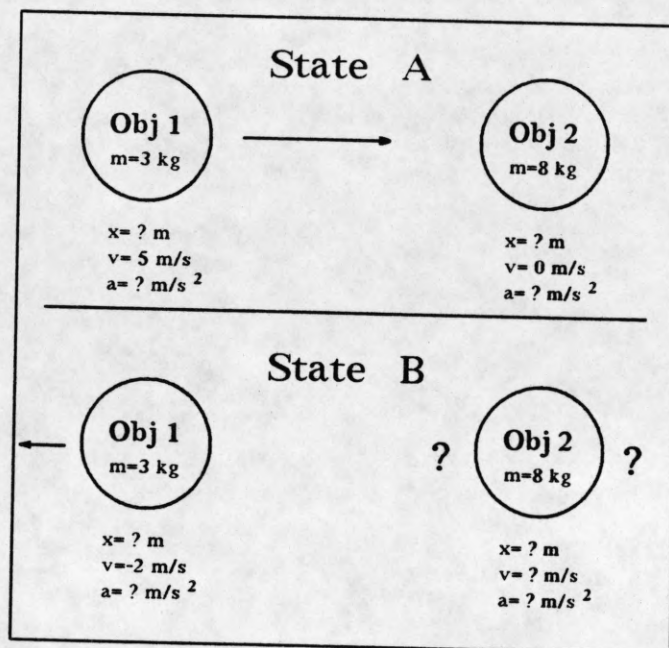


Figure 3. A Two-Body, One-Dimensional Collision Problem

$$\begin{aligned}
 & \text{mass}_{\text{obj1,stateA}} \text{velocity}_{\text{obj1,stateA,X}} + \text{mass}_{\text{obj2,stateA}} \text{velocity}_{\text{obj2,stateA,X}} \\
 & = \text{mass}_{\text{obj1,stateB}} \text{velocity}_{\text{obj1,stateB,X}} + \text{mass}_{\text{obj2,stateB}} \text{velocity}_{\text{obj2,stateB,X}} \\
 & 3\text{kg} * 5\text{m/s} = 3\text{kg} * -2\text{m/s} + 8\text{kg} \text{velocity}_{\text{obj2,stateB,X}} \\
 & 15\text{kgm/s} = -6\text{kgm/s} + 8\text{kg} \text{velocity}_{\text{obj2,stateB,X}} \\
 & \text{velocity}_{\text{obj2,stateB,X}} = 2.63\text{m/s}
 \end{aligned}$$

Figure 4. The Human's Solution

Physics 101 analyzes the solution in figure 4 and determines that summing two objects' momenta (in a world containing *only* two objects) eliminates the force each object exerts upon the other, regardless of the details of these forces. (This is a consequence of Newton's third law.)

Equation 1 presents the result **Physics 101** obtains by extending the human's solution

technique to a world with an arbitrary number of objects.¹ Since each object in a physical situation potentially exerts a force on every other object, in the general case cancelling the net inter-object force upon an object requires summing the momenta of *all* the objects.

$$\frac{d}{dt} \sum_{i=1}^N \text{mass}_i \text{velocity}_i = \sum_{i=1}^N \text{force}_{\text{external},i} \quad (1)$$

This formula says: *The rate of change of the total momentum of a collection of objects is determined by the sum of the external forces on those objects.* Other problems, which involve any number of bodies under the influence of external forces, can be solved by the system using this generalized result. The following presents the process by which **Physics 101** understands and then extends the solution in figure 4.

3. UNDERSTANDING SOLUTIONS

Understanding a solution involves two phases. First, the system attempts to verify that each solution step mathematically follows. If successful, in the second phase **Physics 101** builds an explanation of why the solution works.

Verifying Solutions

In order to accept a user's answer, **Physics 101** has to verify each of the steps in the human's solution. Besides being mathematically correct, the calculations must be physically consistent. To be valid, each of the solution steps must be assigned to one of the following four classifications.

- (1) Instantiation of a known formula; *force = mass × acceleration* is of this type.
- (2) Definition of a new variable to shorten later expressions; *resistance = voltage / current* would fall in this category.

¹ For clarity, a two-object collision problem is presented here. However, the current implementation requires an example involving at least three objects to properly motivate this final result. The reasons for this are described later.

- (3) Rearrangement of a previously-used formula. These equations are mathematical variants of previous steps. The replacement of variables by their values also falls into this category.
- (4) Statement of an unknown relationship among known variables. These steps require full justification, which the system performs symbolically by reasoning about algebra and calculus. Only the equations in this category are candidates for generalization.

The last three steps in figure 4 can easily be verified, as they are simple algebraic manipulations (classification 3 above). The first equation falls into classification 4, as these variables are known to the system, yet this equation is not a variant of any known formula (those of figure 2). A physically-consistent mathematical derivation is needed. Since the two sides of this initial equation only differ as to the state in which they are evaluated, an attempt is made to determine a time-dependent expression describing the general form of one side of the equation.

The actual calculations of the system appear in figure 5. (The top expression is called the *left-hand side* of the calculation, while the other expressions are termed *right-hand sides*.) The goal is to convert, via a series of equality-preserving transformations, the top expression in figure 5 into an equivalent expression whose time dependence is explicit. Once this is done, the system can determine if the first equation in the human's solution (figure 4) is valid.

The annotations in the left-hand column of figure 5 are produced by the system. These annotations indicate how **Physics 101** explains each calculation step. In the first step, the formulae substitutions are chosen as a last resort.² This means that they are not chosen in support of a variable cancellation. In the next step, the formulae substitutions are chosen because the mass terms can be cancelled. Before this cancellation can take place, however, the cancelling terms must be brought together. The calculation continues in a like manner until all the unknown variables are eliminated. Then the known values are substituted and the ensuing arithmetic and calculus is solved. The final result of figure 5 validates the first equation in the user's solution, as an

² Initially, the system chose to replace the velocities by the derivative of the positions. This led nowhere and the system backtracked. No other backtracking occurred during the calculation of figure 5. The system is guided by the goal of cancelling variables, which greatly reduces the amount of unnecessary substitutions during problem solving. **Physics 101's** problem solver is discussed further in [Shavlik86].

$$\begin{aligned}
& \text{mass}_{\text{obj1}} \text{velocity}_{\text{obj1},x}(t) + \text{mass}_{\text{obj2}} \text{velocity}_{\text{obj2},x}(t) \\
\text{SubstFormulae} &= \text{mass}_{\text{obj1}} \int \text{acceleration}_{\text{obj1},x}(t) dt + \text{mass}_{\text{obj2}} \int \text{acceleration}_{\text{obj2},x}(t) dt \\
\text{SubstAlgebra} &= \text{mass}_{\text{obj1}} \int (\text{force}_{\text{net,obj1},x}(t) / \text{mass}_{\text{obj1}}) dt + \text{mass}_{\text{obj2}} \int (\text{force}_{\text{net,obj2},x}(t) / \text{mass}_{\text{obj2}}) dt \\
\text{ConstantsOutOfCalculus} &= (\text{mass}_{\text{obj1}} / \text{mass}_{\text{obj1}}) \int \text{force}_{\text{net,obj1},x}(t) dt + (\text{mass}_{\text{obj2}} / \text{mass}_{\text{obj2}}) \int \text{force}_{\text{net,obj2},x}(t) dt \\
\text{SubstNumIdentities} &= 1 \int \text{force}_{\text{net,obj1},x}(t) dt + 1 \int \text{force}_{\text{net,obj2},x}(t) dt \\
\text{Removeldentities} &= \int \text{force}_{\text{net,obj1},x}(t) dt + \int \text{force}_{\text{net,obj2},x}(t) dt \\
\text{SubstFormulae} &= \int (\text{force}_{\text{external,obj1},x}(t) + \text{force}_{\text{internal,obj1},x}(t)) dt + \int (\text{force}_{\text{external,obj2},x}(t) + \text{force}_{\text{internal,obj2},x}(t)) dt \\
\text{SubstFormulae} &= \int (\text{force}_{\text{external,obj1},x}(t) + \text{force}_{\text{obj2,obj1},x}(t)) dt + \int (\text{force}_{\text{external,obj2},x}(t) + \text{force}_{\text{obj1,obj2},x}(t)) dt \\
\text{SubstAlgebra} &= \int (\text{force}_{\text{external,obj1},x}(t) - \text{force}_{\text{obj1,obj2},x}(t)) dt + \int (\text{force}_{\text{external,obj2},x}(t) + \text{force}_{\text{obj1,obj2},x}(t)) dt \\
\text{CombineCalculus} &= \int (\text{force}_{\text{external,obj1},x}(t) - \text{force}_{\text{obj1,obj2},x}(t) + \text{force}_{\text{external,obj2},x}(t) + \text{force}_{\text{obj1,obj2},x}(t)) dt \\
\text{SubstNumIdentities} &= \int (\text{force}_{\text{external,obj1},x}(t) + 0 \text{ kg m} / \text{s}^2 + \text{force}_{\text{external,obj2},x}(t)) dt \\
\text{Removeldentities} &= \int (\text{force}_{\text{external,obj1},x}(t) + \text{force}_{\text{external,obj2},x}(t)) dt \\
\text{SubstValues} &= \int (0 \text{ kg m} / \text{s}^2 + 0 \text{ kg m} / \text{s}^2) dt \\
\text{AddNumbers} &= \int 0 \text{ kg m} / \text{s}^2 dt \\
\text{SolveCalculus} &= \text{constant}_1
\end{aligned}$$

Figure 5. Verifying the First Step in the Human's Solution

expression that is constant can be equated for any two times.

Explaining Solutions

At this point the system has ascertained that the human's solution does indeed solve the current problem. In the next step, it analyzes its justifications of those equations falling into classification 4. The system must determine the need for including each variable in these equations. This will determine which variables are required in the general form of this equation.

First, the system determines the final status of each variable appearing in the calculation. If the value of a variable is used, the variable is assigned the status *value-used*. The status *fully-cancelled* is assigned to variables directly involved in a cancellation. If the variable is replaced in a formula substitution, its status is determined recursively. The final status of all of its descendant variables (those variables appearing in the substituted expression) are first determined. The variable receives a status of value-used if the values of all its descendant variables are used. Similarly, if all of these variables are fully cancelled, the parent is considered to have been fully cancelled. Otherwise the parent receives the status *partially-cancelled*. For example, in the calculation of figure 5, $velocity_{obj\ 2}$ receives a status of partially-cancelled, $force_{internal, obj\ 2}$ receives a status of fully-cancelled, and $force_{external, obj\ 2}$ gets a status of value-used.

In the second phase of the explanation process, **Physics 101** determines how the final status of the current problem's *unknown* is obtained. The problem's *unknown* is the variable whose value is being sought; in the sample problem, $velocity_{obj\ 2}$. During this process, the system determines the role of each variable in the left-hand side of the calculation.

During a calculation one of three things can happen to a variable: (1) its value can be substituted, (2) it can be symbolically replaced during a formulae substitution, or (3) it can be *cancelled*. Understanding and generalizing variable cancellation drives **Physics 101**.

Obstacles are variables appearing in a calculation but whose values are not known. *Primary obstacles* are obstacle variables descended from the *unknown*. In the momentum problem the only primary obstacle is $force_{internal, obj\ 2}$. If the value of each of the primary obstacles were known, the value of the unknown would be specified. The system ascertains how these obstacles are eliminated from the calculation. Cancelling obstacles is seen as the essence of the solution strategy, because when all the obstacles have been cancelled the value of the unknown can be easily calculated.

First, the system determines that $force_{internal, obj\ 2}$ is *additively cancelled*. Although cancelled additively, this variable originally appears in a multiplicative expression ($a = \frac{f}{m}$). Hence, the system must determine how it is *additively isolated*. **Physics 101** discovers that multiplication by

$mass_{obj 2}$ performed this task. So an explanation of the $mass_{obj 2}$ term in the left-hand side expression is obtained.

The next thing to do is to determine how the terms that additively cancel $force_{internal, obj 2}$ are introduced into the calculation. $Force_{internal, obj 2}$ is replaced by the equivalent $force_{obj 1, obj 2}$, which is cancelled by the equal-and-opposite $force_{obj 2, obj 1}$ descended from $velocity_{obj 1}$. The $force_{obj 2, obj 1}$, too, must first be additively isolated. **Physics 101** discovers that the left-hand side's $mass_{obj 1}$ performs this isolation. The system now has explanations for the $mass_{obj 1}$ and the $velocity_{obj 1}$ terms in the left-hand side.

Cancellation of the primary obstacles requires the presence of additional variables on the left-hand side of the equation. These extra terms may themselves contain obstacle variables. These are called *secondary obstacles*. **Physics 101** must also determine how these obstacles are eliminated from the calculation. The elimination of the secondary obstacles may in turn require the presence of additional variables in the left-hand side expression, which may introduce further obstacles. This recursion must terminate, however, as the calculation resulted in the elimination of all unknown terms.

Once the system determines how all of the obstacles in the calculation are cancelled, generalization can occur. At this time, **Physics 101** can also report any variables in the left-hand side of a calculation that are irrelevant to the determination of the value of the unknown.

4. GENERALIZING SOLUTIONS

Physics 101 performs generalization by using its explanation of the specific solution to guide the determination of the problem's unknown in the general case. This process is illustrated in the following figures.³ The system starts with the generalized unknown, $velocity_{obj 1}$. It then performs the general versions of the specific formulae substitutions that produced the first of the primary obstacles. This can be seen in figure 6.

³ During generalization, **Physics 101** produces a graphical description of its processing. The figures that follow (except figure 10) are actual outputs of the implemented system.

$$\begin{aligned}
& \text{velocity}_{\text{objl},x}(t) \\
&= \int \text{acceleration}_{\text{objl},x}(t) \, dt \\
&= \int (\text{force}_{\text{net,objl},x}(t) / \text{mass}_{\text{objl}}) \, dt \\
&= (1 / \text{mass}_{\text{objl}}) \int \text{force}_{\text{net,objl},x}(t) \, dt \\
&= (1 / \text{mass}_{\text{objl}}) \int (\text{force}_{\text{external,objl},x}(t) + \text{force}_{\text{internal,objl},x}(t)) \, dt
\end{aligned}$$

Figure 6. Introduction of the Primary Obstacle

Recall that the internal force is additively cancelled in the specific case. Hence, the next generalization step is to additively isolate $\text{force}_{\text{internal,objl}}$. The variable $\text{mass}_{\text{objl}}$ is introduced into the left-hand side of the general calculation in order to accomplish this isolation. Figure 7 presents this generalization step.

At this point the general version of the primary obstacle is isolated for an additive cancellation. To perform this cancellation, those terms that will cancel the internal force must be introduced into the general calculation. The system determines that in the specific solution the net

$$\begin{aligned}
& \text{mass}_{\text{objl}} \text{velocity}_{\text{objl},x}(t) \\
&= \text{mass}_{\text{objl}} \int \text{acceleration}_{\text{objl},x}(t) \, dt \\
&= \text{mass}_{\text{objl}} \int (\text{force}_{\text{net,objl},x}(t) / \text{mass}_{\text{objl}}) \, dt \\
&= (\text{mass}_{\text{objl}} / \text{mass}_{\text{objl}}) \int \text{force}_{\text{net,objl},x}(t) \, dt \\
&= 1 \int \text{force}_{\text{net,objl},x}(t) \, dt \\
&= \int \text{force}_{\text{net,objl},x}(t) \, dt \\
&= \int (\text{force}_{\text{external,objl},x}(t) + \text{force}_{\text{internal,objl},x}(t)) \, dt
\end{aligned}$$

Figure 7. Introduction of $\text{Mass}_{\text{objl}}$ to Isolate the Primary Obstacle

internal force acting on object 2 is indirectly cancelled because each of the inter-object forces acting upon object 2 is individually directly cancelled. Recall that in figure 5, the formula $force_{internal, obj 2} = force_{obj 1, obj 2}$ is used. The second from last formula in figure 2 is the general version of this specific formula.

In the general case, *all* of the other objects in a situation exert an inter-object force on object I. *All* of these inter-object forces need to be cancelled. In the specific case, $velocity_{obj 1}$ produced the canceller of object 2's internal force. The $mass_{obj 1}$ term is needed to isolate the canceller for the additive cancellation. So to cancel $force_{internal, obj 1}$ in the general case, a $mass \times velocity$ term must come from every other object in the situation. Figure 8 presents the introduction of the summation that produces the variables that cancel $force_{internal, obj 1}$. Notice how the goal of cancellation

$$\begin{aligned}
 & mass_{obj 1} velocity_{obj 1, x}(t) + \sum_{\substack{obj J = obj 1 \\ obj J \neq obj 1}}^{obj N} mass_{obj J} velocity_{obj J, x}(t) \\
 &= mass_{obj 1} \int acceleration_{obj 1, x}(t) dt + \sum_{\substack{obj J = obj 1 \\ obj J \neq obj 1}}^{obj N} mass_{obj J} \int acceleration_{obj J, x}(t) dt \\
 &= mass_{obj 1} \int (force_{net, obj 1, x}(t) / mass_{obj 1}) dt + \sum_{\substack{obj J = obj 1 \\ obj J \neq obj 1}}^{obj N} mass_{obj J} \int (force_{net, obj J, x}(t) / mass_{obj J}) dt \\
 &= (mass_{obj 1} / mass_{obj 1}) \int force_{net, obj 1, x}(t) dt + \sum_{\substack{obj J = obj 1 \\ obj J \neq obj 1}}^{obj N} (mass_{obj J} / mass_{obj J}) \int force_{net, obj J, x}(t) dt \\
 &= 1 \int force_{net, obj 1, x}(t) dt + \sum_{\substack{obj J = obj 1 \\ obj J \neq obj 1}}^{obj N} 1 \int force_{net, obj J, x}(t) dt \\
 &= \int force_{net, obj 1, x}(t) dt + \sum_{\substack{obj J = obj 1 \\ obj J \neq obj 1}}^{obj N} \int force_{net, obj J, x}(t) dt \\
 &= \int (force_{external, obj 1, x}(t) + force_{internal, obj 1, x}(t)) dt + \sum_{\substack{obj J = obj 1 \\ obj J \neq obj 1}}^{obj N} \int (force_{external, obj J, x}(t) + force_{internal, obj J, x}(t)) dt
 \end{aligned}$$

Figure 8. Introduction of the Cancellers of the Primary Obstacle

motivates generalizing the number of objects involved in this expression. (Some minor steps have been left out of figures 8 and 9, for the sake of brevity.)

Once all the cancellers of the generalized primary obstacle are present, the primary obstacle itself can be cancelled. This is shown in figure 9.

Now that the primary obstacle is cancelled, the system checks to see if any secondary obstacles have been introduced. As can be seen in figure 9, the inter-object forces *not* involving object I still remain. Figure 10 graphically illustrates these remaining forces. All of the forces acting on object I have been cancelled, while a force between objects J and K still appears whenever neither J nor K equal I. This highlights an important aspect of generalizing number. Introducing more entities may create interactions that do not appear in the specific example.

$$\begin{aligned}
 & \vdots \\
 & = \int (\text{force}_{\text{external,objI},x}(t) + \text{force}_{\text{internal,objI},x}(t)) dt + \sum_{\substack{\text{objJ}=\text{obj1} \\ \text{objJ} \neq \text{objI}}}^{\text{objN}} \int (\text{force}_{\text{external,objJ},x}(t) + \text{force}_{\text{internal,objJ},x}(t)) dt \\
 & = \int (\text{force}_{\text{external,objI},x}(t) + \sum_{\substack{\text{objJ}=\text{obj1} \\ \text{objJ} \neq \text{objI}}}^{\text{objN}} \text{force}_{\text{objJ,objI},x}(t)) dt + \sum_{\substack{\text{objJ}=\text{obj1} \\ \text{objJ} \neq \text{objI}}}^{\text{objN}} \int (\text{force}_{\text{external,objJ},x}(t) + \sum_{\substack{\text{objK}=\text{obj1} \\ \text{objK} \neq \text{objJ}}}^{\text{objN}} \text{force}_{\text{objK,objJ},x}(t)) dt \\
 & = \int (\text{force}_{\text{external,objI},x}(t) + \sum_{\substack{\text{objJ}=\text{obj1} \\ \text{objJ} \neq \text{objI}}}^{\text{objN}} [-\text{force}_{\text{objI,objJ},x}(t)]) dt + \sum_{\substack{\text{objJ}=\text{obj1} \\ \text{objJ} \neq \text{objI}}}^{\text{objN}} \int (\text{force}_{\text{external,objJ},x}(t) + \sum_{\substack{\text{objK}=\text{obj1} \\ \text{objK} \neq \text{objJ}}}^{\text{objN}} \text{force}_{\text{objK,objJ},x}(t)) dt \\
 & = \int (\text{force}_{\text{external,objI},x}(t) + \sum_{\substack{\text{objJ}=\text{obj1} \\ \text{objJ} \neq \text{objI}}}^{\text{objN}} [-\text{force}_{\text{objI,objJ},x}(t)]) + \sum_{\substack{\text{objJ}=\text{obj1} \\ \text{objJ} \neq \text{objI}}}^{\text{objN}} \text{force}_{\text{external,objJ},x}(t) + \sum_{\substack{\text{objJ}=\text{obj1} \\ \text{objJ} \neq \text{objI}}}^{\text{objN}} \sum_{\substack{\text{objK}=\text{obj1} \\ \text{objK} \neq \text{objJ}}}^{\text{objN}} \text{force}_{\text{objK,objJ},x}(t)) dt \\
 & = \int (\text{force}_{\text{external,objI},x}(t) + \sum_{\substack{\text{objJ}=\text{obj1} \\ \text{objJ} \neq \text{objI}}}^{\text{objN}} 0 \text{ kg m / s}^2 + \sum_{\substack{\text{objJ}=\text{obj1} \\ \text{objJ} \neq \text{objI}}}^{\text{objN}} \text{force}_{\text{external,objJ},x}(t) + \sum_{\substack{\text{objJ}=\text{obj1} \\ \text{objJ} \neq \text{objI}}}^{\text{objN}} \sum_{\substack{\text{objK}=\text{obj1} \\ \text{objK} \neq \text{objJ}}}^{\text{objN}} \text{force}_{\text{objK,objJ},x}(t)) dt
 \end{aligned}$$

Figure 9. Cancellation of the Primary Obstacle

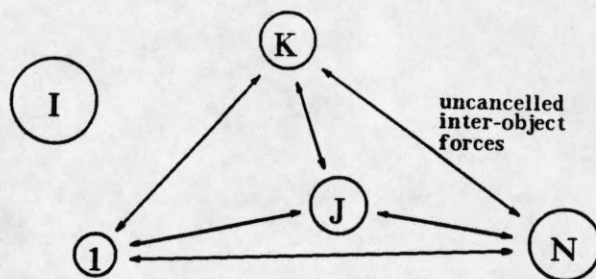


Figure 10. The Uncancelled Inter-Object Forces

Physics 101 cannot eliminate the remaining inter-object forces if the specific example only involves a two-object collision. It does not detect that the remaining forces all cancel one another, since in the two-object example there is no hint of how to deal with these secondary obstacles. A three-body collision must be analyzed by the system to properly motivate this cancellation. (In a three-body collision, $force_{obj\ 3,obj\ 1}$ cancels $force_{obj\ 1,obj\ 3}$; neither of these variables are descendants of $velocity_{obj\ 2}$.) When the specific example involves three objects, the system ascertains that the remaining inter-object forces cancel. In this case, the result previously presented in equation 1 is produced and added to the system's database.

The Cancellation Graph

Figure 11 contains the *cancellation graph* for a three-body collision problem. This data structure is built by the system during the understanding of the specific solution. It holds the information that explains how the specific example's obstacles are eliminated from the calculation. This information is used to guide the generalization process illustrated above. This graph and its relation to the preceding figures is summarized below.

The graph in figure 11 records that the only obstacle of the unknown ($velocity_{obj\ 2}$) is object 2's internal force. This primary obstacle is blocked from an additive cancellation because it is divided by mass (figure 6). Another mass term cancels the additive blocker (figure 7). Once object 2's internal force is isolated, it is additively cancelled by $force_{obj\ 2,obj\ 1}$ and $force_{obj\ 2,obj\ 3}$

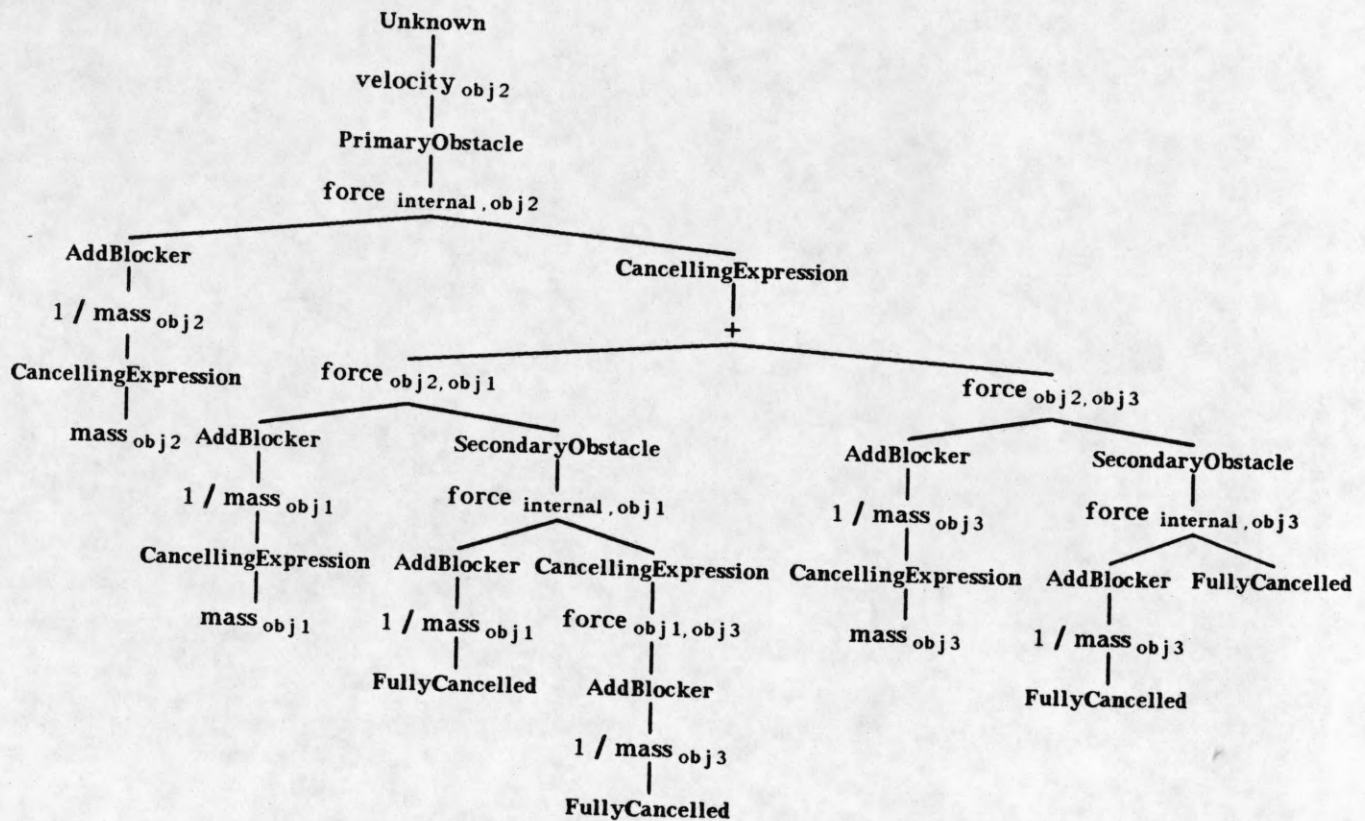


Figure 11. The Cancellation Graph

(figures 8 and 9). However, before cancellation can occur the additive blockers of both of these terms must be cancelled. Introducing these two inter-object forces results in the introduction of two secondary obstacle; the internal forces of objects 1 and 3. Both of these can be additively cancelled, since their additive blockers are already cancelled. The remainder of object 1's internal force is cancelled by the inter-object force between object's 1 and 3 (recall that a portion of this internal force cancelled part of object 1's internal force). Cancelling the internal force of object 1 also fully cancels the other secondary obstacle; the internal force of object 3. In a two-body problem the only secondary obstacle ($force_{internal, obj 1}$) is fully cancelled when the primary obstacle ($force_{internal, obj 2}$) is cancelled. In that case, there is no information to motivate the cancellation of the portions of the other internal forces that remain once the unknown's internal force is cancelled (figure 10).

5. CONCLUSION

We have developed a system that learns in a complex domain requiring both symbolic and numeric reasoning. Our approach is knowledge-based: the system requires and applies detailed knowledge about algebra, the calculus, and Newton's laws. Once a new concept is learned, it is added to the system's collection of knowledge. It is thereby available to help solve future problems and as a stepping stone toward acquiring more difficult concepts. Systems like **Physics 101** can form the basis for intelligent problem solvers in various mathematically-based domains.

By analyzing a worked example involving a fixed number of physical objects, the current implementation of **Physics 101** is able to derive a formula describing the temporal evolution of the momentum of a situation involving *any* number of objects. Generalization of the number of objects involved in the formula is motivated by the system's explanation of how the specific solution worked. The formula acquired can be used to solve a collection of complicated collision problems.

In **Physics 101**, learning is based on the detailed analysis of mathematical solutions to specific problems. The explanations of the solution steps are used to guide the generalization of the solution technique. This research illustrates the need to receive from systems that perform symbolic mathematical computations, not only a final answer, but also the step-by-step details of their computations.

REFERENCES

- [Chafe75] W. Chafe, "Some Thoughts on Schemata," *Theoretical Issues in Natural Language Processing 1*, Cambridge, MA, 1975, pp. 89-91.
- [DeJong81] G. F. DeJong, "Generalizations Based on Explanations," *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, Vancouver, B.C., Canada, August 1981, pp. 67-70.
- [DeJong86] G. F. DeJong and R. J. Mooney, "Explanation-Based Learning: An Alternative View," *Machine Learning 1*, 2 (April 1986), .
- [Fateman85] R. J. Fateman, "Eleven Proofs of $\sin^2x + \cos^2x = 1$," *SIGSAM Bulletin 19*, 2 (May 1985), pp. 25-28.
- [Geddes82] K. Geddes, G. Gonnet and B. Char, "MAPLE User's Manual, Second Edition," Technical Report CS-82-40, University of Waterloo, Ontario, CANADA, December 1982.
- [Hearn84] A. Hearn, "REDUCE User's Manual, Version 3.1," Technical Report CP-78, The RAND Corporation, Santa Monica, CA, April 1984.
- [Mathlab83] Mathlab, "MACSYMA Reference Manual, Version Ten," Laboratory for Computer Science, MIT, Cambridge, MA, December 1983.
- [Minsky75] M. L. Minsky, "A Framework for Representing Knowledge," in *The Psychology of Computer Vision*, P. H. Winston (ed.), McGraw-Hill, New York, NY, 1975, pp. 211-277.
- [Mitchell85] T. M. Mitchell, S. Mahadevan and L. I. Steinberg, "LEAP: A Learning Apprentice for VLSI Design," *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, Los Angeles, CA, August 1985, pp. 573-580.
- [Mitchell86] T. M. Mitchell, R. Keller and S. Kedar-Cabelli, "Explanation-Based Generalization: A Unifying View," *Machine Learning 1*, 1 (January 1986), pp. 47-80.
- [Schank77] R. C. Schank and R. P. Abelson, *Scripts, Plans, Goals and Understanding: An Inquiry into Human Knowledge Structures*, Lawrence Erlbaum and Associates, Hillsdale, NJ, 1977.
- [Shavlik86] J. W. Shavlik and G. F. DeJong, "A Model of Attention Focussing During Problem Solving," *Proceedings of the Eighth Annual Conference of the Cognitive Science Society*, Amherst, MA, August 1986.
- [Sleeman82] D. H. Sleeman and J. S. Brown (ed.), *Intelligent Tutoring Systems*, Academic Press, New York, NY, 1982.
- [Wolfram83] S. Wolfram, "SMP Reference Manual," Computer Mathematics Group, Inference Corporation, Los Angeles, CA, 1983.