MEANS-ENDS ANALYSIS AND THE SOLUTION OF MECHANICS PROBLEMS

by
George F. Luger
Associate Professor, Computer Science
University of New Mexico
Albuquerque, New Mexico 87313, USA

-ABSTRACT-

Several years ago David Marples (1976) proposed an algorithm for derivation of simultaneous equations in the solution of Mechanics problems. Although the original intent of this algorithm was to assist his undergraduate students at Cambridge in solving applied mathematics problems, it has also proven itself a powerful tool in the MECHO automatic problem solving system (Bundy et al, 1978, 1979). This paper will briefly discuss the Marples' algorithm and demonstrate its use with two mechanics problems. Parts of traces of four humans solving the same problems will be given. Adjustments in the MECHO program are made to show how close the Marples' algorithm can fit the data of the human subjects. Brief concluding comments are made on modelling human behavior with a rule-based language.

I. INTRODUCTION

David Marples (1976) proposed an algorithm for production of simultaneous equations in the solution of mechanics problems. This algorithm was originally introduced in his tutorial sessions at Cambridge and was intended to help the students produce a sufficient number of independent simultaneous equations to solve mechanics problems. The technique is general and goal driven. It has been adopted as part of the MECHO automatic problem solver.

In Section II, the algorithm will be explained and compared with The General Problem Solver (Newell & Simon, 1963, 1972). Two problems, a pulley problem and a distance/rate/time problem will be introduced and the MECHO solution of each of these problems presented.

In Section III, parts of protocols of four subjects will be presented, and in Section IV, logically justified adjustments will be made to the Marples' algorithm in an attempt to produce a trace similar to the human protocol. Section V will present some concluding comments.

II. THE MARPLES' ALGORITHM

The Marples' algorithm is goal driven. It takes the goal or unknown to be solved for in a problem and searches back through the givens of the problem in an attempt to find an equation solving for the unknown in terms of the givens. When this is impossible, it creates intermediate unknowns, or subgoals, that will solve for the unknown and then attempts to solve for the intermediate unknowns in terms of the givens of the problem.

To see how this might be done in the solving of applied mathematics problems, consider the usual role of equations in solving a problem:

Equation V=U+A*T for constant accelerations of an object over a time period

1. U is the initial velocity of the time period
2. V is the final velocity of the time period
3. A is the constant acceleration
4. T is the length of the time period

If one of V, U, A, or T is the unknown of the problem, the Marples' algorithm tries to assert the equation V=U+A*T by finding values in the "givens" of the problem for the other three variables. If it can only find values for one or two of the three it will then assert the equation, list the one or two values it has plus the original unknown of the problem as "given" and begin a new search for another (independent) equation with the variable it couldn't find as the unknown. And so the process continues until all the new unknowns can be determined from the givens of the problem.

The first part of the implementation of Marples' algorithm is a "focussing" algorithm that attempts to direct the Marples' algorithm to a set of equations relevant to reducing the "givens-goal difference". For example, if the final velocity V was the unknown in the problem situation above then a search would start to find out what V was: namely, the "final velocity" of a "particle or object" during a "time". When V was thus identified as the final velocity of an object during a time period, equations relevant to this situation would be identified first and a queue of these equations prepared and tested for possible given-goal reduction. Thus the Marples' algorithm would not search through all possible equations that had a V unknown but only those relevant to the particular unknown situation.

Consider now as examples of the running Marples' algorithm two problems from different areas of mechanics, a pulley and a distance/rate/time problem.

> A man of 12 stone and a weight of 10 stone are connected by a light rope passing over a pulley. Find the acceleration of this man. (Palmer and Snell, 1956).

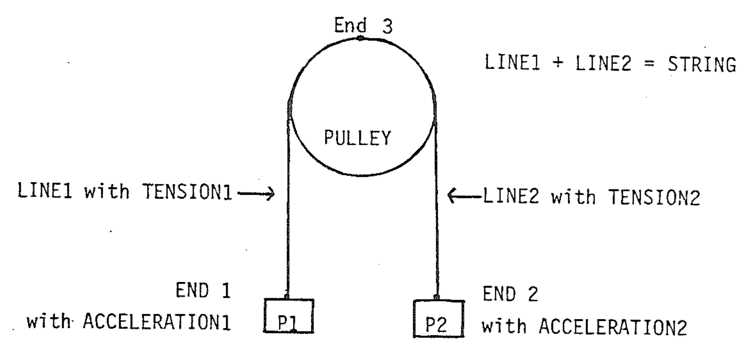
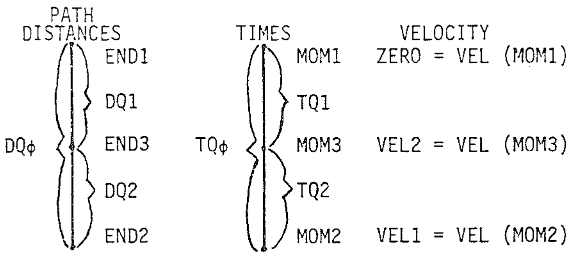


FIGURE 1 A representation of the entities created by the pulley schemata. TENSION1 = TENSION2 and ACCEL1 = ACCEL2 by schema inferencing.

First, two objects, the man and a weight, are identified as connected to the rope hanging over the pulley, each object is assigned a mass and an acceleration in a direction (c.f. Figure 1). The acceleration of the man, say A1, is identified as the sought unknown. The focussing algorithm first identifies the unknown A1 as the acceleration of the man during a time period. A1 is "bound" to the situation and a queue of possible equations examined that relate A1 to the givens. This list is examined and all equations rejected because they cannot solve for A1 without introducing new unknowns. The list is then reexamined and new unknowns allowed. The "resolution of forces" equation F=M*A (c.f. appendix) is the first in the queue and it is asserted. Originally A was unknown, M was found to be the mass of the man (known) but F, the sum of forces acting at the contact point of man and rope cannot be determined since the tension T1 in the rope is not known. Thus 12*g+T1=12*A1 is asserted A1, 12 and g are known and T1 is the new unknown. The focussing algorithm then identifies T1 as the tension in the string during the time period, a new queue of equations are proposed and the "resolution of forces" equation for the contact point of the rope and weight solves for T1 with no new unknowns. -10*g-T1=10A1 with 12*g+T1=12A1- are seen as sufficient to solve the pulley problem.

The tower problem is slightly more complex in that four simultaneous equations are needed to solve the problem:

A particle is dropped from the top of a tower. If it takes <u>t</u> seconds to travel the last <u>h</u> feet to the ground, find the height of the tower.



<u>FIGURE 2</u> The representation of the tower problem created by the schemata.

The situation is shown in Figure 2. The total height of the tower is H and H=H1+h where h is known. Velocity V0 is the initial velocity (0), V1 is the final velocity and V2 the velocity at the end of time T1 and the beginning of t.

H, the unknown, is identified as the total length of the path of the falling object, and the Marples' algorithm examines possible equations for finding H. The equations are given in the appendix, and the progress of the Marples' algorithm through the problem is given below. When each equation is asserted a new list of givens and unknowns is prepared and the Marples' algorithm and focussing is called again (in this problem four times).

1. given (g,t,h)
   unknown (H) - The total height

2. assert $H=0*T+\frac{1}{2}*g*T^2$ (No. 6, appendix)
   givens (g,t,h,H); unknown (T) - The total time
3. assert T=T1+t (No. 8, appendix)
   givens (g,t,h,H,T); unknown (T1) - The time of top period
4. assert V2=0+g*T1 (No. 5, appendix)
   givens (g,t,h,H,T,T1); unknowns (V2) - The velocity at midpoint
5. assert $h=V2*t*+\frac{1}{2}*g*t^2$ (No. 6, appendix)
   givens (g,t,h,H,T,T1,V2); unknowns ( )

The four equations above are seen as independent and sufficient to solve the problem.

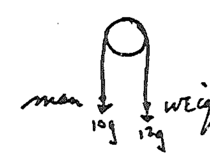
To conclude this section, the means-ends analysis implied in the Marples' algorithm may be compared to that of The General Problem Solver (Newell & Simon, 1963). The equations used by the Marples' algorithm are much like the "table of differences" used by GPS to reduce given-goal differences. The focussing technique prepares possible equations for the goal reduction just as GPS considers different given-goal combinations from the "table of differences". What is unique about MECHO, is that this is one of the first applications of means-ends analysis to solving problems in applied mathematics.

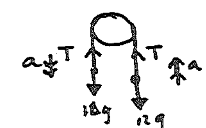
III. FOUR PROTOCOLS OF HUMAN SUBJECTS

In this section parts of four protocols of subjects solving the pulley and tower problems are presented. The subjects were post-graduate students at the University of Edinburgh and all had had some mechanics or applied mathematics in their undergraduate education.

<u>Protocol A (Pulley problem)</u>

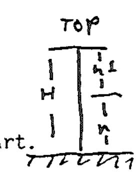
1. Mechanics problem ...
2. We'll treat the man and weight both as particles ... point masses ...
3. So man and weight ... Man has a force of 12 stone vertically downwards ...
4. Unknown at the moment ...
5. and assuming this frictionless pulley, T is the same on both sides ...
6. We'll give the rope an acceleration ... vertically down on the man's side ...
7. and vertically up on the weight's side ...
8. So resolving vertically down for the man: 12g - T = 12 a
9. The vertically downwards acceleration
10. And the vertically upwards for the weight T - 10g = 10a
11. so the thing requires the acceleration of the man which is <u>a</u> ...
12. So we just eliminate T from these two equations ... etc.



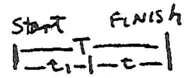


<u>Protocol B (Tower problem)</u>

1. There is a tower, suppose it has height H
2. H is made up of h, the given height
3. and h1, the unknown portion of the tower
4. We also know time t
5. Call the total time of falling T, T is made up of t plus t1 where t1 is the time for the top part.

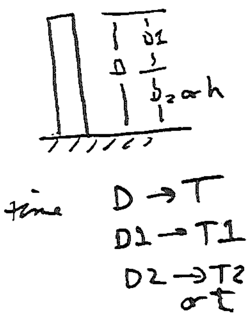


6. I need H and I'm given t and h and g the acceleration of the object
7. I know that H = h1 + h
8. Now to get h1, a distance
9. h1 equals one half acceleration times t1 squared $h1=\frac{1}{2}g*t1^2$
10. and I have an equation with t1 already T = t1 + t
11. but I still need to find T, the total time ...
12. Now, for the whole period, $H = \frac{1}{2}g \times t^2$
13. Reviewing, I have one, two, three, four equations (indicates each)
14. and H,T,h1 and t1 are unknowns that should do it ...



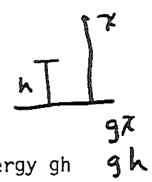
<u>Protocol C (Tower problem)</u>

1. T1 is the time the ball crosses D1
2. and T2 the time for D2, but T2 is t ...
3. Now I want D, knowing g is the acceleration of the ball $D = \frac{1}{2}g\ T^2$
4. and I know T : T = T1 + T
5. So now I need T1 ... in the top time period
6. The final velocity ... Call it VM ...
7. is acceleration times time VM = g x T1
8. and now to get VM ...
9. The total distance D ... That won't help ...
10. The distance h and velocities ...
11. if VF is final velocity VF = VM + g x t
12. and I can easily get the final velocity VF:
13. Because I've already got T ...



<u>Protocol D (Tower problem)</u>

1. Do this by energy ...
2. It initially starts with potential energy mg
3. where m is the mass of the body m is a constant
4. we can forget about m, call it 1
5. gx is the initial potential energy
6. when it reaches height h, it will have potential energy gh
7. and it will have kinetc energy gh
8. so it will have $gx - hx = \frac{1}{2}v^2$
9. where v is its velocity at height h
10. we don't know what height ... what speed it is once it hits the ground
11. we want to use one of the other constant acceleration ones $v^2 = u^2 + 2\ S$
12. we want to ... an equation relating height, velocity and time, that is
13. under constant acceleration ah which is $S = vt + \frac{1}{2}\ at^2$



IV. FOCUSSING

In Section II, the Marples' algorithm for MECHO's goal driven search was presented and compared to GPS means-ends analysis (Newell & Simon, 1963). MECHO's set of possible equations served as a table of connections for goal reduction and the focussing mechanism prepared a queue of possible equations to make the connections.

In the Pulley problem above the focussing produced immediate results. In fact MECHO, like the human solver performed very little search in coming to the Resolution of Forces formula. Two applications of this formula and the problem is solved. The Tower problem was much more interesting. In fact, there are several different sets of simultaneous equations (and of course permutations within each set) that will solve the problem. Each solver (B,C,D) produced a different set of equations and each of these sets differed from MECHO's trace of the previous section.

In this section the focussing technique and table of connections is altered in an attempt to produce traces similar to the human protocols of the tower problem. First note what will not be changed. The semanti knowledge for the Tower problem will not be changed. That is, facts suc as the unknown H the height of the tower and t and h the givens will be used without change by MECHO throughout this section; similarly V2 will remain the velocity at the "midpoint". Furthermore, the first part of focussing that binds situation variables for sought unknowns will remain unchanged. That is, H will be bound to the LENGTH of the PATH during the TIME. What will change is the queue of possible equation instantiations for situations and alterations will be made in the equations withi the table of connections. These latter will be seen shortly.

The 10 clauses below are used to form the queue of formulae to be attempted in any problem situation. Resolve, relative velocity, etc. refer to the equation names of the Appendix.

1. relates(resolve; FORCE, ACCELERATION, MASS).
2. relates(relative velocity; VELOCITY).
3. relates(relative acceleration; ACCELERATION).
4. relates(constant acceleration-1; ACCELERATION,VELOCITY,DURATION
5. relates(constant acceleration-2; ACCELERATION,LENGTH,VELOCITY, DURATION).
6. relates(constant acceleration-3; VELOCITY,LENGTH,DURATION).
7. relates(average velocity; VELOCITY,LENGTH,DURATION).
8. relates(constant velocity; VELOCITY,LENGTH,DURATION).
9. relates(lenthsum; LENGTH).
10. relates(timesum; DURATION).

After H is recognized as the LENGTH of the path during the episode by th first step in focussing, the second step prepares the queue by scanning the 10 clauses above to find which formulae will help solve for LENGTH. This proposes a queue of constant acceleration-2, constant acceleration- average velocity, and length sum. These equations are tried in that ord Each, of course, fails because new unknowns are introduced. On the seco pass, when new unknowns are allowed, constant acceleration-2 is accepted This process continues until the problem is solved.

We hypothesize that the human subject has a stack of equations that relate to specific situations. These equations are employed when attempti to reduce the given-goal differences in a specific problem. In fact, th stack might be quite similar to that produced by the 10 clauses above. This clause queue like the table of connections need <u>not</u> contain the ful equations, only their names. These names may serve to reference the actual equation formulae which are stored with their full sets of conditio for instantiation.

This conjectured implementation of the GPS model may be tested by making simple alterations in the 10 clauses above to see if the different queue of equations to be formulated can produce a different set of simultaneous equations. In particular, we attempt to produce traces similar to the human protocols B,C, and D.

Subject B used the length sum equation with top priority when solving for LENGTH, and timesum when a DURATION was sought. Thus, if clause 9 and 10 are placed before the constant acceleration clauses the order of "relates" above becomes 1, 2, 3, 9, 10, 4, 5, 6, 7, 8. When this re-arrangement of clauses was run in MECHO the following trace occurred:

B'
1. Attempting to solve for H in terms of g,h,t.
2. H=H1+h solves for H but introduces H1.
** H1 is a LENGTH, lengthsum cannot be used again, so constant acceleration-2 is used**
3. $H1=0xT1+\frac{1}{2}xg*T1^2$ solves for H1 but introduces T1
4. T=T1+t solves for T1 but introduces T
** constant acceleration-2 is now the first on the queue for T since timesum may not be used again, and MECHO always tries to solve without further introduction of unknowns**
5. $H=0*T+\frac{1}{2}*g*T^2$
6. Equations 2-5 solve the Tower problem.

If the ZERO term (initial vel. x time) is removed from 3 and 5 these equations are exactly those produced by subject B above.

In an attempt to produce a trace similar to protocol C, the "relates" clause (9) for lengthsum is returned to its original position, (i.e., 1, 2, 3, 10, 4, 5, 6, 7, 8, 9). The subject of protocol C does not use the constant acceleration-2 equation to its full potential, that is, he only uses the equation when the initial velocity is zero. Marples (1976) comments on this use of equations by engineering students when he notes they often apply an equation without knowing its full power. In this instance, the subject uses constant acceleration-2 for relating acceleration, time, and distance and not in its full use of relating initial velocity, acceleration, time, and distance. If it is conjectured that this happens with subject C, constant acceleration-2 equation is changed to this limited used by rewriting 5 above to "relates (constant acceleration-2; ACCELERATION,LENGTH,DURATION)" and removing "U*T" from the isformula clause of constant acceleration-2 c.f. appendix

MECHO is now run with these changes and the following trace results:

C'
1. trying to solve for H in terms of g,t,h.
2. $H=\frac{1}{2}*g*T^2$ solves for H but introduces T.
3. T=T1+t solves for T but introduces T1.
4. VEL2 = ZERO + g * T1 solves for T1 but introduces VEL2.
** Recall that the final velocity at the bottom is VEL1 and the velocity at the "midpoint" VEL2. The solver using constant acceleration-2 properly would now be done. Our subject grinds on**
5. VEL1 = VEL2 + g * t solves for VEL2 but introduces VEL1.
6. VEL1 = ZERO + g * T solves for VEL1.
7. Equations 2-6 solve the Tower problem.

This trace is remarkably similar to protocol C.

The subject of protocol D decided to use energy equations as was explicitly stated. This was not expected by the investigator taking protocols or designing MECHO to solve distance/rate/time problems. But in principle

there was no reason why energy equations could not be used. They were, in fact, already in the system and used to solve de Kleer's problems (Bundy, 1978). A new entry for the "relates" table was constructed: No. 11. relates (conserve energy; VELOCITY,LENGTH) and this was given priority over all other LENGTH relation clauses. Further, constant acceleration-3 is equivalent to conserve energy and was removed. Finally, subject D favored constant acceleration-2 over constant acceleration-1 formula for solving VELOCITY problems, so this order was changed. The "relates" list was 1, 2, 3, 4, 10, 11, 6, 8, 9, 5.

MECHO was run in this situation; its trace:

D'
1. Trying to solve for H in terms of g,t,h.
2. $\frac{1}{2} *VEL1^2 - \frac{1}{2} * ZERO^2 = g * H$ solves for H but introduces VEL1
** The energy equation is attempted again. This time to solve for VEL1**
3. $\frac{1}{2} * VEL1^2 - \frac{1}{2} VEL2^2 = g * t$ solves for VEL1 but introduces VEL2
4. $h = VEL2 * t + \frac{1}{2} * g * t^2$ solves for VEL2
5. 2-4 solve the Tower problem.

It can be seen that this trace is very close to the protocol of subject D above. Further comments will be made in the next section.

V. SUMMARY AND CONCLUSIONS

The goal of this paper has been to describe the action of the Marples' algorithm in MECHO's solution of pulley and distance/rate/time problems. After creation of a knowledge base, the Marples' algorithm was invoked and using means-ends analysis, in many ways similar to GPS, produced sets of simultaneous equations sufficient for solving the problem.

Experienced human subjects solving the same problems were presented with strategies for producing sets of simultaneous equations sufficient to solve a problem in many ways similar to those of the Marples' algorithm. Finally, with slight changes the Marples' algorithm could produce sets of equations almost identical to those produced by the human subjects.

In the pulley problem the protocol indicates the subject goes immediately to the resolution of forces equation. This is used to create the intermediate unknown of the tension in the string. Forces are again resolved at the other end of the string to solve for tension and two simultaneous equations are produced sufficient to solve the problem. The Marples' algorithm in MECHO proceeds in exactly the same fashion with the important difference that the resolution-of-forces equation, the first equation considered, is rejected because it introduces a new unknown (tension). All other equations in the queue trying to find acceleration for a particle in a time period are examined and rejected before the return to the resolution-of-forces equation and introduction of the tension as a new unknown. The difference between the experienced human and MECHO is obvious and interesting. The human realizes immediately a new unknown must be introduced, accepts it, and goes on, while MECHO tries as long as possible to avoid this course of action. Other than this, the protocol of the human and MECHO's trace are remarkably similar.

The similarities between trace and protocol are even stronger with the tower problem where many different sets of simultaneous equations sufficient for solving the tower problem may be produced. The formation of any set of these equations depends entirely on the use of the focussing technique, that is, the queue that is generated for possible equations. The similarity of protocol C with the trace of C' of Section IV is striking. Indeed, when it was hypothesized that if the human subject had been able to use initial velocity values in constant acceleration-2 equations when the initial velocity was not zero and had a slightly altered focus, then MECHO's trace would match exactly the protocol of subject C. In section IV, these alterations were made and the resulting protocols compared. The results indicate the robustness of the Marples' algorithm and focussing to generate different sets of simultaneous equations to fit closely the traces of the human subjects.

One of the principal advantages of writing the MECHO problem solver in PROLOG (Warren, 1978) here represented by predicate logic assertions, is that PROLOG actually computes using the predicate logic statements themselves. In fact, PROLOG was designed as a predicate logic theorm prover. Thus facts, inferences, and default values are entered into the program as the potential for "meaningful bits of behavior". This allows such actions as removing a rule from the program, substituting another rule, or simply changing the order of the rules and then checking the results of these changes on the running computer program. Thus a PROLOG "fact", "inference rule", or "default assignment" may be paired with the corresponding competency in the human subject and the effect of its presence or absence in the human subject may be simulated by the running program. This can be seen when the "conservation of energy" equation was added for solution of the Tower problem (IV). The new rule added resulted in the exhibition of a new competency, and conversely, the absense of the rule marked the absense of the related ability.

A modular set of rules also allows general purpose algorithms, such as the Marples' algorithm, to be implemented and the effects of the presence of the algorithm to be seen by running the program. In a very similar fashion Larkin (1978) and Simon and Simon (1977) can run sets of production rules in an attempt to simulate the difference of skills in the expert or novice problem solver.

The presence of production or behavior rules also provides a model for the interpretation of missing or ambiguous behavior of the human subject. In D, for example, "Do this by energy" indicates an energy equation will be called to find the value for H. D5 states "gx is the initial potential energy" ... and D 6-7 "when it reaches height h it will have potential energy gh ..." and finally in D 8-9 "so it will have gx - gh = $\frac{1}{2}V^2$ where V is the velocity at height h ..." - what does this all mean? The protocol D2 to D10 is at best confusing. Reading MECHO's trace on the same problem goes a long way towards making sense of these statements. D'2 gives a full description of gx: "g * H = $\frac{1}{2}$ $VEL1^2$", where g * H is gx and VEL1 the final velocity. Similarly for gh: D'2 has "g * h + $\frac{1}{2}$ $VEL1^2$ - $\frac{1}{2}$ $VEL2^2$, when VEL2 is the velocity at the top of h. Now simply subtract D'3 from D'2 and line D9 of the protocol is precisely understood. The subject of protocol D was a particularly bright individual that liked to skip steps and simplify as he went along. Although MECHO is not able to imitate this behavior completely - especially the subject's proclivity for short slightly ambiguous statements - its rule system does give a precise and complete performance and often provides data sufficient to disambiguate the human subject's behavior.

There are, of course, many things that MECHO, as presently designed, does not do that human subjects do quite easily. We have already seen how sub-

ject D simplified as he went along and omitted "unnecessary" bits of equations. Similarly, other subjects - this could be easily added to MECHO- left out terms of equations that were zero. Also MECHO is exhaustively thorough in its search while human subjects are not, and this MECHO will never use five equations where four are sufficient. However, as noted in Section IV, MECHO can offer an explanation of precisely why a subject needed five equations when four would have been sufficient.

References

1. Bundy, A. Will it Reach the Top? Prediction in the Mechanics World. Artificial Intelligence (1978), 10, 111-22.

2. Bundy, A., Luger, G., Mellish, C. & Palmer, M. Knowledge about Knowledge: Making decisions in mechanics problem solving. Proceedings of AISB Conference - 1978.

3. Ernst, G. & Newell, A. GPS: A Case Study in Generality and Problem Solving. New York: Academic Press, 1969.

4. Marples, D. Argument and Technique in the Solution of Problems in Mechanics and Electricity, Cambridge (UK) CUED/C, EDUC/TRI, 1974.

5. Newell, A. and Simon, H. GPS: A Program that Simulates Human Thought. In Computers and Thought, Feigenbaum & Feldman (ED.), New York: McGraw Hill, 1963.

6. Newell, A. & Simon, H. Human Problem Solving, Englewood-Cliffs, N.J.: Prentice-Hall, 1972.

7. Larkin, J.H. Skill Acquisition for Solving Physics Problems. C.I.P. Report #409, Dept. of Psychology, Pittsburgh: Carnegie-Mellon University, 1978.

8. Palmer, A. & Snell, K. Mechanics. London: University of London Press, 1956.

9. Polya, G. How to solve it. Princeton: Princeton University Press, 1945.

10. Simon, H. and Simon, D. Individual Differences in Solving Algebra Problems, CIS #342, Dept. of Psychology, Pittsburgh: Carnegie-Mellon University, 1977.

11. Warren, D. Implementing PROLOG - compiling predicate logic programs. DAI Tech. Report. University of Edinburgh, 1978.

I would like to thank Alan Bundy, Lawrence Byrd of the MECHO group and Richard Yound of APU - Cambridge for encouragement in writing this paper and criticism of earlier drafts and the British SRC for supporting this research.

Appendix: PROLOG clauses for possible equation formation in Pulley and Tower problems

Note that words with first letter capitalized, e.g., Object or Direction, as well as single capitol letters, e.g. M or T1, represent variables.

1. isformula (F = M * A, resolve (Object, Direction, Time)):- mass (Object, A, Direction, Time), acceleration (Object, A, Direction, Time), sumforces (Object, Direction, Time, F).
2. isformula (V13 = V123, relative velocity (Obj1, Obj2, Obj3, Time)):- relative velocity (Obj1, Obj2, V12, Dir12, Time), relative velocity (Obj1, Obj3, V13, Dir13, Time), vectoradd (V12, Dir12, V23, Dir23, V123, Dir13).
3. isformula (A123 = A123, relative acceleration (Obj1, Obj2, Obj3, Time)):- relative acceleration (Obj1, Obj2, A12, Dir12, Time), relative acceleration (Obj2, Obj3, A23, Dir23, Time), relative acceleration (Obj1, Obj3, A13, Dir13, Time), vectoradd (A12, Dir12, A23, Dir23, A123, Dir13).
4. isformula (S = V * T, constant velocity (Object, Time)):- constant velocity (Object, Time), velocity (Object, V, Direction, Time), duration (Time, T), distance (Object, S, Time).
5. isformula (V = U + (A *T), constant acceleration1 (Object, Time)):- constant acceleration (Object, Time), duration (Time, T), initial velocity (Object, U, Direction, Time), final velocity (Object, V, Direction, Time).
6. isformula (S = U * T + (A * (T:2)/2), constant acceleration2 (Object, Time)):- constant acceleration (Object, Time), acceleration (Object, A, Direction, Time), duration (Time, T), initial velocity (Object, U, Direction, Time), distance (Object, S, Time).
7. isformula ((V:2) + (U:2) = 2 * A * S, constant acceleration3 (object, Time)):- constant acceleration (Object, A, Direction, Time), distance (Object, S, Time), initial velocity (Object, U, Direction, Time), final velocity (Object, V, Direction, Time).
8. isformula (T = Sum, timesum (Time)):- Partition (Time, Points), duration (Time, T), sumdurations (Points, Sum).
9. isformula (D = Sum, lengthsum (Path, Time)):- partition (Path, D, Time), sumlength (Points, Sum, Time).
10. isformula (V:2)/2 - (U:2)/2 = g * H, energy (Object, Time)):- motion (Object, Path, Start, Side, Time), incline (Path, 270, Time), length (Path, H, Time), final velocity (Object, V, Direction, Time), initial velocity (Object, U, Direction, Time).

** g is known by MECHO as the gravitational constant **