A HEURISTIC PROGRAM THAT SOLVES SYMBOLIC INTEGRATION PROBLEMS IN FRESHMAN CALCULUS

by James R. Slagle

A large high-speed general-purpose digital computer (IBM 7090) was programmed to solve elementary symbolic integration problems at approximately the level of a good college freshman. The program is called SAINT, an acronym for "Symbolic Automatic INTegrator." The SAINT program is written in LISP (McCarthy, 1960), and most of the work reported here is the substance of a doctoral dissertation at the Massachusetts Institute of Technology (Slagle, 1961). This discussion concerns the SAINT program and its performance.

Some typical samples of SAINT's external behavior are given so that the reader may think in concrete terms. Let SAINT read in its card reader an IBM card containing (in a suitable notation) the symbolic integration problem $\int xe^{x^2}\,dx$. In less than a minute and a half, SAINT prints out the answer, $\frac{1}{2}e^{x^2}$. Except where otherwise noted, every problem mentioned in this chapter has been solved by SAINT. Note that SAINT omits the constant of integration, and we, too, shall ignore it throughout our discussion. After working for less than a minute on the problem $\int e^{x^2}\,dx$ (which cannot be integrated in elementary form) SAINT prints out that it cannot solve it.

SAINT performs indefinite integration, also called antidifferentiation. In addition it performs definite and multiple integration when these are trivial extensions of indefinite integration. SAINT handles integrands that represent explicit elementary functions of a real variable which, for the sake of brevity, will be elementary functions. The elementary functions are the functions normally encountered in freshman integral calculus, except that SAINT does not handle hyperbolic notation. The elementary functions are defined recursively as follows:

a. Any constant is an elementary function.
b. The variable is an elementary function.
c. The sum or product of elementary functions is an elementary function.
d. An elementary function raised to an elementary function power is an elementary function.
e. A trigonometric function of an elementary function is an elementary function.
f. A logarithmic or inverse trigonometric function of an elementary function (restricted in range if necessary) is an elementary function.

Currently SAINT uses twenty-six standard forms. It uses eighteen kinds of transformations including integration by parts and various substitution methods (but excluding, among others, the method of partial fractions). Since the SAINT program uses heuristic methods, it is by definition a heuristic program. Although many authors have given many definitions, in this discussion a heuristic method (or simply a heuristic) is a method which helps in discovering a problem's solution by making plausible but fallible guesses as to what is the best thing to do next.

Indefinite Integration Procedure of SAINT

This section describes how SAINT performs indefinite integration. An attempt is made to orient the reader before a detailed description of the procedure is given. The executive organization of SAINT is like that of the Logic Theorist of Newell, Shaw, and Simon (1957). It will help to take a preview of Sec. 14 (especially Fig. 3). The "try for an immediate solution" mentioned twice in Fig. 3 may be described roughly as follows: As soon as a new goal g is generated, SAINT uses its straightforward methods in an attempt to achieve it. While doing this, SAINT may add g or certain of g's subgoals to the "temporary goal list." If g is achieved, an attempt is made to achieve the original goal. Slagle (1961) includes among other things, a full description together with a detailed example and suggestions for future work.

As a concrete example we sketch how SAINT solved

$$\int \frac{x^4}{(1 - x^2)^{5/2}}\, dx$$

in eleven minutes. SAINT's only guess at a first step is to try substitution: $y = \arcsin x$, which transforms the original problem into

$$\int \frac{\sin^4 y}{\cos^4 y}\, dy$$

For the second step SAINT makes three alternative guesses:

A. By trigonometric identities $\int \frac{\sin^4 y}{\cos^4 y}\, dy = \int \tan^4 y\, dy$

B. By trigonometric identities $\int \frac{\sin^4 y}{\cos^4 y}\, dy = \int \cot^{-4} y\, dy$

C. By substituting $z = \tan (y/z)$ $\int \frac{\sin^4 y}{\cos^4 y}\, dy = \int 32 \frac{z^4}{(1+z^2)(1-z^2)^4}\, dz$

SAINT immediately brings the 32 outside of the integral.

After judging that (*A*) is the easiest of these three problems SAINT guesses the substitution $z = \tan y$, which yields

$$\int \tan^4 y\, dy = \int \frac{z^4}{1+z^2}\, dz$$

SAINT immediately transforms this into

$$\int \left(-1 + z^2 + \frac{1}{1+z^2}\right) dz = -z + \frac{z^3}{3} + \int \frac{dz}{1+z^2}$$

Judging incorrectly that (*B*) is easier than

$$\int \frac{dz}{1+z^2}$$

SAINT temporarily abandons the latter and goes off on the following tangent. By substituting $z = \cot y$,

$$\int \cot^{-4} y\, dy = \int -\frac{dz}{z^4(1+z^2)} = -\int \frac{dz}{z^4(1+z^2)}$$

Now SAINT judges that

$$\int \frac{dz}{1+z^2}$$

is easy and guesses the substitution, $w = \arctan z$ which yields $\int dw$. Immediately SAINT integrates this, substitutes back and solves the original problem.

$$\int \frac{x^4}{(1-x^2)^{5/2}}\, dx = \arcsin x + \tfrac{1}{3} \tan^3 \arcsin x - \tan \arcsin x$$

The indefinite integration procedure may be described as follows:

1. Goals

In each application of the present procedure, the solutions of certain problems, namely, performing integrations with side conditions, are goals. How goals are generated, manipulated, and achieved is described later. For now, let us limit ourselves to describing what we shall call the "original goal," which consists of the originally given integrand and variable of integration.

2. The Goal List

The original goal is made the first member of a list called the goal list. From time to time new goals may be generated. Each newly generated goal is added to the end of the goal list.

3. Standard Forms

Whenever an integrand of a newly generated goal is of "standard form," that goal is immediately achieved by substitution. An integrand is said to be of standard form when it is a substitution instance of one of a certain set of forms. For example, $\int 2^x\, dx$ is an instance of $\int c^v\, dv = c^v/(\ln c)$ and hence has the solution $2^x/(\ln 2)$. Currently SAINT uses twenty-six standard forms (Slagle, 1961).

4. Algorithmlike Transformations

Whenever an integrand is found to be not of standard form, it is tested to see if it is amenable to an algorithmlike transformation. By an algorithmlike transformation is meant a transformation which, when applicable, is always or almost always appropriate. For a goal, a transformation is called appropriate if it is the correct next step to bring that goal nearer to achievement. Three of the eight algorithmlike transformations used in SAINT are:

a. Factor constant, *i.e.*,

$$\int cg(v)\, dv = c\int g(v)\, dv$$

b. Decompose, *i.e.*,

$$\int \Sigma g_i(v)\, dv = \Sigma \int g_i(v)\, dv$$

c. Linear substitution, *i.e.*, if the integral is of the form

$$\int f(c_1 + c_2 v)\, dv$$

substitute $u = c_1 + c_2 v$, and obtain an integral of the form

$$\int \frac{1}{c_2} f(u)\, du$$

for example, in

$$\int \frac{\cos 3x}{(1 - \sin 3x)^2}\, dx$$

substitute $y = 3x$.

5. The AND-OR Goal Tree

When a heuristic transformation (to be described in Sec. 11) or an algorithmlike transformation is applied to a goal, new goals are generated. These goals, in turn, may generate more goals, and a certain hierarchy is created. Such a hierarchy is conveniently represented by a graph or tree growing downward. To facilitate understanding, the terminology of ordinary and family trees is adopted by analogy, for example pruning, alive, dead, child, parent, descendant, ancestor, etc.

Suppose we have an integration to perform, or more generally, any goal g, which we shall represent graphically by a point. A goal may be transformed into one or more subgoals which may be related to the goal in many ways. This integration procedure incorporates two common relations, namely AND and OR.

a. AND relationship

An AND relationship between a goal and at least two subgoals exists when the achieving of all of the subgoals causes the achieving of the goal. Figure 1 depicts a relationship with three subgoals. The arc joining the three branches denotes the AND relationship.

b. OR relationship

An OR relationship between a goal and its subgoals exists when the achieving of any one of the subgoals causes the achieving of the goal. Examples of this will appear later.

From these two basic relationships, more complicated relationships among goals may be built up; for example, see Fig. 2*a* and *b* in Sec. 12.

6. The Temporary Goal List

The first attempt on new goals is performed by the procedures "imsln" ("IMmediate SoLutioN") described in Sec. 13 below. Any goal en-

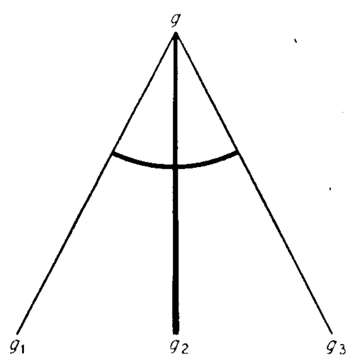

Figure 1. An AND relationship.

countered by imsln which is neither of standard form nor amenable to an algorithmlike transformation is added to the end of the "temporary goal list" (not to be confused with the "goal list") and later transferred to the "heuristic goal list" described in Sec. 10 below. If the goal were added directly to the heuristic goal list rather than to the temporary goal list, time might be wasted by finding the goal's character (cf. Sec. 8).

7. The Role of the Resource Allotment

The resource allotment, or the total amount of work space, is a side condition of the original goal. Before proceeding to apply heuristic transformations, it must be verified that the resource allotment has not been exceeded. If the resource allotment has been exceeded, SAINT reports this fact as its final answer. Although other kinds of resources, for example, time, might also be considered, the only kind of resource allotment handled by SAINT is the total amount of work space. For hand simulation, the work space can be measured by the number of pages or by the number of lines used for the final and all intermediate results.

8. Character of a Goal

When a goal is taken off the temporary goal list its "character" is obtained, that is, an ordered list of "characteristics." A characteristic of a goal is a feature which might be useful either in estimating the cost of attempting its attainment or in selecting appropriate heuristic transformations (see Sec. 11). In SAINT, the character is composed of eleven characteristics of the integrand (Slagle, 1961) including its function type (whether it is a rational function, algebraic function, rational function of sines and cosines, etc.) and its depth. The depth of an integrand is the maximum level of function composition which occurs in that expression:

$$\begin{aligned} x &\text{ is of depth } 0, \\ x^2 &\text{ is of depth } 1, \\ e^{x^2} &\text{ is of depth } 2, \\ xe^{x^2} &\text{ is of depth } 3. \end{aligned}$$

As one might guess, this helps get a crude estimate of the problem's difficulty.

9. Relative Cost Estimate

Although other estimates could be tried, for the relative cost estimate of a goal we take simply the depth of its integrand. This makes use of the fact that, ordinarily, the deeper the integrand the more will be the resources needed to investigate that goal.

10. The Heuristic Goal List

A list of goals requiring heuristic transformations, or, more briefly, a heuristic goal list, is an ordered list of those goals which are neither of standard form nor amenable to an algorithmlike transformation. A member of the heuristic goal list is called a heuristic goal. New such goals are inserted in order of increasing relative cost estimate.

11. The Heuristic Transformations

A transformation of a goal is called heuristic when, even though it is applicable and plausible, there is a significant risk that it is not the appropriate next step. A transformation may be inappropriate either because it leads no closer to the solution or because some other transformation would be better. The heuristic transformations are analogous to the methods of detachment, forward chaining and backward chaining in the Logic Theorist of Newell, Shaw, and Simon (1957). The ten types of heuristic transformation (Slagle, 1961) used by SAINT are designed to suggest plausible transformations of the integrand, substitutions, and attempts using the method of integration by parts. Below is given only the most successful heuristic, "substitution for a subexpression whose derivative divides the integrand."

Let $g(v)$ be the integrand. For each nonconstant nonlinear subexpression $s(v)$ such that neither its main connective is MINUS nor is it a product with a constant factor, and such that the number of nonconstant factors of the fraction $g(v)/s'(v)$ (after cancellation) is less than the number of factors of $g(v)$, try substituting $u = s(v)$. Thus, in $xe^{x^2}\,dx$, substitute $u = x^2$. (When SAINT tried this problem it used this heuristic but surprised me by substituting $u = e^{x^2}$, which is somewhat better.)

12. Pruning the Goal Tree

Whenever some goal g has been achieved, the goal tree is pruned, that is, certain closely related goals are automatically achieved and certain other goals, newly rendered superfluous, are killed.

The pruning procedure will be clarified by an example. In Fig. 2*a* the achieving of g_{221} allows g_{22} to be achieved (since, as indicated by the black dot, g_{222} has already been achieved). In turn, the achieving of g_{22} allows g_2 to be achieved (since there is an OR relationship). Since the achieving of g_2 now has rendered g_{23} superfluous, it is killed. However, another of g_2's children g_{12} is not killed since, through its other parent g_1 it has direct living ancestry to the original goal g. The original goal g cannot be achieved from the achieving of g_2 since there is an AND relationship and g_1 has not yet been achieved. Therefore, the result of the pruning process is as shown

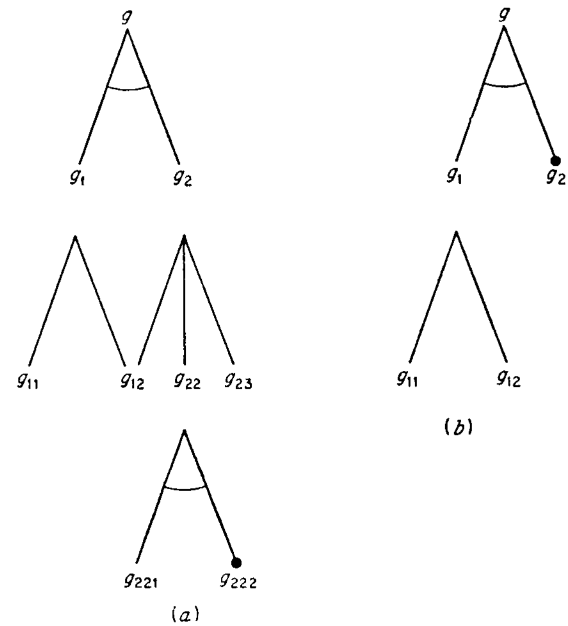

Figure 2.

in Fig. 2*b*. If either g_{11} or g_{12} is later achieved, the original goal could and would be achieved.

13. Trying for an Immediate Solution

As soon as a new goal g is generated, SAINT uses its straightforward methods in an attempt to achieve it. While doing this, SAINT may add g or certain of g's subgoals to the temporary goal list. If g is achieved, an attempt is made to achieve the original goal.

14. Executive Organization

Precisely how all the various elements 1 through 13 are pieced together to form an integration procedure is described below. The original goal is given as a triplet, namely, the integrand, the variable of integration, and the resource allotment. The procedure (see Fig. 3) is as follows:

a. If a try for an immediate solution with the original goal is successful, return with the answer, the actual indefinite integral.

b. If the resource allotment has been exceeded, report failure.

c. Obtain and associate with each goal on the temporary goal list its character and relative cost estimate. Take the goals off the temporary goal list, and insert each one in the heuristic goal list according to its relative cost estimate. If no goals remain on the heuristic goal list, report failure.

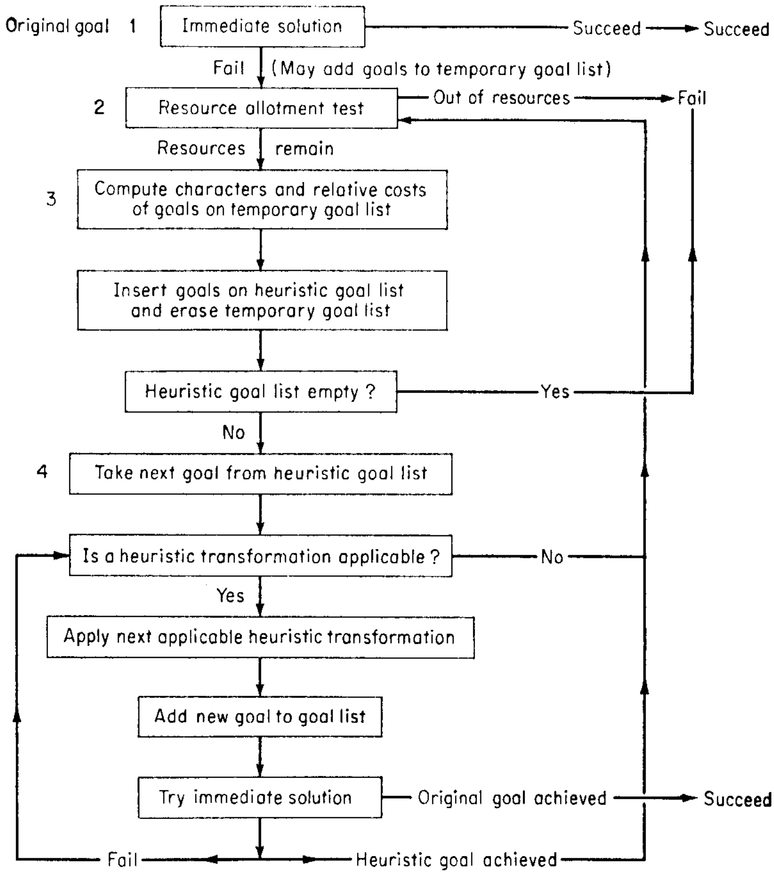

Figure 3.

d. Take the next goal g_i off the heuristic goal list and let it be the goal under consideration in the following inner loop.

e. If no heuristic transformations applicable to g_i remain, go to step *b*.

f. Apply the next heuristic transformation applicable to g_i. As soon as a new goal g is so generated, add it to the goal list, and try for an immediate solution with g. Then there are three cases. If this try achieves the original goal, return with the answer. Failing this, if g_i is achieved, go to step *b*. Otherwise go to step *e*.

Definite Integration Procedure

SAINT can perform some definite integrations by first finding the corresponding indefinite integrals. Thus, for example, for the problem

$$\int_0^3 x\sqrt{x^2 + 16}\, dx$$

SAINT first finds the indefinite integral,

$$\int x \sqrt{x^2 + 16}\, dx = \tfrac{1}{3}(x^2 + 16)^{3/2}$$

SAINT substitutes the limits and obtains the answer $^{61}/_3$.

Multiple Integration Procedure

SAINT can perform multiple integration when it can perform the required definite integrations, *e.g.*,

$$\int_1^1 \int_{y^2}^{2-y^2} dx\, dy$$

Experiments and Findings with SAINT

This section describes some of SAINT's typical observed behavior and how one modification changes its behavior. Slagle (1961) describes other experiments and gives further details. The experiments to measure SAINT's behavior involve 86 problems. Largely for the purposes of debugging, 32 of the problems were selected or constructed by the author, who fully expected SAINT to solve them all. More objectively, the remaining 54 problems were selected from MIT freshman calculus final examinations by the author's assistant, Gerald Shapiro, with instructions to select the more diverse and difficult problems, provided only that the method of partial fractions was not needed for the solution. The measures of behavior that we use are:

1. Power
The power of a version of SAINT refers to the size of the class of problems that it can solve.
2. Time
All the times mentioned refer to the IBM 7090 computer.
3. Number of subgoals and unused subgoals
The original goal is not included in the number of subgoals. An unused subgoal is a subgoal which is not needed in the solution chain.
4. Level
The level of a solution is the maximum level at which a used subgoal occurs in the goal tree during that solution.
5. Heuristic level of a solution
This measure is similar to "level" except that only the goal-tree branches representing heuristic transformations are considered rather than all the branches representing algorithmlike or heuristic transformations.

A. Unmodified SAINT

The SAINT program described in the preceding sections tried to solve all 86 problems selected by the author and Gerald Shapiro. In this attempt, the computer spent about half of its time in reclaiming abandoned memory registers for reuse. Approximately half of the remaining time was spent in pattern recognition, that is, in finding characters and in recognizing when an integrand is of standard form or amenable to an algorithmlike or heuristic transformation. As the author expected, SAINT solved all 32 of his problems. Of the 54 MIT problems, SAINT solved 52 and quickly (in less than a minute) reported failure for the other 2. Both of the failures are excluded from the averages in the table below, which summarizes SAINT's average performance.

SAINT's Average Performance

	Minutes	Subgoals	Unused subgoals	Level	Heuristic level
32 author problems	3.3	6.4	2.0	3.5	1.0
52 MIT problems	2.0	4.7	0.8	2.9	0.8
All 84 problems	2.4	5.3	1.25	3.0	0.9

In this paragraph we complement the tabulation of SAINT's average performance on MIT problems with some examples of SAINT's extreme behavior. For this purpose, only MIT problems are considered since they were selected more objectively. SAINT seemed to find $\int_1^2 (dx/x)$ the easiest problem since it generated no subgoals at all and took the least time, namely 0.03 minute. SAINT took the most time, 18 minutes, for

$$\int \frac{\sec^2 t}{1 + \sec^2 t - 3 \tan t}\, dt$$

whose solution ties for the maximum heuristic level of four. The other problem whose SAINT solution has a heuristic level of four is

$$\int \frac{x^4}{(1 - x^2)^{5/2}}\, dx$$

The maximum heuristic level obtained by the unmodified Logic Theorist is two, which occurred for two of 38 solutions (Newell, Shaw, and Simon, 1957*a*). SAINT generated the most subgoals (18) and had the maximum level (8) for $(\sin x + \cos x)^2\, dx$. In 37 of the 52 problems, SAINT generated only subgoals that were needed in the solution chain. In this regard, SAINT registered its best performance on one of these 37 problems

$$\int_{5/2}^{4} \int_{x(11-2x)}^{4(x-1)(5-x)} dy\, dx$$

for which SAINT generated 16 subgoals, all of which were needed in the solution chain.

B. BSAINT, *i.e.*, SAINT Trying Heuristic Goals in Order of Generation

Instead of trying heuristic goals in order of increasing depth, BSAINT tries heuristic goals merely in the order in which they were generated. In measures of performance including time and the number of unused subgoals, SAINT was better than BSAINT in three of the four problems which caused a difference in behavior.

Main Conclusions

The conclusions are based on experience, namely, the experiments described in the preceding sections and the author's experience concerning the creation, structure, and performance of SAINT. Throughout this section, a parenthesized mention of an experiment is an appeal for support of a conclusion to an experiment described in the previous section.

1. A machine can manifest intelligent problem-solving behavior, that is, behavior which, if performed by people, would be called intelligent (experiment *A*).
2. A heuristic program can easily include programs for handling an AND-OR goal tree (such as found in SAINT), which is often useful in complex goal-achieving schemes.
3. In SAINT, pattern recognition plays a very important part in three senses.
    a. Pattern recognition consumes much of the program and programming effort.
    b. Pattern recognition is used frequently and with great variety, for instance, in determinations involving standard forms, algorithmlike and heuristic transformations, and relative cost estimates.
    c. Pattern recognition consumes much of the time in solving integration problems (experiment *A*).
4. The tripartite division of methods into standard forms, algorithmlike transformations, and heuristic transformations is very useful in problem-solving. Standard forms in SAINT and "substitution" in the Logic Theorist may be instances of an "immediately achieve" procedure which seems to be a basic component of a goal-achieving scheme. The input to the procedure is a goal. The output is "no" (the goal is not yet achieved) or one or more of the following three items, namely, "yes," how to achieve the goal, or the achievement of the goal. In each domain, the procedure for immediately achieving a goal must be supplied anew and, since it is a very frequently used procedure, should operate very rapidly. The algorithmlike transformations also seem to be a basic component of a goal-achieving scheme, but this remains to be seen since they are not present in all schemes, for ex-

ample, the Logic Theorist. The organization of SAINT's heuristic transformations (corresponding to that of the Logic Theorist's methods of detachment, forward chaining, and backward chaining) seems to be an often convenient but not a basic component of a goal-achieving scheme.

5. A fourfold increase in SAINT's memory size (now 32,768 registers) could have been readily converted into a hundredfold increase in speed, since the reclaiming of abandoned memory registers for reuse, which now accounts for about half of the running time, would become insignificant and since a compiled program would run about fifty times faster. Much computer time and space could be saved if one computer instruction represented the very frequently used symbol manipulating functions.

6. The present speed of SAINT compares very favorably with the speed of the average college freshman (experiment A). With a now commercially available large high-speed digital computer, such as the IBM 7030 (STRETCH), a compiled but otherwise unimproved SAINT program would run eight hundred times faster, which would far surpass in speed even the most gifted of mathematicians at this task. At present commercial rates, an IBM 7090 SAINT solution of an average MIT final-examination problem costs about fifteen dollars, far more expensive than a human solution. However, a STRETCH SAINT solution would cost only about two dollars or, if compiled, only about four cents. This rapidly decreasing cost trend in computers, not to mention possible improvements in the SAINT program, will result in solutions which are far cheaper by machine than by man.