Ap 8, 64 TMY

Plan (788)

.01 : 787.40 : SN General Induction Machine

This machine will be at first for "study probs", but it will be → any type of induction prob. can be studied.

Some basic types that I'm most interesd in:

a) Learning Eng → SL or any MT.

b) " How to prove thrms.

c) " to "simplify" derns (i.e. ↑ pcost) in various ways.

d) " to do any probs. that Newton's GPS can solve

.15 e) " descriptor assignment for IR. This is done by lo[oking] at both th. name of th. "descriptor" and th. set of "relevant" and "irrelevant" docs. Again, I will try to get TM to do it somewhat th. way I do it.

Also consider th. prob. of TM's learning to invent descriptors to be used on a certain equiv. of prob.

Th. machine will have means (for most types of probs) for putting in corpus pairs (I, O pairs). Th. prob. th[at] for TM, is to get a set of probable O's for a

~~There will be mean~~ TM_i's "state" at any time, wi[th] a ppm. or "op."

There will be means for "telling" TM_i poss. solns. that be inducable from th. past corpus. Also TM_i can be asked of any new op. (so an estimate of search time could be made) — also th. sample cases that were rele[vant]

This will be true for any value of i.

~~There will be~~

TM_{i+1}'s job $(i \geq 1)$ is to speed up (or "speed up") TM_i. Th. Thrm. proving prob. (and GPS probs.) are mainly done mainly by TM_2 and TM_i $(i \geq 2)$.

For a "first study prob" it would seem to be bet[ter to] do most of th. work, rather than ...

TMγ TMγ TMγ

Th Ap 9, 64

289

(See 734, 730 for various TM probs. Also TMB 105

288.40 ; Possl. induction probs. & pros, cons:

1) Thrm. proving : (con) Is a ~~[illegible]~~ TM₂ problem.

(pro) (a) I have lots of ideas (b) work of Simon, Newell and others is usable

(c) TM₂ is usable for any TM, and I must work on it eventually any way.

2) IR from (a) requests in Eng., or in a synthetic lang.
a/o (b) descriptor assignment using sample of "rel." and "irrel" classes of docs

[See 788.15: for more details of this.

(pro) Useful in itself for AF, MIH. — is relevant to IR, and English processing, which I'll have to do eventually. is TM₁ prob (entire

(con) I really can't get a very good soln. w.o. TM really learning to understand English. Also, it isn't clear that it would be able to give TM good "suggs." — and one of the big things I want to do in the "study prob." is see how well I can take my intuitive heurs. and give them to TM.

3). Music prediction a/o writing — O.K., except I know little about music

4) Arith. Learning : (pro) largely TM₁

(con) I don't have a good idea for a good trg. seq. — Solving looks like a GPS-type prob., i.e. TM₂, rather than TM₁.

5) Chess, Checkers : try to reproduce moves that eg. I would make (from suitable trg. seq.) Later, try to reproduce moves of "masters". As the [illegible] for this is being made, I think means for tremendous ↑ in skill will become clear.

(pro) clearly a TM₁ prob.; Much previous work on mechanics of getting machine, etc. ; I have a fair intuitive feel for this.

(con) The heurs obtained wouldn't be very general.

4th Ap 9, 64 TMγ

Plan (290)

789.4.0 : My impression is that GPS-type probs (Thm. proving, Symb. integration, would be best to start on as a study prob. These are characterized by one being required to find an object that satisfies a certain criterion. The general TM_2 prob. would seem to be of this type.

GPS could be a TM_2.

TM_2's prob. is to find a "min code" for TM_1's corpus in "min time".

Superficially, it would seem that TM_1's work is essentially trivial — even in "Arith. Learning" or "Eng → SL learning" — That the real interest, as in GPS, is in TM_2. — Which would seem to indicate that I might as well work on GPS-type probs.

In GPS, the goal is to construct a string satisfying a certain property.

In TM_2, the goal is to find regs. in TM_1's corpus. Such a reg. will manifest itself as a ↑ in pcost of coding this corpus. — So the goal is to find a string that is a code of the corpus, that has a pcost as large as possl. Or — just to find as many regs in the corpus as possl., with each reg. being as "good" (in the sense of ↑ pcost) as possl.

Th Ap 9, 64 TM8

Plan ← This continuation from 780.40 is unclear. There is one likely page 780, & one less likely one. It is possl. that there is a third page 780 that this really refers to!
780.40 spac
: 780.40 → I will have to do it eventually, some Eng → SL MT has been done by others.

(con) I don't understand linguistics very well. — I'd have to invent a SL (or select and improve an existing one) { Loglan? Sc. Amer. Jun 1960

TN In Eng → SL "discourse analysis", I can just use a sent. by-xltn., then figure out what th. pronouns refer to in SL!
— Since SL is more amenable to logical xfmns: xfmn, ifmn.

→ A Big Trouble with Eng → SL MT as a prelim. to full QA., is that interesting meanings, like figs. of speech (765.10ff) must be induced from th. raw text — then included in th. Eng → SL MT. → Is there any way to do th. Eng → SL MT ∋ th. figs of speech" can be induced from th. resultant SL corpus?

Well, TM can get most of th. meaning by Eng → SL (using no figs. of speech xltn) then looking for regs. in th. SL corpus. After TM has learned to answ. Q's this way (i.e. in 2 elistic steps) he can look for less el. rules for (faster) Q answering. Meanings of various figs. of speech will then be induced. This is in accord with my genl. view that direct raw corpus. QA is th. goal, and Eng → SL is but an el. heuristic step, to be discarded later.

→ Methodologically it would probly be best to work on several of probs. of 789 more or less simultly, to a shallow depth — to take advantage of any simple conceptual synergys — and also to be sure I'm not missing anything at th. outset.

E.g. Eng → SL MT and one or more GPS type probs.

In particular, th. general QATM looks very diflt — tho very simpl., I should continue work on various el. aspects of it — as well keep an eye on a ~~[illegible]~~ less el. form of th. entire machine. Perhaps keep sept. section for QATM, and sub-sections on ... ; on SL's of various types; on various kinds of Q's; kinds of sentences (see report on IR that col. has the ... sentences). ← is this on discourse analysis?

F Ap 10, 64 TMℓ

Phn

1:791.40: Re: Thrm. Provings (e.g.): It may well be that "self-im will not help TM much — that a human does about as well as he can, with his speed and mem. capacity, that there are no better ways of working probs., other than th. "optzd. trial" method using pcosts of symbols as modified by th. code of th. rest of th. corpus, and statistical info on expected time and mem. needed for each tri

If so, then we will have to "explain" many proofs to TM up to a certain pt. in th. trng. seq. — and after that pt., he will ask for aid, less and less freqly. Th. "soonness" of th.'s pt. will depend on th. speed and mem. capacity that th. computer has. With a "hy capacity machine", we can "afford" to let TM do long searches w.o. telling him th. explann. of a gu. proof. — Essentially its a Q of th. rel. "cost" of human v.s. computer time.

Hvr., I can hire people to prepare these "explanns" of proofs. Also, I mite design a special unit so that th. unit could take th. outline of an explann. (like a "hint") and usually find th. "true" explann. from th.s.

Th. point I'm interested in is: Just how low pcost is assoc. with what would be considered "creativity" for a human? [Note that observer O_i considers a proof "creative" if O_i's pcost of th. proof is very small. Th. creator of th. "creation" may find it rather trivial; (i.e. hy pcost).]

Perhaps what I want to ask is: When a gu. human and a M have about th. same set of concepts (≐ abss.) pcosts for them, and it takes th. human ~ 1 day certain proof. — Then what pcost does uch "computing

FAp 10, 64 TM J

Plan — Note, hvr, that there are usually random guesses involved in most searches, so particular timing of a search has much "variance" as a measure of ~~expected~~

:792.40: can take an entirely difrnt. time from any other trial) ?

With a figure for this, we can estimate the relative capacity of a gn. computer config. v.s. a man.

Hvr., with a very hy capacity machine, we mite expect some synergistic effect — i.e. th. machine could work probs in many fields, and have a large memory of heurs. He could do what a very creative man could do in 1 lifetime, then continue! Also th. machine could ~~[illegible]~~ actually do th. adding-on to itself to ↑ its own capacity, at low (dollar) cost. We would give him fairly inexpensive materials to use as "components".

As for application, having a single man who works 1000 times as fast as 1 man, would be good for doing large projects — in which, ordinarily, coördination of several men is a very dift. prob.

.20 O.k. : At th. present time, I have a fairly Good idea (at a "hi level") as to th. general operation of PMTM. Th. 2 modes of most interest at present: ① Thrm. proving ② QA.

① will follow th. work of Sim-New, since they were most interested in heurs to simulate human prob. solving. I will, hvr., carry this on further, for more diflt. probs — and make it possl. for TM to devr. his own heurs sometimes. Also, I will use proby. evals. for optimum search speeds, and will make it easy to give TM exlanns of proofs.

In ② I will start with Eng. → SL MT. Then non-el. QA using raw Eng. ~~[illegible]~~ text and Q's. This will automatically include IR prob. with optimum descriptor invention and assignment.

I should write a more detailed review or outline than 85.40. In particular, give th. reasons for various choices, detailed mechanics, expected diffys, and expected limitations.

F Ap 10, 64 TMγ

Plan

61: 793.40! One of th. impt. things that I have just a vague idea of, is (in Theorem proving e.g.) just how old trials are derbd.; and just how pcosts are assigned, just how "expanns" control pcost assignment, and just how a ~~[illegible]~~ search is made for a new problem. (There are corresp. probs. in QA (learning)).

A possl. way this mite work: One starts with a set of postulates (≡ strings) and a set of permissable xfmns on them, that produce "new strings from old." At any time, there are various "obs" that can look at th. "state of th. system" (th. "state" is th. initial set of strings, plus th. set derived thus far, plus th. goal string (≡ thrm. to be proved), and th. output of th. ob. will determine what to do next. This can be in th. form of deciding which ~~[illegible]~~ new obs. to apply a/o which ops to apply (ops. create new strings [illegible], which xfms th. state of th. system, — obs. also change th. "state of th. system$_2$" — in which "system$_1$" is th. set of strings, but "system$_2$" is system, plus th. knowl. of certain descriptions (i.e. "params") of system.

Th. results of an ob. or set of obs. can control th. pcost of th. "next thing to do." in accord with statistics on th. corpus of successful proofs.

Th. trial obs. and ops. are constructed of other obs. and ops. in a hyer order TM. Th. "trials" of an $\binom{op}{ob}$ is made with resp. to its effectiveness in deriving new proofs a/o th. pcost ~~[illegible]~~ or ~~[illegible]~~ "c cost" (≡ computing in time a/o memory a/o whatever). This "effectiveness in ccost" sounds like my old "Utility"!

[illegible]ld take a look at some of S. and N's

S Ap 11, 64
Plan 795

01: 794.40: Outline of Outline:

I. What PMTM is:

b) Why it was chosen as a close-to-final form for TM.

c) Advantages of cross-couplings betw. modes.

. d) List of some of Th. modes, their properties and uses.

1) ..

2) ..

II. a) How TM_2, TM_3, etc. will be used.

~~b) How TM_2 is used for GPS-type Probs.~~

III How "suggs." are gvn. to TM in early Tng. — Th. meaning of "understanding".

~~IV~~ Th. sugg. channel — as used by a TM that can "understand" SL a/or Eng.

18 IV Early study probs: GPS (thrm. proving) and ⟨Eng → SL MT; SL IR; Eng QA⟩; Why these 2 were selected; Th. expected eln. of both; Some sub-probs in both; Genl. forms of expected Tng. seq.

21

So I PMTM

a) what it is.

b) some modes

c) why chosen as final form of TM (is close to human universality)

d) Impt. of mode cross-couplings.

II TM_1, TM_2, TM_3 . . . .

a) How used where TM_1 is not trivial

— examples e.g. Eng → SL MT.

b.) How used for trivial TM_1 (i.e. GPS type probs.).

III Th. Tng. seq.

a) Use of "open TM at first,

1). for giving suggs., "explans"

A) Nature of explans".

Use of "sugg. channel" in TM

III d) Use of proof (w.o. "explann") as "hint" in advanced TM.

e) Note that

IV Early study probs (see 795.18 – .21)

V Expected performance on these early probs.

a) Early performance with answs. and complete explanns. Slowly ↓ comp[lete]
of explanns, ↓ completeness and presence of answers (a/o "proofs");

VI. General Administration probs.

a) Amount of explann and proof and answ. gn. depends on it rel. machine money cost v.s. human money cost .. — will vary as T.M. matures.

b) Completeness of coding for IR will depend on expected time cost of delay in recoding for final answ.

.18

SN At the present time, I have no clear idea as to how I would go about searching for a good predn. code or a good rep. for a gn. corpus. It is my hope, hvr; that by trying to explicate my intuitive solns. of (e.g. Thm proving and Exp + SC, MT) some probs. that involve induction in an essential way, I will find some rather good, genl. ways to do this..

It will be important that I recognize which parts of my normal intuitive methods of solving these probs. are due to considerations other than purely th. desire for an "optimal soln." in a minimal time".

Perhaps it will be possl. to rent "Third shift"'s of various machines at [re]duced rates — since, during this time there need be no input output — just T.M. trying to "solve" various probs. On each prob. [he] will spend a certain amt. of time, and if he doesn't solve it, in th[at] time, [go] on to th. next. It would be possl. to use several machines this simultly — ideally, they should communicate new-found good abss. to [each o]ther immediately whenever they were found, — but this could wait until th. [next] th. shift — at which time copies of tapes would be exchanged [Differe]nt. machine would work, ideally, on a difnt "type" of prob. [so a]s to minimize importance of this interaction. [Tele]phone lines for intercommunication (that would no[t] [need] sharing) could be used for this

Su Ap 12, 64 TMJ

Plan

1.796.40: machines — usiŋ up <u>all</u> of Th. time not used by Th. other time-sh[aring]
If there <u>is</u> phone communication, and search time for each problem is
>> communication time betw. machines, Then all of Th. machines
could be used in ~~an~~ parallel, random search, for each search
pseudo random nos. could be obtained for each computer in
non-overlapping (Th. other computers) ways.

Re: Thrm proving in Logic, ~~[illegible]~~ Geom, alg, Trig identities, etc. : Looking
at New and Sim. (Unesco 1959 paper) it would seem That even
using Their simple heurs., Th. ~~[illegible]~~ searches would be quite <u>short</u>.
Say I did Geom-proving. Th. intermed. goal would be to put enuf
heurs (with explanns) into Th. machine, so that at a certain pt.
in Th. trng seq. of Thrms., the machine could keep up with
a normal, bright student — in Th. sense of having about Th.
same heurs as Th. students at That pt., so That TM would <u>learn</u>
new heurs. by himself, w.o. being gn. proof ~~etc~~ and/or
explanns — just a trng. seq. of Thrms. to be proved.

From Then on out, we need only continue This trng. seq. as
we would with a human — in fact we could train a
bunch of mathically naive brite humans <u>along with</u> TM, so we could see
when there was much disparity betw. Th. search time for
TM and That for Th. humans. We could ask TM
for <u>several</u> solns. in each search (actually, we would do This anyway, to reduce "local max" troubles), so as to reduce
"randomness error" in computing TM's "search time".
(hvr., TM <u>does</u> give us Th. pcost of his solns. — which reduces
randomness error somewhat — but search time ∝ $\frac{1}{p_{cost}}$
[t]ime may be ∝ $(p_{cost})^{-1}$ for <u>long</u> searches, in which time for each trial is automatically sharply lim[ited]
<u>certain trials take more</u> time Than others.

Remember; That is mainly Th. invention of heurs and (meta)^2 heurs
[...]ted in — <u>not</u> Th. actual skill of TM, or comparison of
[...] humans.

Su Ap 12, 64 TM8

Plan

01: 797.40: Some impt. things I want to get out of this prelim^y work on thrm

1) Some idea of how to solve th. genl. prob. of finding good codes ~~a~~ a corpus (see 796.20–.29) (w/o good regs)

2) Some ways to deal with th. prob. of local maxima. Do this by examining some known/apparent departures from local maxima — i.e. some specific "new ways of looking at things" that have been successful, and how they could have been done rigorously. — it may be that th. best way is to try for better "hyer order" regs, in standard ways.

Try to find some real "new ways of looking at things", and look carefully at th. "old ways of looking at things", to see just what th. situation was — i.e. in what sense a "local max" existed, and just how "th. new way" could have been made "very reasonable", on both a heur. and ~~not~~ a heur level.

Some examples of such "breakthrus" (≡ local max jumps) are

1) Ein. special theo.
2) " . genl. theo.
3) Q.M. – various points.
4) Heliocentric v.s. epicycle theory.
5) Th. "8 fold way".

Th. trouble with such historical examples is that they require a lot of research to find out what th. "conventional ideas of th. time" were.

Perhaps I could find some more recent examples — say ones that I was more intimately involved in.

In particular, it would be nice to find one in Math rather than physics — in that formalization in Math is much easier.

Related to both 1) and 2): ~~Then~~ Consider th. problem B 141 (~~[illegible]~~ "coding using defns.") : Occasionally one will ... redn. in which, th. ~~[illegible]~~ situation to be predicted, completes an npm, ... have enuf cases of that npm to make it worth while ... might look like an "unexpected" bre...

So Ap. 12, 64 TM 8

Plan

0.1:798.40: Is R. folg. a ~~g~~ good example of a "local max"?:

(a) Say I have been watching a seq. of symbols, and coding it first as a Bern. seq., then as a "seq. with D" (ZTB 141). Then, R. "breakthru" occurs if I notice that every 10th symbol is A.

(b) Or it is a seq. of integers ($i = 0/1/9$) — at first I just treat it as a like Markoff seq. (ZTB 141). Then (breakthru) I notice that th. seq. satisfies a certain diference eq. with modulo 10 arithmetic, or so other alg. rule.

Well, in (b), I don't think that such a "breakthru" could be found in any reasonable time unless th. trg. seq. leading to it, made an "diferne eq. mod 10" a reasonable hypoth.

SN] This sugg. a way in which TM mite turn up some entirely new, unnoticed by man, regs. i.e., By spending much more time on a certain (impt) search than th. (equiv.) time that men have spent on it. TM mite thus find a very obscure (search-wise) but very impt. reg. this way.

For many impt. ~~reg~~ regs. (e.g. "laws of nature"), as soon as a fairly good reg. has been found in a certain ~~reg~~ region of phenomenon space, then most workers in th. field stop searching widely and look around only near th. new good reg. — So TM could, perhaps, do much better in many such cases.

→ My impression is that th. so-called "Breakthrus" in ~~Sc~~ Physics not local maxima for th. "Breaker" — tho they were for other ists. Th. "Breaker" simply had a ~~g~~ ~~ver~~ difrnt., very good metho , that th. other men didn't notice or give much wt. to. coding method was "at a hy level" (e.g. code of ; or a meta-meta-meta- .

M Ap 13, 64 TM 8.

Plan

11 799.40 coding methods in ll for a gn. corpus, is enuf to deal adequately w all "local maxima".

If a "local max" seems to exist, I guess th. best thing to do is call in a cleverer man — i.e. look for better codes of th. ~~previous~~ probs. of th. recent past. This simply means spending more search time on them — and, certainly, retaining all reasonably good reps. found.

Th. practice of having many physicists lying around, is equiv. to having many somewhat difrnt codes for th. corpus of — physics data.

This may mean "back tracking" by looking for better codes of (recent) past probs. To avoid this necessity, one should spend an "adequate" amt. of time finding alternate solns. (ē codes) for each problem.

SN) If I get around to doing Geom. proofs, or even in some Alg. proofs, I will want to understand under what conds. TM would make (or accept) "false proofs" (of which there are many examples — e.g. assuming in Geometry certain quantities sum to form a third, — when actually, their diffrnce is th. third).

Also, I may want TM to be able to accept and work math proofs that are not entirely rigorous. (Tho this mite be an "advanced" TM prob.-type, preparitory to Gup + SL)

For thrm. proving: I could just use S and N's work. They done work on simulation of humans prob. solving, so they have used many of th. actual heurs. used by humans. A Q has tng. seq. that they've done long enuf, so that by work I can get TM to th. pt. where he dosn't need to be g... "explanns" that would be needed for a 6...

M Ap 13, 64 TMJ

Plan

.01: 800.40: If the Tmp. seq isn't long enuf, there are lots of the parts of Math that F could teach TM — e.g. Theo. of nos.

— In some parts, I don't know much myself, and so I could "learn along w. TM." Look in Birkhoff and MacLane for some sections of **Math**. "E.G. there are ~ 150 pp on linear spaces!

Also Halmos "fin. dim. vector spaces" and Recent (Feb 64 ff) arts. in "Sp... on Matrices"

Hvr., it would be good, (and easiest), to try to formalize S a... work first to see just what the probs. are. Later, I mite try some other segs. of thrms. in some other part of math.

As I see it in Log. Theorist (and probly in GPS), there are about 3 main heurs: 1) Idea of decreasing the difference betw. what one has and what one wants, by using various ops.

2) Making up sub-goals.

3) "Planning" (≡ Model/making of a simplified proof situation).

What I can do, is look at, say, 10 proofs in which these ideas were used — to assign ~~p~~costs to each of the heurs.

I should do this as quickly as possl. (say using only 1 or 2 of their heurs) to draw up an example of just what sort of "learning" I have in mind. (Dec 61)

) Try to Get N and S's latest work. — The "Science" paper seems about the latest. The "Comp. and Tho't paper" is a bit earlier.

Consider e.g. the single heur for the type/goal of "...fm a into b," Using characters (≡ Chars.) [≡ results of obs.] and operators [redacted] and "differences", ~~...~~

...re, the ops. are ~~...~~ defined by the problem itself.

... obs. are not, and will have to be defined in terms of som... ...t. concepts.

... of an "exhaustive set of obs" (≡ complete) — so that if the "

... rase ... id...

Tu Ap 14, 64

TM 8

Plan

.01 : 801.40 : Idea of a set of obs. that are somewhat orth with resp. to th. So th. obs. are good at localizing error, in th. sense of that differ betw. 2 object in quality q_i, will tend to be modified by application of operator ϕ_i, w.o. modifying other qualities of th. 2 objects. — Here a "quality" is th. result of an ob operating on an object. An ob may be "count th. no. of "+" signs."; or "Are there an even no. of "~" signs to th. rt. of "="?"; etc

---

A "natural" way to invent obs.: **First**, there must be some basic obs "built in", to recognize when 2 objects are identical (at least!). What one does is apply th. various given ops. to various objects, and one trys to characterize, by obs., what each op. will do to a string.

.15 Also, one tries to get a "complete" set of obs. This way 801.38

These genl. ideas re: type 1 probs "may be sufft. impt. to be inserted at birth" — if "type 1 probs." are that impt. ← what are these?? (July 64). Theorem proving is a type 1 prob., but I'm not sure that this is necly a good way to look at thrm proving! —

Best thing to do would be to look at some of their proofs and "protocols". Actually, th. main tricks are: sub-goal devising; differencing (≡ means-ends analy); and planning (≡ models). It should be easy to get some proofs that don't use planning; but there may not be many w.o. sub-goal devising. Hvr., there will be lots of parts of proofs in which ∃ subgoals ∋ only differencing is used to achieve them. So I can get lots of instances of use of this heur.

I want to get various heurs. broken down so that all or parts of are in difrnt. TM_i levels.

Th. devising of good sub-goals, is in an impt. way contingent on ~~th~~ good "ing" — i.e. ~~How~~ To breake a goal down into sub-goals effectively, or when a gn. sub-goal is easier ~~than~~ (or more likely) to a[...] [...]-goals, or than th. original goal. Th. ~~problem~~ difty of [...] be recursively dependent[...]

TMY

Tu Ap 14, 64
Plan

This is wrong: ordinarily intermediate sub-goals  a gn. goal is achievable only by m... – so th. choice is which seq. of s...

01: 803.40: To devise sub-goal thrms., one works back from th. thrm. to be pro... obtaining equiv. thrms., or < pairs of thrms that imply what is to be prove... These can, in turn be "sub-goaled." At each pt., hvr, on... must evalt. th. expected cost of achieving each sub-goal, so... know where to exert one's efforts – e.g. whether a gn. set of sub-g... was "worth while" in th. sense of "easier" than [redacted] th. goal they are a subs. o...

One trouble now, is that I don't have a clear Idea how s... actually do proofs: [redacted] In genl., a "proof" involves xfmns on set of strings to produce a gn. string. → One way to do this i... by "sub-goaling" – e.g. ~~to prove~~ "A" is thrm. to be proved. $B \cup C \supset A$, so B and C become sub-goals. say $D \cup E \supset B$ – So D and E and C ~~imply~~ become sub-goals. Now, if we can prove D and E and C by "direct differencing" we have A proved! This sort of analysis is able to get [redacted] one type of proof. Are there other impt. types? – Perhaps this is the only type achievable w.o. using more hours (like "planning").

A problem, even in this simple proof type is – how "difl't" is a gn. sub-goal? We can answ. this to some extent if that goal is achievable by direct differencing but not [redacted] if other sub-goals have to be devised for it. If a sub-goal is to be obtained by direct differencing, a measure of its difty, is th. amount of "difference" betw. it and th. thing we will xfm into it.

If, hvr, a sub-goal is derived from 1 or more other strings (or other objects), then th.) expected difty is more difl't to estimate ~~...~~

One could just expand out all of these various trees from th. goal to... ...ered, and evaluate ~~...~~ pts. of the branches only when they ca... ...d by "differencing" from known strings.

...too hard – i.e. there are too many trees and br...

...heuristics useful...

~~today~~ W Ap 14, 64 — they make TMS

Plan — I don't think so. / Th. difty function a funct-of Th. differences only even when Th. difference is quite large.

01: 804.40 : Th. moral of Th. last ¶ is: that N and S must use some sort of "difty" evalu. scheme for a sub-goals not achievable by direct difmeng

In one of Their "logic" examples Th. human subject uses rules 10 a/o 11 a/o 12 – which combine 2 strings to yes a third. See whether GPS does Th.3 and how.

Essentially, what I want now, is a fairly complete dern. of GPS's mach in solving ~~logical~~ symb. log. proofs. In particular, I want a dern.

DEF of all heurs used in M.E.A (means-ends-analysis = differencing)

List of N.S. papers that I have around: (I have 17 of Them! — from 1956 to

1) Learning, Generality : Sept 62
2) Probs. of basic orgnn: (SOS 1962) May 62
3) Comp. sim. of hum thinking (Science Dec 61)
4) GPS – a prog. that sims Hum Thot – Munic (61) – in Comp and Thot (p 279)
5) ~~Datam~~ Comp Sim hum thot prob sol. – Datamation Jun-July 61
6) Sim. of hum Thinking Mar 61 (comps. and world of future) (p 95).
~~6) [illegible]~~
7) ~~2)~~ Report on GPS. (Unesco 1959).
9) Sim. of hum Thot. Jun 59 (Rand) GPS, "protocols"
9.5) Intell Learning in a GPS (SOS 1960 p 153) — really May 59 (very ≈ to 8)
10) Chess oct 58 (Comp and Thot p 39)
12) Empirical explrns. with LT mach. (Rand 57) also (Comp Thot 109).
11) Creative Thinking 58 rand (early GPS, "protocols"
8) A Variety of Int. learn in GPS (July 59 RAnd (very — same as (9.5) — How GPS mite be used to derr. a good set of "differences" for "differencing": pp 20–28 derbs. language for debg. differences: pp 32 – tells how GPS mite actually learn useful differences.
13) Pgmg. LT mach.
14) L.T. Mach: (Sept 1956) Info Theo symp. [detailed dern. of LT
15) " " July 56 (RAND)
16) Current dev. in Complex info. procg. May 56.

N, S, Simon. Is This very ≈ to Th. Unesco '59 paper?

mite be useful to get: Report on a GPS Pgm — Jan 59 RAND Paper 1

Simon: Heur. Pgmr. → (Journ ACM Oct 63, pp 493–506)

Sat Ap 18, 64

Plan — Note this carefully —

.01: 805.40 : Another impt. heur trick that N and S ~~are~~ used: They have this observed difference, D_i and they know that Op_i will "reduce" this difference. Hvr., Op_i cannot ordinarily be applied directly to th. object of interest. So we have as a sub-goal — ~~to~~ xfm th. object to a form in which Op_i can be applied. This makes it possl. to have more steps in a "proof", and still have an adequate subgoal structure: i.e. Direct diff gives sub-goals with large jumps., th. sub-goals of xfmg objects so that certain xfmns can be applied to them, gives smaller-jump sub-goals.

It isn't clear to me whether GPS used th. same heurs that LT did. Also, was GPS able to prove all thrms that LT did? Also — did LT's proofs follow th. "protocols" of human subjects?

.20 What I want now: Preferably ~~over~~ all impt. details of a GPS proof that either followed a human protocol closely a/o is intuitively "reasonable" — that used a min. of heurs — hopefully no "planning".

Xfm goals → reduce goals, xfm goals
reduce goals → Apply goals.
Apply Goals → xfm goals, Apply goals.
} p9 doc (8) ("A variety ...").

It is my impression that "Apply" goals are solved very much like "xfm" goals i.e. th. input forms desired are → one can use "differencing" betw. them, and th. present form of th. object to be xfmd. Sometimes there will be > 1 permissable input form — in which case, th. particular way to compare a gn. expressn. to any input form — in which case, th. particul... difference of interest will usually determine which to use. If ambiguity exists, use both (or more) forms.

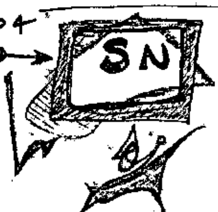
Haven't been too successful in finding .20! Perhaps drop that ap... ...me that their "differencing" idea is pretty good. Then ...hat would have to be specified. Then devis... ...s (N and S ...

Sun Ap 19, 64 TM

Plan

01: 806.40 : in a way **that** is closer to intuitive. Try making a few "protocols" for my <u>own</u> solns. of some of these symb. log. probs.

I <u>could</u> start out with th. idea of "differencing" as a "builtin"

04 → 

Minsky and a grad student have found a way to (partially) map ~~operations on~~ th. continuous functions of a complex var: (+, −, ×, e^{ix}, sin, cos) onto functions of th. integers modulo $n!+1$ (where $n!+1$ is a large prime no.) [i.e. a <u>finite</u> set of nos.] Th. point is that fairly complicated operations become fairly simple. It is possl. to check functional forms <u>quickly</u> using this trick.

There <u>mite</u> be a way to <u>map</u> some <u>other</u> TM operations onto th. integers a/o some other trick using fixed pt. arith and Boolian arith, and any other available fast instructions.

So I can try to devise a TM search scheme that uses comp. instrs. in an <u>optimum way</u>, for <u>very fast</u> searches.

This is similar to th idea of <u>using</u> <u>DNA reproduction</u> for searches; one must first find a good method to do th. <u>mapping</u>.

More recently Rollo Silver did much work on this — perhaps wrote a paper a/o report on it.

Sun Ap 19, 64

TMJ

Plan

01 : 807.40: Consider the problem : " ~~conver~~ xfm $(R \supset \sim P) \cdot (\sim R \supset Q)$ to $\sim(\sim$

Use Newell's 7 xfmn. types (e.g. p 281, Comps. and Thot).

protocol: Clearly we must get rid of the R's.

The only rules that mite help here are R 8, R11, R12

.06 1) Plan : to use R12 we need 2 expressns;
we can get 2, using R8.
Say 2 derivn. of $R \supset \sim P$ and $\sim R \supset Q$

Then xfm them to $F_1 P \supset \binom{\sim R}{R}$ and $\binom{\sim R}{R} \supset F_2 Q$ or
$F_1 Q \supset \binom{\sim R}{R}$ and $\binom{\sim R}{R} \supset F_2 P$

or $F_1(P, Q) \supset F_2(R)$ and $F_2(R) \supset F_3(P, Q)$ etc.

So $R \supset \sim P$; $\sim R \supset Q$

try to get ↑ into the form : $F_1(P) \supset \sim R$

look for rules with "$\supset$" as main connective, that

reverse sides : R2 does it!

$R \supset \sim P \rightarrow P \supset \sim R$

Then via R12 $\rightarrow P \supset Q$ and via R6 ; $\sim P \vee Q$ via R5 $\sim(P \vee \sim Q)$
via R1 $\sim(\sim Q \cdot P)$

---

2) Plan : using R8 : $\sim(\sim Q \cdot P)$

R5 : $Q \vee \sim P$

w.o. using R12 (i.e. just R5 and R11, there is some Q as to whether we can ever drop R. — woops! perhaps via R7!

e.g. $R \cdot (Q \vee \sim P) \rightarrow R$ and $Q \vee \sim P$

$(R \cdot Q) \vee (R \cdot \sim P)$

But I doubt that $(R \supset \sim P) \cdot (\sim R \supset Q)$ implies R or $F_1(R)$

Mon Ap 20, 64 TMY
Plan

01 : 808.40 : Discn : I can't immediately think of ways to use R10 or to elim. R in a useful way — but anyway, see if I can make th. ~~proof~~ "protocol" of 808.06 "better" (i.e. more detailed) We start with a branch on whether to use R11, R8 or R12 to elim. R.

(note that we could possibly use R8 for this like 808.35 - :36) tho I doubt if this works.

Say we chose th. R12 branch, and then lets continue see what heurs we would need to complete th. proof. Then we can use these same heurs on th. R5, R8 branches, to see how much time we spend on them.

So : **First** : How does (R ⊃ ~P)·(~R ⊃ Q) differ from ~(~Q ... th. presence of R : is th. most promenant difference.

we must use R11, R8 or R12 somewhere in th. proof, since all other xfms do not ↓ th. no. of vars.

Say we decide that R12 is at least th. one.

So : Apply R12 is first Goal.

We need 2 strings to apply R12.

Th. only xfm that will ↑ th. no. of strings is R8

So, next necy goal is apply R8.

We can do this directly in 2 ways, so we now have 3 strings, (R ⊃ ~P)·(~R ⊃ Q) ; R ⊃ ~P ; ~R ⊃ Q

Woops! we don't need R8 to ↑ no. of strings! Any will ↑ no. of strings by at least 1. Hvr., there is ... info content, if we have only strings ... having ...

Mon Ap 20, 64 TM8

Plan

.01: 809.40: We could avoid this difty by using th. "Plan", ≡ of elim

i.e. $\begin{pmatrix} F_1(Q) \\ F_1(P) \end{pmatrix} \supset \begin{pmatrix} \sim R \\ R \end{pmatrix} ; \begin{pmatrix} \sim R \\ R \end{pmatrix} \supset \begin{pmatrix} F_2(P) \\ F_2(Q) \end{pmatrix} \rightarrow \begin{pmatrix} F_1(Q) \\ F_1(P) \end{pmatrix} \supset \begin{pmatrix} F_2(P) \\ F_2(Q) \end{pmatrix}$.

This ~~≡~~ idea is close to SandN's concept of "planning" modeling in a space with elements that have ~, · and V elimd. In a modified model space, we could have just "V" (say) el and replaced by "·" ; a/o "~" elimd. Some other tricks: Have ~~A and B stored in such a way~~ A · B stored in such a way that its code is essentially equiv. to that of B · A (similarly with "V"). Then ~~can~~ omit ~~the~~ R1 in th. "model" system

Also A ⊃ B can be notated so that its notation is ≡ ~B ⊃ ~A, and ~~other~~ omit R2. Other rules can be similarly omitted in th. reduced system. Hvr., note that certain of these simplifcns. in th. model system ~~—~~ do not lose info — so they simplify but leave th. system "identical." ⟶ 814.20

SN On parallel Search: My present impression is that ~~≡~~ searches for proofs (and probly / searches for all other things) proceed pretty much like th. construction of s.s in a stochastic PSG — i.e. problistic n-way choices w. known probtys at each pt. There is, hvr., 1 big difrnce, in that in most non-PSG cases, one can end up in a non-terminating loop e.g. — say we had a ~~≡~~ PS Grammar ~~with~~ ~~A→AB~~

$$\Sigma \xrightarrow{.8} a \qquad (a; b \text{ are terminal symbols})$$
$$\xrightarrow{.2} AB$$
$$A \rightarrow AB$$
$$B \rightarrow b$$

... er make th. choice Σ → AB, we end up in th.

... A → AB ...

Mon Ap 20, 64 TM 8 (811?)

Plan

01: 810.40 in any PSG that represents a true PSL, but certainly can occur in a stochastic search.

One (more or less arby. way) to stop such searches: Assume that th. various branches are being searched in ||, by a very large machine — So that every time a 2way branch appears, a new "arith unit" is assigned to th. problem. As soon as one unit finds a soln. (or as soon as n units (n being fixed at, say, 9) find solns.), th. unsuccessful units terminate.

This biases th. search toward short solns. — which is as desired. This will bias th. stoch. probys toward short solns. also.

One trouble is, that techniq. above is not a stoch. but an exhaustive search! As a result, there is no bias toward solns. of hy aprip — (other that shortness time-wise) → see 963. for this — see 963 for riteway to do this

→ To obtain hy aprip solns. — if there is an m-way choice ~~a hollow~~ with $p = p_1, p_2 \cdots p_m$

Then ~~overall choices~~ pick choices at random until th. total probty of all choices chosen is >, say, .3. No! — see

Now, in actuality, we will probly not have a hyly || machine so one can analyse th. prob. so that th. time that a non-|| machine spent on each branch was th. same as if a || machine were doing it. One way to do this: Say one has reason to believe that th. mean time expected for th. completion of th. correct soln. trace will be 1 ms and certainly < 10 ms. Then one ... as tho one were simulating, say, 1000 machines in ||. One then ... each trace for up to 1 ms., then places th. state in m... th. next trace trial. After