



THIS PAGE IS
INTENTIONALLY
LEFT BLANK.

Kazimir Majorinc

O STRUKTURALNIM OSNOVAMA RJEŠAVANJA PROBLEMA U PLANNERU.

Povijest Lispa 42.



Razmjena vještina
Hacklab u mami
7. prosinca 2013.

DESCRIPTION AND THEORETICAL ANALYSIS (USING SCHEMATA) OF

PLANNER:

A LANGUAGE FOR PROVING THEOREMS AND

MANIPULATING MODELS IN A ROBOT

Carl Hewitt

April 1972

Hewittov Planner je zasigurno jedan od najambicioznijih pokušaja proširenja Lispa. **Hewitt** pokušava, zasnovati „rješavanje problema“ na način analogan zasnivanju matematike, onako kao što je teorija skupova osnova matematike. Ni manje ni više.

2. On the Structural Foundations of Problem Solving

Problem solving might seem to be an extremely vague topic, but nevertheless much of the artificial-intelligence research has concentrated on it. In its broadest sense, problem solving encompasses *all* of computer science because any computational task can be regarded as a problem to be solved. For our purposes, however, we desire a somewhat narrower definition, one that excludes routine computational methods such as those used in inverting a 50×50 matrix or in solving a set of linear differential equations.

Nils Nilsson, *Problem solving methods in AI*, 1971

Osnove „rješavanja problema“ moraju uzeti u obzir promjenu situacije u svijetu.

Zasnivanje rješavanja problema mora odgovoriti na pitanje „kakvo znanje o samom sebi i koliko znanja o samom sebi rješavač problema mora imati.“

Osnovni problemi zasnivanja rješavanja problema:

PROCEDURALNA UGRADBA (EMBEDDING): kako se znanje o svijetu može efektivno ugraditi u procedure.

POOPĆENO KOMPAJLIRANJE: koji su dobri načini za transformiranje „goal oriented jezika“ u efikasne algoritme.

PROVJERA: kako provjeriti da procedura radi ono što je potrebno.

PROCEDURALNA APSTRAKCIJA: koje su dobre metode za apstrakciju općeg znanja u odnosu na specijalno znanje.

Dvije krajnosti u specifikaciji procesiranja

EKSPLICITNO PROCESIRANJE - mogućnost da se akcije opišu i kontroliraju do najsitnijih detalja

IMPLICITNO PROCESIRANJE - mogućnost da se opiše željeni rezultat i pri tome ne kaže puno kako bi taj rezultat bio postignut

PLANNER je „goal-oriented formalizam“ u kojem se opisuje što se želi učiniti radije nego kako to učiniti. Ali, može se opisati i kako.

Potreba za dva različita formalizma

Formalizam za opisivanje PROBLEMA

Formalizam za opisivanje METODA koje se smiju koristiti pri rješavanju problema

Dva formalizma su potrebna da bi se spriječilo „varanje“, tj. da se pri opisu problema zadaju i metode za njegovo rješavanje.

Između dviju krajnosti i općenitije od standardnih pristupa u umjetnoj inteligenciji. Oni koji su zagovarali „apstraktnu“ krajnost su, čini se, sa svoje strane kritizirali Hewitta za varanje.

Struktura podataka

Podaci su spremeljeni u „GRAPH MEMORY“ (graf memorije?) koja se sastoji od usmjerenih, označenih grafova. Označeni su i vrhovi i bridovi.

Operacije **PUT** i **GET** su definirane za sve podatke.

Podaci imaju **INDIKATORE**.

INDIKATORI imaju **VRIJEDNOSTI**.

Primjerice, **<+ X Y <* X Z>>** je izraz, posebna vrsta podatka.

SIMPLIFIED je indikator izraza.

CANNONICAL je vrijednost indikatora **SIMPLIFIED**. -

Planner iskorištava dualnost imperativnih i deklarativnih izraza.

Primjer: (implies A B).

Teza o proceduralnom ugrađivanju

Intelektualne strukture treba analizirati kroz proceduralne analogije. Postoje dvije vrste proceduralnih analogija:

OPISI - procedure koje prepoznaju koliko dobro kandidati zadovoljavaju opise

PLANOVİ - procedure koje pokušavaju postići cilj.

Kontrolna struktura

„Pattern directed multiprocess backtrack control structure“.

„Procedura je uzorak onoga što nastoji postići. Procedura ne samo da radi dobro, nego i izgleda kao da radi dobro“

$$(1 \leq i \leq 10) \ \& \ (1 \leq j \leq 10) \ \& \ (i + j = i * j)$$

Strukturalne metode za rješavanje problema

Metode koje se zasnivaju na analizi formalne strukture umjesto semantike

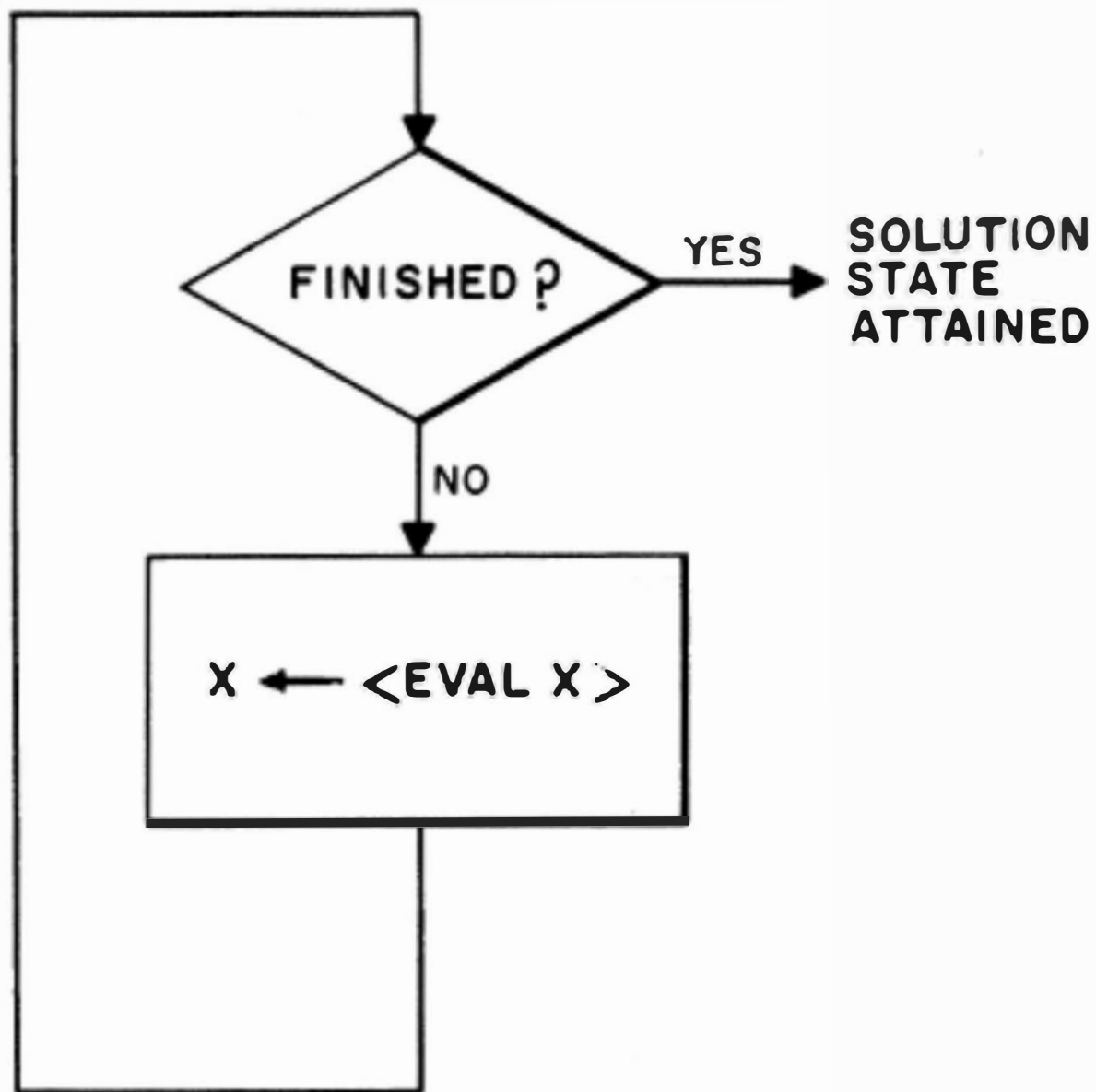
$$(1 \leq i \leq 10) \ \& \ (1 \leq j \leq 100) \ \& \ (i + j = i * j)$$

Čovjek može na temelju ovakvog izraza

- (1) pronaći rješenje matematičkim putem ili
- (2) napisati program koji ga rješava. Svi programi su slični.

I jedno i drugo su strukturalne metode za rješavanje problema. Želi se postići da Planner može rješavati probleme na isti način kao čovjek, te da može otkrivati strukturalne metode i predlagati ih za diskusiju

Progresivno profinjenje



kraj