# PUI

PETR-MAREK.COM

- DOMÉNOVĚ NEZÁVISLÉ PLÁNOVÁNÍ
- VLASTNOSTI, HEURISTIKY A ALGORITMY

- PLANNING PROBLEM
  - STATES $S$
  - INITIAL STATE
  - GOAL STATE
  - ACTIONS THAT TRANSFORMS ONE STATE TO ANOTHER
  - PLAN
    - SEQUENCE OF ACTIONS THAT TRANSFORMS INITIAL STATE TO GOAL STATE
  - POSSIBLE TASKS
    - FIND OUT WHETHER THERE IS A SOLUTION
    - FIND ANY SOLUTION
    - FIND OPTIMAL SOLUTION
    - FIND BEST SOLUTION IN GIVEN TIME
    - FIND SOLUTION SATISFYING SOME PROPERTY
- STATE MODEL FOR CLASSICAL AI PLANNING
  - FINITE SPACE STATE $S$
  - INITIAL STATE $s_0 \in S$
  - SET OF GOAL STATES $S_G \subseteq S$
  - APPLICABLE ACTIONS $A(s) \subseteq A$ FOR $s \in S$
  - TRANSITION FUNCTION $s' = f(a, s)$ FOR $a \in A(s)$
  - COST FUNCTION $c: A^* \rightarrow [0, \infty)$

- TRANSITION SYSTEM
  - $\langle S, I, \{a_1, \ldots, a_m\}, G \rangle$
    - $S$ - FINITE SET OF STATES
    - $I \subseteq S$ IS A FINITE SET OF INITIAL STATES
    - ACTION $a_i \subseteq S \times S$ IS BINARY RELATION ON $S$
    - $G \subseteq S$ IS FINITE SET OF GOAL STATES

- APPLICABLE ACTION
  - ACTION $a$ IS APPLICABLE IN STATE $s$ IF $sas'$ EXIST FOR AT LEAST ONE STATE $s'$

- DETERMINISTIC TRANSITION SYSTEM
  - ONLY ONE INITIAL ~~SYSTEM~~ STATE
  - ALL ACTIONS ARE DETERMINISTIC
  - HENCE ALL FUTURE STATES ARE COMPLETELY PREDICTABLE
  - $\langle S, I, O, G \rangle$
    - $S$ IS FINITE SET OF STATES
    - $I \in S$ IS INITIAL STATE
    - ACTIONS $a \in O$ (WITH $a \subseteq S \times S$) ARE PARTIAL FUNCTIONS
    - $G \subseteq S$ IS A FINITE SET OF GOAL STATES

- SUCCESSOR STATE
  - $s' = APP_a(s)$

- PLAN
  - FOR $\langle S, I, A, G \rangle$ IS SEQUENCE $\pi$
    $\pi = a_1 \dots a_n$
  - SUCH THAT
    $a_1 \dots a_n \in A$ AND $s_0, \dots, s_n$ IS SEQUENCE OF STATES
  - 1. $s_0 = I$
  - 2. $s_i = APP_{a_i}(s_{i-1})$ FOR EVERY $i \in \{1, \dots, n\}$
  - 3. $s_n \in G$
  - THIS CAN BE ALSO EXPRESSED AS
    $APP_{a_n}(APP_{a_{n-1}}(\dots APP_{a_1}(I))) \in G$

- DETERMINISTIC PLANNING TASK
  - 4-TOUPLE $\langle V, I, A, G \rangle$
    - $V$ IS FINITE SET OF STATE VARIABLES
    - $I$ IS INITIAL STATE OVER $V$
    - $A$ IS FINITE SET OF ACTIONS OVER $V$
    - $G$ IS CONSTRAINT OVER $V$ DESCRIBING THE GOAL STATES

- SAS
  - TOUPLE $\langle V, A, I, G \rangle$
    - $V$ IS FINITE SET OF STATE VARIABLES WITH FINITE DOMAINS $dom(v_i)$
    - $I$ IS INITIAL STATE OVER $V$
    - $G$ IS PARTIAL ASSIGNMENT TO $V$
    - $A$ IS FINITE SET OF ACTIONS $a$ SPECIFIED BY $PRE(a)$ AND $EFF(a)$ BOTH BEING PARTIAL ASSIGNMENTS TO $V$

- STRIPS
  - TOUPLE $\langle P, A, I, G \rangle$
    - P IS FINITE SET OF ATOMS (BOOLEAN VARS)
    - $I \subseteq P$ IS INITIAL SITUATION
    - $G \subseteq P$ IS GOAL SITUATION
    - A IS A FINITE SET OF ACTIONS $\alpha$ SPECIFIED BY $PRE(\alpha)$, $ADD(\alpha)$ AND $DEL(\alpha)$, ALL SUBSETS OF P

- PROGRESSION PLANNERS
  - FIND SOLUTION BY FORWARD SEARCH
  - SEARCH SPACES
    - SEARCH NODES CORRESPOND TO STATES
      - WHEN WE ENTER THE SAME STATE ALONG DIFFERENT PATH WE DON'T HAVE TO CONSIDER THE STATE AGAIN
    - SEARCH NODES CORRESPOND TO OPERATOR SEQUENCES
      - BUT DIFFERENT SEQUENCES CAN LEAD TO IDENTICAL STATES

- BACKWARD VS. FORWARD SEARCH
  - IT IS NOT SYMMETRIC
  - FORWARD
    - ONE INITIAL TO SET OF GOALS
    - WHEN WE APPLY OPERATOR $o$ IN STATE, THERE IS SINGLE UNIQUE SUCCESSOR $s'$
  - BACKWARD
    - SEVERAL GOAL STATES TO ONE INITIAL STATE
    - APPLICATION OF OPERATOR $o$ IN $s'$ CAN LEAD TO SEVERAL $s$

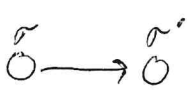
- CLASSIFICATION OF SEARCH ALGORITHMS
  - UNINFORMED
  - INFORMED (HEURISTIC)
  - SYSTEMATIC ALGORITHMS
    - CONSIDER LARGE NUMBER OF SEARCH PATHS SIMULTANEOUSLY
  - LOCAL SEARCH ALGORITHMS
    - WORK WITH FEW CANDIDATE SOLUTIONS AT A TIME
  - USEFULNESS
    - SATISFICING PLANNING
      - HEURISTIC SEARCH OUTPERFORMS
      - AND SOMETHING BETWEEN SYSTEMATIC AND LOCAL SEARCH
    - OPTIMAL
      - UNINFORMED IS BETTER
      - SYSTEMATIC IS REQUIRED
- UNINFORMED, SYSTEMATIC
  - DFS, BFS, ITERATIVE-DFS
- UNINFORMED, LOCAL
  - RANDOM WALK
- HEURISTIC, SYSTEMATIC
  - GREEDY BEST-FIRST
  - A*
  - WEIGHTED A*
  - IDA*

- HEURISTIC LOCAL
  - HILL CLIMBING
  - BEAM SEARCH
  - GENETIC ALGORITHMS
  - SIMULATED ANNEALING

- HEURISTIC SEARCH
  - HEURISTIC FUNCTION
    - $h: \Sigma \to \mathbb{N}_0 \cup \{\infty\}$
    - ESTIMATES DISTANCE FROM $\sigma$ (STATE) TO NEAREST GOAL NODE
    - EFFICIENCY OF ALGORITHM CLOSELY RELATES TO HOW ACCURATELY $h$ REFLECTS THE ACTUAL GOAL DISTANCE
  - OPTIMAL (PERFECT) HEURISTIC
    - $h^*$
    - MAPS EACH SEARCH NODE TO LENGTH OF SHORTEST PATH TO ANY GOAL STATE
  - SAFE
    - $h^*(\sigma) = \infty$ FOR ALL $\sigma \in \Sigma$ WITH $h(\sigma) = \infty$
    - " RETURNS INFINITY FOR NODES WITHOUT CONNECTION TO GOAL STATES"
  - GOAL-AWARE
    - $h(\sigma) = 0$ FOR ALL GOAL NODES $\sigma \in \Sigma$
  - ADMISSIBLE
    - $h(\sigma) \leq h^*(\sigma)$
    - " ALWAYS UNDERESTIMATE"

- CONSISTENT

  - $h(\sigma) \leq h(\sigma') + 1$




  - " HEURISTIC IS DECREASED BY COST OF PATH IF WE TRAVEL CLOSER TO GOAL"

- WEIGHTED $A^*$

  - $f(n) = g(n) + W \cdot h(n)$

  - IF $W = 1$ ... $A^*$

  - IF $W = 0$ ... BREATH FIRST (DIJKSTRA)

  - IF $W \to \infty$ ... GREEDY BEST FIRST

- GENERAL IDEA OF ADMISSIBLE HEURISTIC

  - OBTAINED AS OPTIMAL COST FUNCTION OF RELAXED PROBLEMS

  - FOR EXAMPLE

    - EUCLIDIAN DISTANCE

    - MANHATTAN DISTANCE

    - SPANNING TREE IN TSP

- DOMINATION OF HEURISTICS

  - IF $h_2(\sigma) \geq h_1(\sigma)$ FOR ALL NODES $\sigma$

    - THAN $h_2$ DOMINATES $h_1$

    - $h_2$ IS BETTER FOR SEARCH

- COMBINING ADMISSIBLE HEURISTICS

  - ADMISSIBLE HEURISTICS $h_1, \ldots, h_k$

  - $h(\sigma) = \max_{i=1}^{k} \{h_i(\sigma)\}$

    - IT IS ALSO ADMISSIBLE HEURISTIC AND DOMINATES ALL INDIVIDUAL $h_i$

- HEURISTIC USED IN STRIPS
  - NUMBER OF STATE ~~HEURISTICS~~ VARIABLES THAT DIFFER IN CURRENT STATE AND STRIPS GOAL
  - "MORE TRUE GOAL LITERALS → CLOSER TO GOAL"
  - BUT NOT THAT GREAT
    - NOT INFORMATIVE
      - SMALL RANGE OF VALUES
      - MANY SUCCESSORS HAVE THE SAME VALUE
    - SENSITIVE TO REFORMULATION OF TASKS
    - IGNORES PROBLEM STRUCTURE
      - VALUE DOESN'T DEPEND ON SET OF ACTIONS

- GENERAL PROCEDURES TO COME UP WITH HEURISTICS
  - "SOLVE EASIER VERSION OF PROBLEM"
  - RELAXATION
    - LESS CONSTRAINED PROBLEM
  - ABSTRACTION
    - SMALLER VERSION OF PROBLEMS

- RELAXATION

  - "WE WOULD LIKE TO IGNORE BAD SIDE EFFECTS OF ACTIONS"

    - EXAMPLE

      - 8-PUZZLE

      - IF WE MOVE TILE FROM X TO Y, GOOD EFFECT IS THAT X IS NOW FREE

      - BAD EFFECT IS THAT Y IS NOT FREE, PREVENTING US TO MOVE TILES THROUGH IT

  - STRIPS

    - EFFECTS MAKING ATOMS TRUE

      - GOOD

    - EFFECTS MAKING ATOMS FALS

      - BAD

    - IGNORE ALL DELETE EFFECTS

    - RELAXATION

      $a = \langle PRE(a), ADD(a), DEL(a) \rangle \rightsquigarrow a' = \langle PRE(a), ADD(a), \emptyset \rangle$

- CAN BE SOLVED BY GREEDY ALGORITHM

  - IT IS SOUND

    - IF RETURNS A PLAN, IT IS CORRECT SOLUTION

    - IF RETURNS UNSOLVABLE, THAN PROBLEM IS UNSOLVABLE

  - RUNS IN POLYNOMIAL TIME $O(|P|)$

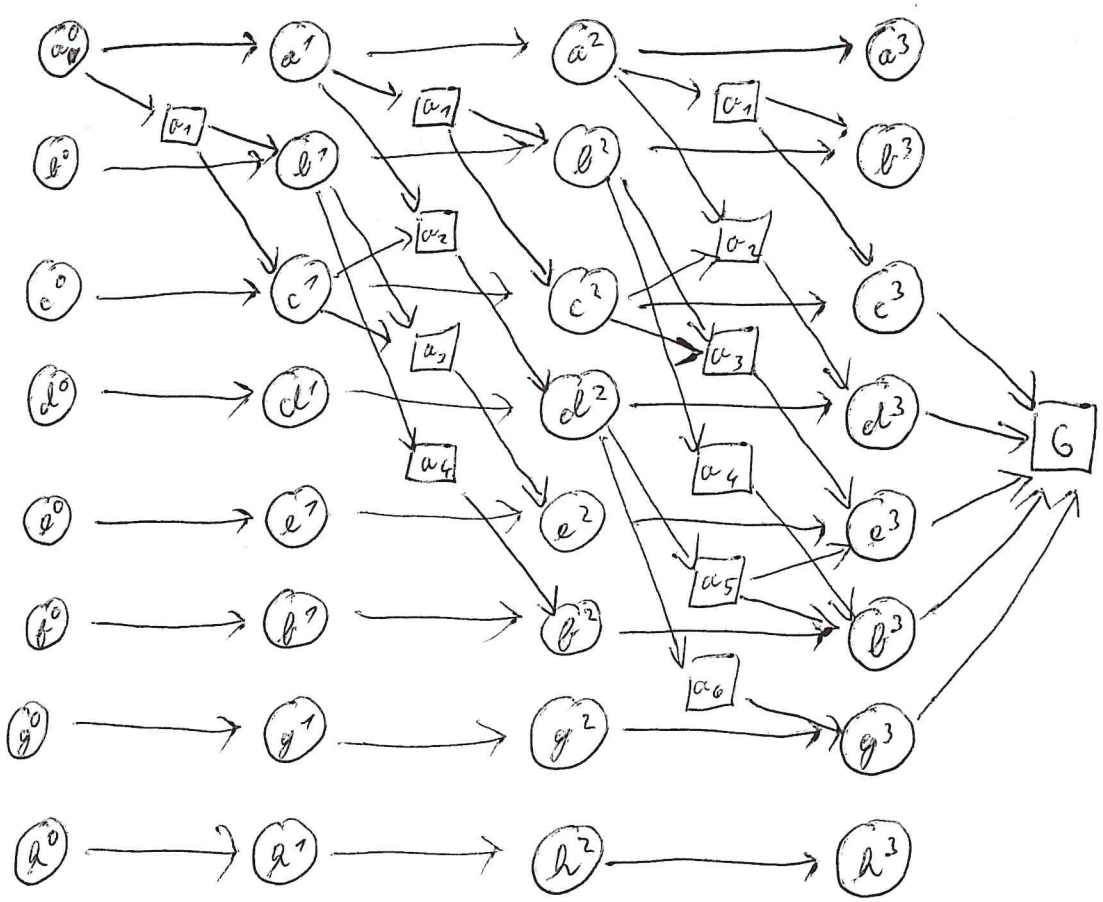
- POSSIBLE USABILITY OF RELAXATION
  - USI OPTIMAL PLANNER FOR RELAXED PROBLEM
    - PP-HARD
    - $h^+$ HEURISTIC
  - COMPUTE ESTIMATION OF ITS DIFFICULTY IN DIFFERENT WAY
    - $h_{MAX}$
    - $h_{ADD}$
  - COMPUTE SOLUTION NOT OPTIMAL, BUT REASONABLE
    - $h_{FF}$

- GRAPHICAL INTERPRETATION

$I = \{a=1,\ b=0,\ c=0,\ d=0,\ e=0,\ f=0,\ g=0,\ h=0\}$

$a_1 = \langle \{a\}, \{b,c\}, \emptyset \rangle$ $a_2 = \langle \{a,c\}, \{d\}, \emptyset \rangle$ $a_3 = \langle \{b,c\}, \{e\}, \emptyset \rangle$ $a_4 = \langle \{e\}, \{f\}, \emptyset \rangle$

$a_5 = \langle \{d\}, \{g\}, \emptyset \rangle$



- SIMPLEST RELAXED PLANNING HEURISTICS ARE FORWARD COST HEURISTICS

- $h_{MAX}$ , $h_{ADD}$

- NODE ANNOTATORS ARE COST VALUES

- COST OF NODE IS ESTIMATE HOW EXPENSIVE IN TERMS OF REQUIRED OPERATORS IS TO MAKE THIS NODE TRUE

- PROPAGATE COST BOTTOM UP

- USE COMBINATION RULE FOR
  - ACTION NODE
  - PROPOSITIONAL NODE

- AT ACTION NODE ADD 1 AFTER COMBINATION RULE

- HEURISTIC VALUE IS COST OF AUXILARY GOAL NODE

- $h_{MAX}$
  - ACTION' NODES

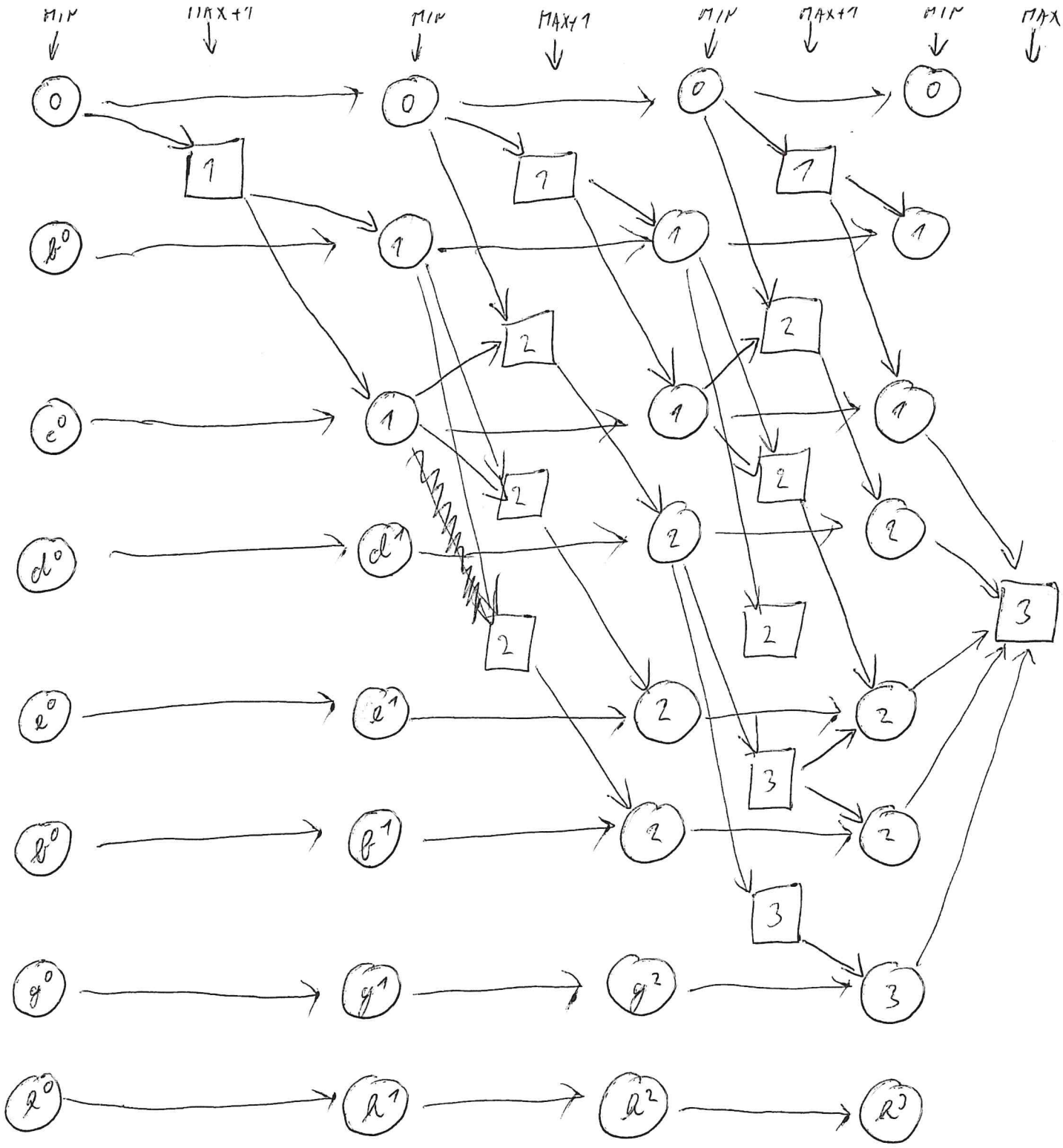
    $COST(n) = MAX\left(\{COST(n_1), \ldots, COST(n_k)\}\right) + 1$

    $MAX(\emptyset) = 0$

  - PROPOSITION NODES

    $COST(n) = MIN\left(\{COST(n_1), \ldots, COST(n_k)\}\right)$

    - IF WE HAVE CHOICE OF WAY HOW TO ACHIEVE PROPOSITION, SELECT THE CHEAPEST ONE


MIN
MAX+1
MIN
MAX+1
MIN
MAX+1
MIN
MAX
0
1
b⁰
c⁰
d⁰
e⁰
f⁰
g⁰
h⁰
d¹
e¹
f¹
g¹
h¹
g²
h²
h³
2
3

- $h_{ADD}$

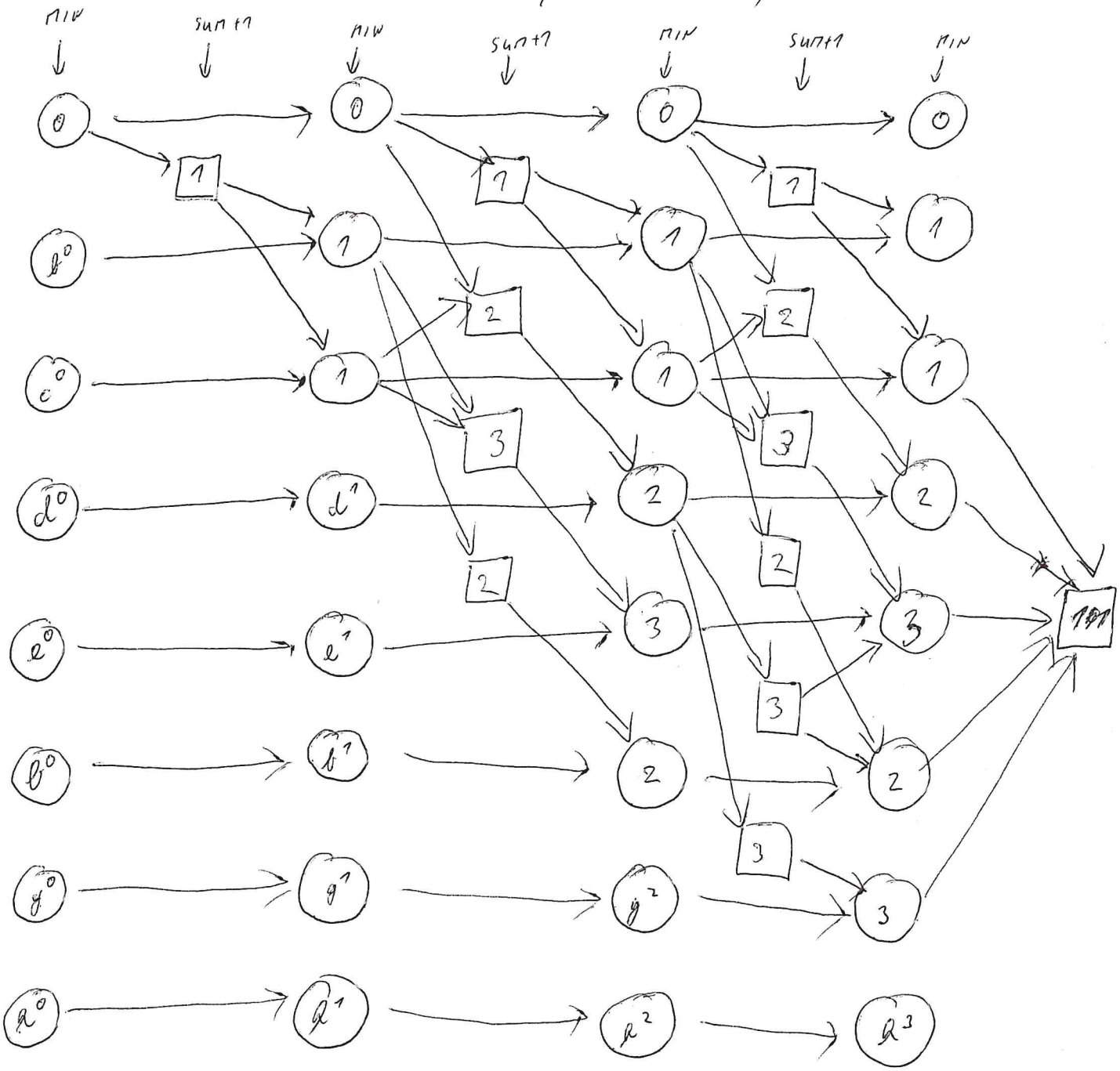
  - ACTION NODES

    $COST(n) = (COST(n_1) + \ldots + COST(n_k)) + 1$

    - IF WE HAVE TO ACHIEVE SEVERAL PRECONDITIONS, ESTIMATE IT BY COST OF ACHIEVING EACH IN ISOLATION

  - PROPOSITIONAL NODES

    $COST(n) = MIN(\{COST(n_1), \ldots, COST(n_k)\})$




- SAFE, GOAL-AWARE
- VERY INFORMATIVE
- NOT ADMISSIBLE AND, THUS NOT FOR OPTIMAL PLANNING

- $h_{FF}$

  - ANNOTATIONS ARE BOOLEAN VARIABLES

  - COMPUTED TOP-DOWN

  - GOAL NODE IS INITIALY MARKED

  - JUSTIFIED ACTION NODE

    - ALL TRUE IMMEDIATE PREDECESSORS ARE MARKED

  - JUSTIFIED PROPOSITION NODE

    - AT LEAST ONE IMMEDIATE PREDECESSOR IS MARKED

  - FOUR RULES TO ANNOTATE

    - MARK ALL IMMEDIATE PREDECESSORS OF MARKED UNJUSTIFIED ACTION NODE

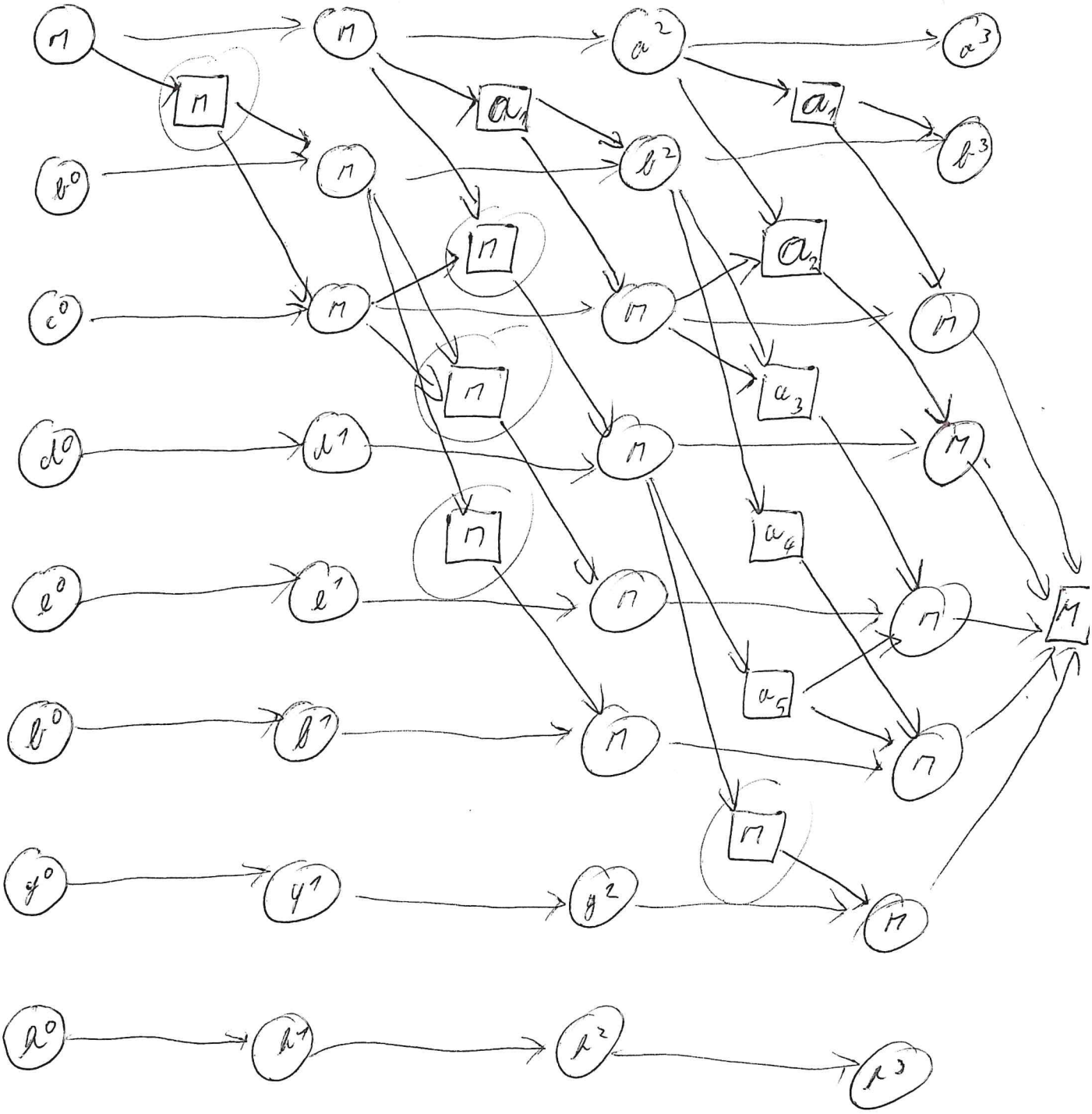
    - MARK IMMEDIATE PREDECESSOR OF MARKED UNJUSTIFIED PROP NODE WITH ONLY ONE IMMEDIATE PREDECESSOR

    - MARK IMMEDIATE PREDECESSOR OF MARKED UNJUSTIFIED PROP NODE CONNECTED VIA IDLE ARC

    - MARK ANY IMMEDIATE PREDECESSOR OF MARKED UNJUSTIFIED PROP NODE

  - EARLIER RULES ARE MORE PREFERRED

- VALUE OF HEURISTIC IS NUMBER OF MARKED ACTION NODES



$h_{FF} = 5$

- SAFE, GOAL AWARE
- NOT ADMISSIBLE, NOT CONSISTENT
- ALWAYS MORE ACCURATE THAN $h_{ADD}$
- CAN BE COMPUTED IN LINEAR TIME

- RELATION BETWEEN RELAXATION HEURISTICS

$$h_{MAX}(s) \leq h^+_{ADD}(s) \leq h^*(s)$$ (optimal on whole problem)

(optimal on relaxed problem)

$$h_{MAX}(s) \leq h^+_{ADD}(s) \leq h_{FF}(s) \leq h_{ADD}(s)$$

- $h^*$ AND $h_{FF}$ ARE PAIRWISE INCOMPARABLE

- $h^*$ AND $h_{ADD}$ ARE INCOMPARABLE

- ABSTRACTION

  - DROPPING SOME DISTINCTIONS BETWEEN STATES

  - BUT TRYING TO PRESERVE THE TRANSITION BEHAVIOR AS MUCH AS POSSIBLE

  - ABSTRACTION OF TRANSITION SYSTEM $T$ IS DEFINED BY ABSTRACTION MAPPING $\alpha$

    - IT DEFINES WHICH STATES SHOULD BE DISTINGUISHED AND WHICH NOT

    - CREATED TRANSITION SYSTEM $T'$ IS SMALLER

    - GOAL DISTANCE IN $T'$ IS USED AS HEURISTIC

  - CREATING $T'$

    - WE WANT ADMISSIBLE HEURISTIC
      - SO $h^*(\alpha(s))$ SHOULD NEVER OVERESTIMATE $h^*(s)$

    - WE HAVE TO ENSURE THAT ALL SOLUTIONS IN $T$ ALSO EXIST IN $T'$

      - IF $s$ IS GOAL STATE IN $T$, THAN $\alpha(s)$ IS GOAL STATE IN $T'$

      - IF $T$ HAS TRANSITION FROM $s$ TO $t$, THEN $T'$ HAS A TRANSITION FROM $\alpha(s)$ TO $\alpha(t)$

- IT ALSO MUST BE EFFICIENT COMPUTABLE
  - $\alpha(s)$
  - $R^*(\alpha(s))$

- WE CAN ALSO USE MAXIMUM OF SEVERAL ADMISSIBLE HEURISTIC, WHICH CREATES NEW ADMISSIBLE HEURISTIC WHICH DOMINATES ALL OLD ONES

- OR WE CAN ADD SEVERAL ADMISSIBLE HEURISTIC
  - BUT IT CAN BECOME NON-ADMISSIBLE

- TRANSITION SYSTEM
  - 5-TOUPLE $\langle S, L, T, I, G \rangle$
    - $S$ IS FINITE SET OF STATES
    - $L$ IS FINITE SET OF LABELS
    - $T \subseteq S \times L \times S$ IS TRANSITION RELATION
    - $I \subseteq S$ IS SET OF INITIAL STATES
    - $G \subseteq S$ IS SET OF GOAL STATES
  - $T$ HAS TRANSITION $\langle s, \ell, s' \rangle$ IF $\langle s, \ell, s' \rangle \in T$

- TRANSITION SYSTEM OF $SAS^+$ PLANNING TASK
  - $\Pi = \langle V, I, O, G \rangle$ IS $SAS^+$ PLANNING TASK
  - TRANSITION SYSTEM OF $\Pi$ IS
    $T(H) = \langle S', L', T', I', G' \rangle$

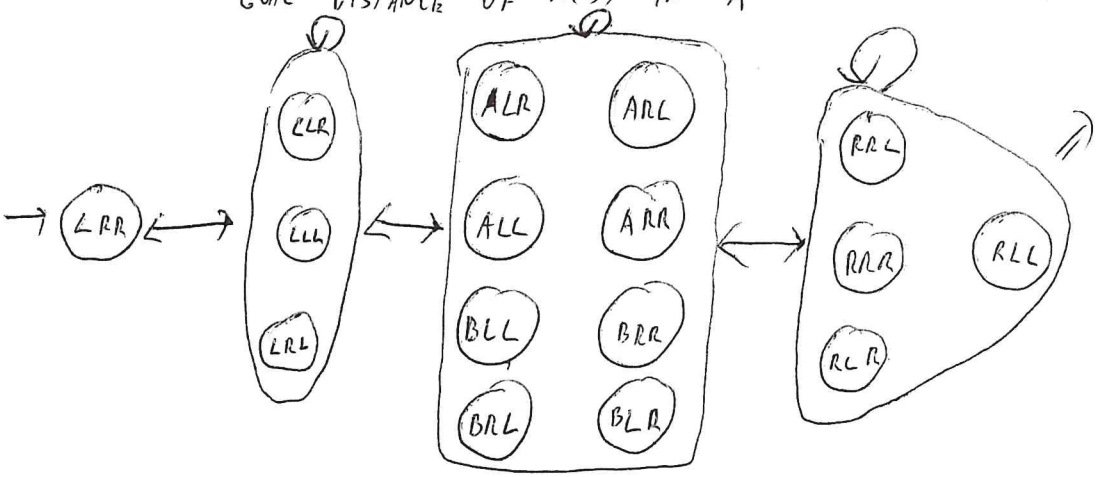
    WHERE
    - $S'$ IS SET OF STATES OVER $V$
    - $L' = O$
    - $T' = \{ \langle s', o', t' \rangle \in S' \times L' \times S' \mid APP_o(s') = t' \}$
    - $I' = \{I\}$
    - $G' = \{ s' \in S' \mid s' \models G \}$

- ABSTRACTION MAPPING
  - $T = \langle S, L, T, I, G \rangle$
  - $T' = \langle S', L', T', I', G' \rangle$
  - $L = L'$
  - $\alpha : S \to S'$
  - $T'$ IS ABSTRACTION OF $T$ WITH ABSTRACTION MAPPING $\alpha$ IF
    - FOR ALL $s \in I$, WE HAVE $\alpha(s) \in I'$
    - FOR ALL $s \in G$, WE HAVE $\alpha(s) \in G'$
    - FOR ALL $\langle s, l, t \rangle \in T$ WE HAVE $\langle \alpha(s), l, \alpha(t) \rangle \in T'$

- ABSTRACTION HEURISTIC INDUCED BY $A$ (ABSTRACTION) AND $\alpha$ (ABSTRACTION MAPPING) $h^{A,\alpha}$
  - IS HEURISTIC FUNCTION

    $h^{A,\alpha} : S \to \mathbb{N}_0 \cup \{\infty\}$

    WHICH MAPS EACH STATE $s \in S$ TO $h^*_A(\alpha(s))$
    - "GOAL DISTANCE OF $\alpha(s)$ IN $A$




$h^{A,\alpha}(LRR) = 3$

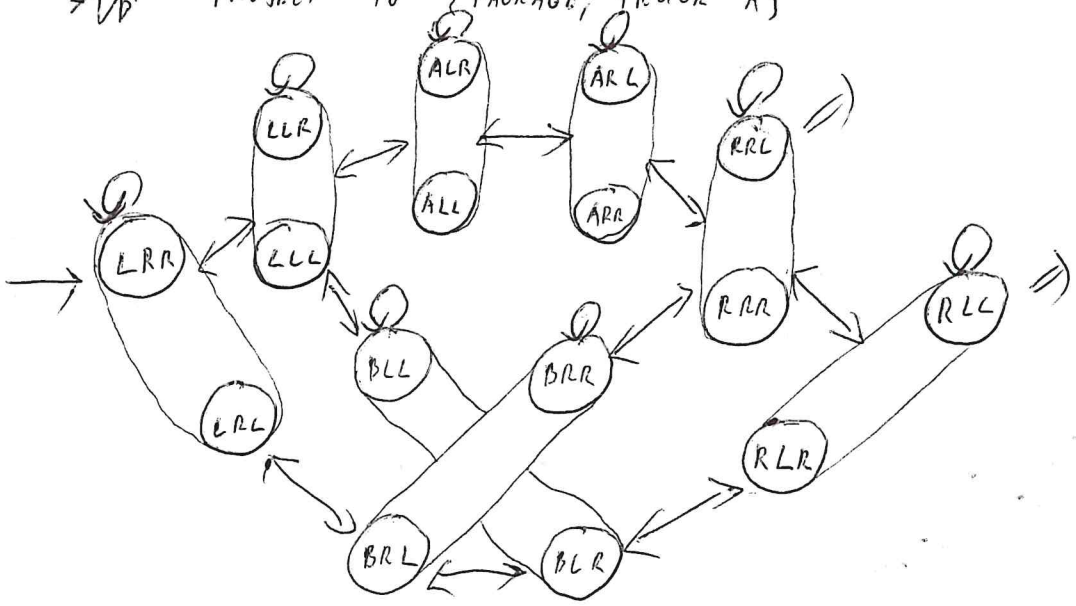
- LET $\Pi$ BE SAS$^+$
  - LET A BE ABSTRACTION OF $T(\Pi)$ WITH ABSTRACTION MAPPING $\alpha$
    - THAN $h^{A,\alpha}$ IS SAFE, GOAL AWARE, ADMISSIBLE AND CONSISTENT

- ORTHOGONAL ABSTRACTION MAPPINGS
  - $\alpha_1, \alpha_2$ BE ABSTRACTION MAPPINGS ON $T$
  - $\alpha_1$ AND $\alpha_2$ ARE ORTHOGONAL IF $\emptyset$:
    - FOR ALL TRANSITIONS $\langle s, \ell, t \rangle$ OF $T$ WE HAVE $\alpha_i(s) = \alpha_i(t)$ FOR AT LEAST ONE $i \in \{1, 2\}$

- ADDITIVITY OF ORTHOGONAL ABSTRACTION MAPPINGS
  - $h^{A_1,\alpha_1} \dots h^{A_n,\alpha_n}$ ARE ABSTRACTION HEURISTICS FOR SAME PLANNING TASK $\Pi$ SUCH THAT $\alpha_i$ AN $\alpha_j$ ARE ORTHOGONAL FOR ALL $i \neq j$
  - THEN $\sum_{i=1}^{n} h^{A_i,\alpha_i}$ IS SAFE, GOAL-AWARE, ADMISSIBLE AND CONSISTENT HEURISTIC FOR $\Pi$

- IN PRACTICE WE WANT INFORMATIVE HEURISTIC WHICH IS SMALL AT THE SAME TIME
  - CONFLICTING GOAL

- PATTERN DATABASE HEURISTICS
  - ABSTRACTION HEURISTICS
  - SOME ASPECTS OF TASK ARE REPRESENTED IN PERFECT PRECISION
  - OTHER ASPECTS ARE NOT REPRESENTED AT ALL
  - EXAMPLE
    - PACKAGE = {L, R, A, B}
    - TRUCK A = {L, R}
    - TRUCK B = {L, R}
    - PROJECT TO {PACKAGE, TRUCK A}




- MERGE AND SHRINK ABSTRACTIONS
  - INSTEAD OF PERFECTLY REFLECTING FEW STATE VARIABLES
  - REFLEX ALL STATE VARIABLES BUT IN POTENTIONALLY LOSSY WAY
  - MERGE
    - A AND A' CAN BE MERGED INTO PRODUCT ABSTRACTION
  - SHRINK
    - DUE TO MEMORY LIMITATIONS
    - WE CAN USE ANY ABSTRACTION OR INTERMEDIATE RESULT AND CONTINUE WITH MERGING PROCESS

- FINITE DOMAIN REPRESENTATION
  - TOUPLE $\langle V, A, I, G \rangle$
    - $V$ - FINITE SET OF STATE VARIABLES WITH FINITE DOMAINS $DOM(V_i)$
    - $I$ - INITIAL STATE IS COMPLETE ASSIGNMENT TO $V$
    - $G$ - GOAL IS PARTIAL ASSIGNMENT TO $V$
    - $A$ - FINITE SET OF ACTIONS
      - $PRE(a)$, $EFF(a)$
        - PARTIAL ASSIGNMENTS TO $V$
      - IN COST SENSITIVE PLANNING
        - EACH ACTION HAS COST $c(a)$

- LANDMARK
  - FORMULA THAT MUST BE TRUE AT SOME POINT IN ANY PLAN
  - THEY CAN BE PARTIALLY ORDERED
  - SOME CAN BE DISCOVERED AUTOMATICALLY
  - CURRENT APPROACHES CONSIDER ONLY LANDMARKS THAT ARE FACTS OR DISJUNCTIONS OF FACTS

- ACTION LANDMARK
  - ACTION WHICH OCCURS IN EVERY PLAN
  - LANDMARKS CAN IMPLY ACTION LANDMARKS AND
  - ACTIONS LANDMARKS CAN IMPLY LANDMARKS
  - SOME ACTION LANDMARKS CAN BE DISCOVERED AUTOMATICALLY TOO

- NATURAL ORDERING

  $A \rightarrow B$

  - A IS TRUE SOME TIME BEFORE B IS TRUE

- NECESSARY ORDERING

  $A \rightarrow_{nec} B$

  - A IS TRUE ONE STEP BEFORE B IS TRUE

- GREEDY - NECESSARY ORDERING

  $A \rightarrow_{gnec} B$

  - A IS TRUE ONE STEP BEFORE B BECOMES TRUE FOR THE FIRST TIME

- $A \rightarrow_{nec} B \Rightarrow A \rightarrow_{gnec} B \Rightarrow A \rightarrow B$

- DECIDING IF GIVEN FACT IS LANDMARK IS PSPACE - COMPLETE

- A IS LANDMARK $\Leftrightarrow$ $\Pi'_A$ IS UNSOLVABLE

  - $\Pi'_A$ IS $\Pi$ WITHOUT OPERATORS THAT ACHIEVE A

- LANDMARK DISCOVERY

  - BY BACKCHAINING

    - ALL GOALS ARE LANDMARKS

    - IF B IS LANDMARK AND ALL ACTIONS THAT ACHIEVE B SHARE A AS PRECONDITION

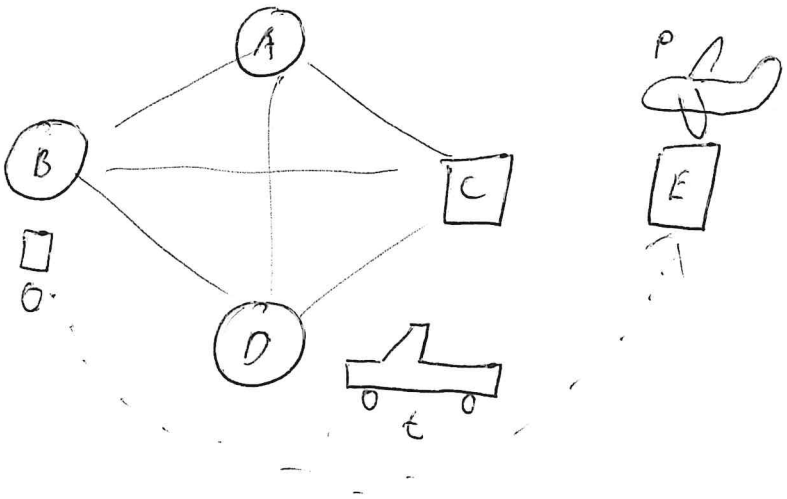
      - A IS LANDMARK

      - $A \rightarrow_{nec} B$

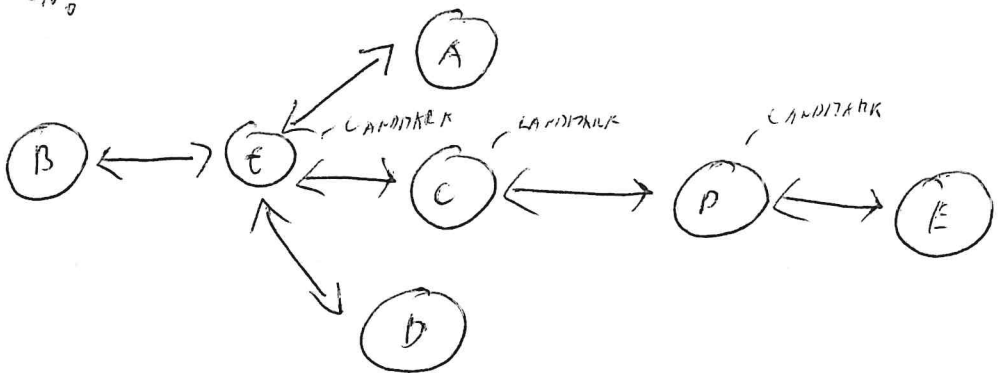
- DISJUNCTION LANDMARKS
  - B IS LANDMARK
    ⇓
  - ALL ACTIONS THAT ACHIEVE B HAVE A OR C AS PRECONDITION THEN
    - $A \vee C$ IS LANDMARK
  - GENERALIZES TO ANY NUMBER OF DISJUNCTS

- DOMAIN TRANSITION GRAPHS
  - DTGs
  - CAN FIND LANDMARKS
  - DTG OF $N \in V$ REPRESENTS HOW THE VALUE OF $N$ CAN CHANGE
  - $DTG_N$ IS DIRECTED GRAPH WITH NODES $D_N$ THAT HAS ARC $\langle d, d' \rangle$ IFF
    - $d \neq d'$
    - AND THERE EXISTS ACTION WITH $N \rightarrow d'$ AS EFFECT AND EITHER
      - $N \mapsto d$ AS PRECONDITION OR
      - NO PRECONDITION ON $N$

- EXAMPLE




- DTG$_{P_0}$




- USING LANDMARKS

  - DISCOVERING LANDMARK HAPPENS IN PRE PROCESSING PHASE

  - LANDMARKS CAN BE USED AS SUBGOALS FOR BASE PLANNER

  - PROS

    - FASTER PLANNING

      - HAVE TO SEARCH TO LESSER DEPTH

  - CONS

    - CAN LEAD TO LONGER PLANS

    - NOT COMPLETE IN PRESENCE OF DEAD-ENDS

  - NUMBER OF LANDMARKS STILL NEEDED TO ACHIEVE CAN BE USED AS HEURISTIC

- LP - BASED HEURISTICS

  - COST PARTITIONING

    - CREATE COPIES $\Pi_1, \ldots, \Pi_m$ OF PLANNING TASK $\Pi$

    - EACH HAS ITS OWN OPERATOR COST FUNCTION

      $cost_i : O \to \mathbb{R}_0^+$

    - FOR ALL $O$: REQUIRE $cost_1(o) + \ldots + cost_m(o) \leq cost(o)$

    - SUM OF SOLUTION COSTS IN COPIES IS ADMISSIBLE HEURISTIC

      $h^*_{\Pi_1} + \ldots + h^*_{\Pi_m} \leq h^*_{\Pi}$

    - WAYS TO DEFINE COST FUNCTIONS

      - UNIFORM: $cost_i(o) = cost(o)/m$

      - ZERO-ONE: FULL OPERATOR COST IN ONE COPY, ZERO IN ALL OTHERS

  - OPTIMAL COST PARTITIONING

    - USE VARIABLES FOR COST OF EACH OPERATOR IN EACH TASK COPY

    - EXPRESS HEURISTIC VALUES WITH LINEAR CONSTRAINTS

    - MAXIMIZE SUM OF HEURISTIC VALUES

  - LP FOR SHORTEST PATH IN STATE SPACE

    MAX GOAL_DIST

    ST $DISTANCE_{S_I} = 0$ - FOR INITIAL STATE $S_I$

    $DISTANCE_{S'} \leq DISTANCE_S + cost(o)$ - FOR ALL TRANSITIONS $S \xrightarrow{o} S'$

    $GOAL\_DIST \leq DISTANCE_{S_*}$ - FOR ALL GOAL STATES $S_*$

- OPTIMAL PARTITIONING FOR ABSTRACTIONS

$$\text{MAX} \sum_{\alpha} \text{GOAL\_DIST}^{\alpha}$$

St

FOR ALL OPERATORS $O$

$$\sum_{\alpha} \text{COST}_{O}^{\alpha} \leq \text{COST}(O)$$

$$\text{COST}_{O}^{\alpha} \geq 0 \qquad \text{FOR ALL ABSTRACTIONS } \alpha$$

FOR ALL ABSTRACTIONS $\alpha$

$\text{DISTANCE}_{S_I}^{\alpha} = 0$ - FOR ABSTRACT INITIAL STATE $S_I$

$\text{DISTANCE}_{S'}^{\alpha} \leq \text{DISTANCE}_{S}^{\alpha} + \text{COST}_{O}^{\alpha}$ - FOR ALL TRANSITION $S \xrightarrow{O} S'$

$\text{GOAL\_DIST}^{\alpha} \leq \text{DISTANCE}_{S_*}^{\alpha}$ - FOR ALL ABSTRACT GOAL STATES $S_*$

- OPERATOR COUNTING

- LINEAR CONSTRAINTS WHOSE VARIABLES DENOTE NUMBER OF OCCURRENCES OF A GIVEN OPERATOR
- MUST BE SATISFIED BY EVERY PLAN THAT SOLVES THE TASK
- EXAMPLES

$Y_{O_1} + Y_{O_2} \geq 1$ - "MUST USE $O_1$ OR $O_2$ AT LEAST ONCE

- LP

$$\text{MIN} \sum_{O} x_O \cdot \text{COST}(O)$$

$$\text{SE} \quad Y_O \geq 0$$

AND SOME OPERATOR-COUNTING CONSTRAINTS

- IT IS ADMISSIBLE HEURISTIC
- SOLVABLE IN POLYNOMIAL TIME
- ADDING CONSTRAINTS MAKE HEURISTIC MORE INFORMED

- STATE EQUATION HEURISTIC

  - SEQ

  - FACTS CAN BE PRODUCED (MADE TRUE) OR CONSUMED (MADE FALSE) BY OPERATOR

  - NUMBER OF PRODUCING AND CONSUMING OPERATORS MUST BALANCE OUT FOR EACH FACT

$$0 = \sum_{o \text{ produces } f} Y_o - \sum_{o \text{ consumes } f} Y_o$$

  - SPECIAL CASE FOR INITIAL AND GOAL STATE

$$G(f) - S(f) = \sum_{o \text{ produces } f} Y_o - \sum_{o \text{ consumes } f} Y_o$$

- POTENTIAL HEURISTICS

  - HEURISTIC DESIGN AS OPTIMIZATION PROBLEM

    - DEFINE SIMPLE NUMERICAL STATE FEATURES $f_1, \ldots, f_n$

    - CONSIDER HEURISTICS THAT ARE LINEAR COMBINATIONS OF FEATURES

$$h(s) = w_1 f_1(s) + \ldots + w_n f_n(s)$$

      - $w_n$ IS POTENTIAL

    - GOAL IS TO FIND POTENTIALS ($w_i$) FOR WHICH $h$ IS ADMISSIBLE AND WELL INFORMED

    - SUCH HEURISTIC IS VERY FAST TO COMPUTE IF ~~HEURISTICS~~ FEATURES ARE

- FEATURES
  - FEATURE IS NUMERICAL FUNCTION DEFINED ON STATES OF TASK

$$f : S \to \mathbb{R}$$

- ATOMIC FEATURE

$$- f_{X=x}(s) = \begin{cases} 1 & \text{IF VARIABLE } X \text{ HAS VALUE } x \text{ IN STATE } s \\ 0 & \text{OTHERWISE} \end{cases}$$

- EXAMPLE $R(s) = 3 f_{X=a} + \frac{1}{2} f_{X=b} - 2 f_{X=c} + \frac{5}{2} f_{X=d}$

- HOW SET WEIGHT
  - REQUIREMENTS
    - WE WANT HEURISTIC WHICH IS
      - ADMISSIBLE
      - CONSISTENT
      - WELL INFORMED
  - LINEAR PROGRAMMING
    - CONSTRAINTS
      - GOAL-AWARENESS ($R(s)=0$ FOR GOAL STATE $s$)

$$\sum_{\text{GOAL FACTS } f} w_f = 0$$

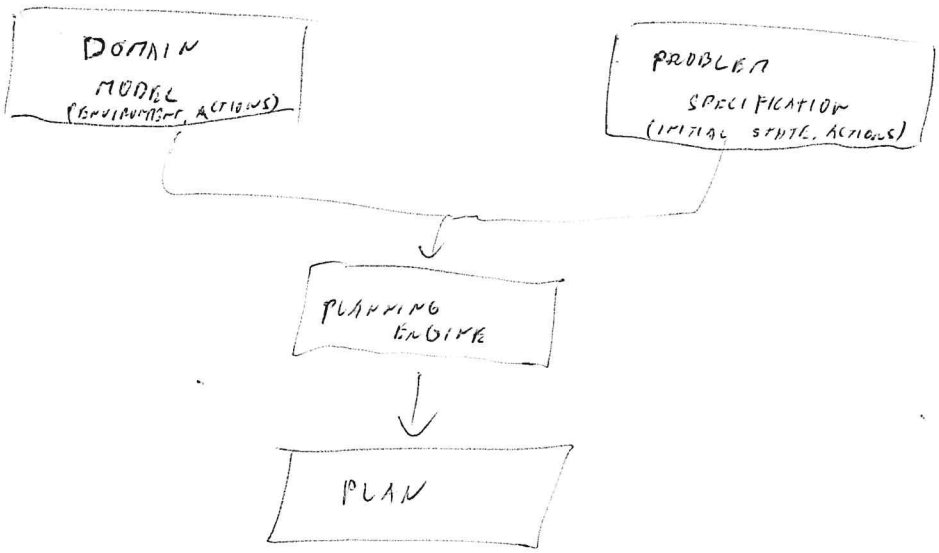
- CONSISTENCY

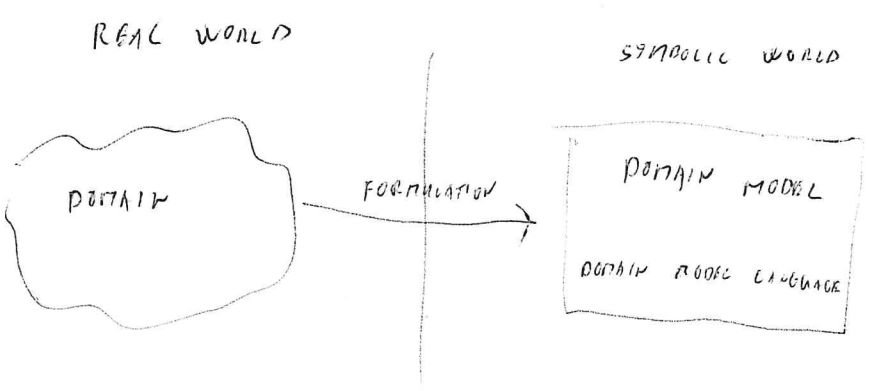
$$\sum_{f \text{ CONSUMED BY } o} w_f - \sum_{f \text{ PRODUCED BY } o} w_f \leq \text{COST}(o) \quad \text{FOR ALL OPERATORS } o$$

- IF GOAL-AWARE AND CONSISTENT $\Rightarrow$ ADMISSIBLE AND CONSISTENT

- WELL INFORMED
  - ENCODE QUALITY METRIC INTO OBJECTIVE FUNCTION AND MAXIMIZE IT
  - EXAMPLES
    - MAXIMIZE HEURISTIC VALUE OF GIVEN STATE (INITIAL FOR EXAMPLE)
    - MAXIMIZE AVERAGE HEURISTIC VALUE OF ALL (OR SOME) STATES
    - MINIMIZE ESTIMATED SEARCH EFFORT

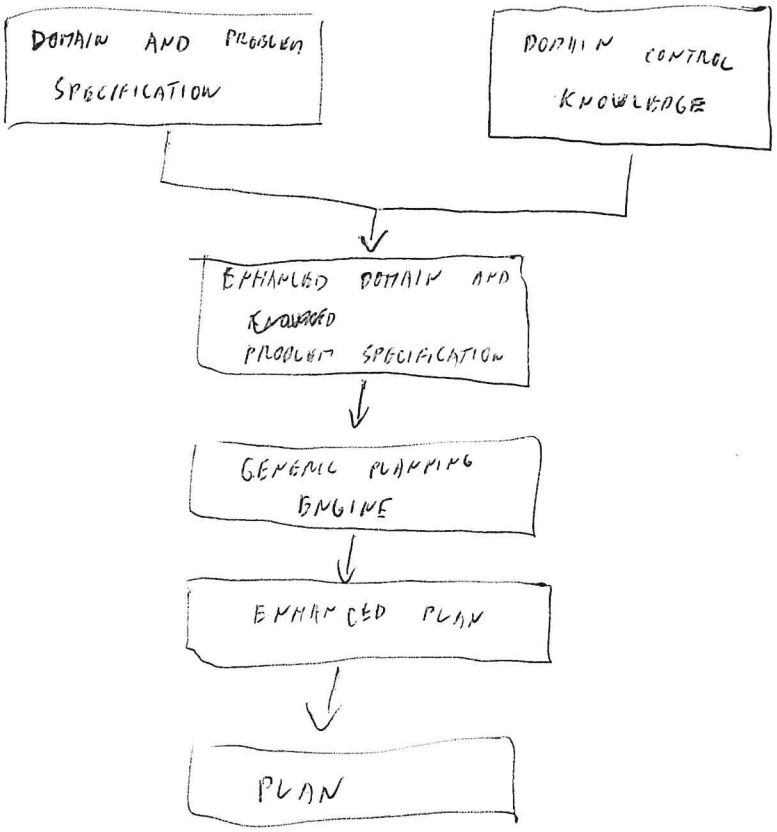
- DOMAIN INDEPENDENT PLANNING CONCEPT



- MODELING LANGUAGES
  - PDDL
    - MOST WIDESPREAD
  - NDDL
  - AMPL
  - RDDL

- DOMAIN CONTROL KNOWLEDGE
  - DCK
  - CAPTURES USEFUL DOMAIN-SPECIFIC INFORMATION
  - PROVIDES GUIDANCE FOR PLANNING ENGINES
  - CAN BE
    - PLANNER SPECIFIC
    - PLANNER INDEPENDENT



- TEMPORAL PLANNING

  - PLANING IN SITUATIONS WHERE ACTIONS

    - HAVE NON-ZERO DURATION

    - MAY OVERLAP IN TIME

  - WE NEED REPRESENTATION OF TIME

  - STATE VARIABLE

    - PARTIALLY SPECIFIED FUNCTION TELLING WHAT IS TRUE AT SOME TIME $t$

  - TEMPORAL ASSERTIONS

    - EVENT

      - $x@t : (v_1, v_2)$

        - AT TIME $t$, $x$ CHANGES FROM $v_1$ TO $v_2 \neq v_1$

    - PERSISTENCE CONDITION

      - $x@[t_1, t_2) : v$

        - $x = v$ THROUGHOUT INTERVAL $[t_1, t_2)$

    - WHERE

      $t_i$ ARE ~~CONSTANTS~~ CONSTANTS OR TEMPORAL VARIABLES

      $v_i$ ARE CONSTANTS OR OBJECT VARIABLES

- PLANNING IN MULTIAGENT SYSTEMS

  - MULTIAGENT SYSTEM

    - CONSISTS OF AGENTS

      - BOUNDED ENTITY

      - INTERACTING WITH OTHERS

    - AGENT

      - REACTIVE AGENT

        - REACT TO IMMEDIATE STIMULI

      - PROACTIVE AGENT

        - PROACTIVELY TRIES TO SATISFY ITS NEEDS AND GOALS

      - PLANNING AGENT

        - PLANS ITS ACTIONS IN ADVANCE

    - HOW SEES OTHER AGENTS

      - ADVERSARY

        - GAME THEORY

      - PART OF ENVIRONMENT

        - POMDP

      - COOPERATIVE

- MULTIAGENT PLANNING PROBLEM

  - $M = \langle A, \{\Pi_i\}_{i=1}^{n} \rangle$

    - $A$ ... SET OF AGENTS

    - $\{\Pi_i\}_{i=1}^{n}$ ... SET OF $n$ AGENT PLANNING PROBLEMS

      - $\Pi_i = \langle V_i, O_i, s_0^i, s_* \rangle$

        - $V_i = V^{PUB} \cup V_i^{PRIV}$ - SET OF PUBLIC AND PRIVATE VARIABLES

        - $O_i = O_i^{PUB} \cup O_i^{PRIV}$ - SET OF PUBLIC AND PRIVATE OPERATORS

- $s_0^i$ - INITIAL STATE OF AGENT $i$ (FULL VALUATION OF $V_i$)

- $s_*$ - PUBLIC GOAL STATE (PARTIAL VALUATION OVER $V^{PUB}$)

- SOLUTION

$$\pi_1 = (o_1, \varepsilon, \varepsilon, \dots, o_2, o_3)$$

$$\vdots$$

$$\pi_n = (\varepsilon, o_4, \varepsilon, \dots, o_5, \varepsilon)$$

- $\varepsilon$ IS NO-OPERATION

- ACTIONS IN EACH TIME-STEP MUST NOT BE MUTUALLY EXCLUSIVE

- GLOBAL PROBLEM

$$\pi^G = \langle V = \tilde{\bigcup_{i=1}} V_i, O = \tilde{\bigcup_{i=1}} O_i, s_0, s_* \rangle$$

- WHY CAN'T WE SOLVE GLOBAL PROBLEM CENTRALLY

- COST OF COMMUNICATION

- COST OF FORMALIZATION

- SPEED IMPROVEMENT

- PRIVACY

- WEAK PRIVACY

- DO NOT SHARE PRIVATE INFORMATION

- BUT IT CAN BE INFERRED FROM PUBLIC INFORMATION AND EXECUTION OF ALGORITHM

- STRONG PRIVACY

- NO PRIVATE INFORMATION CAN BE INFERRED FROM PUBLIC INFORMATION AND RUN OF ALGORITHM

- HARD TO ACHIEVE

- MAP PLANNING PARADIGMS
  - PLAN-BASED COORDINATION
  - STATE-BASED COORDINATION

- PLAN-BASED COORDINATION
  - LET $\pi^D = (o_1^D, \dots, o_k^D)$ BE A PUBLIC PLAN
    - WE SAY THAT $\pi^D$ IS EXTENSIBLE BY AGENT $i$
      - IF BY INSERTING SOME $o_{i_1}, \dots o_{i_j} \in O_i^{PRIV}$ INTO RESPECTIVE $\pi = (o_1, \dots, o_q)$ WE OBTAIN $\pi'$ WHICH IS A SOLUTION TO THE AGENT PROJECTION $\pi^{D,i}$
  - LET $\pi^D$ BE A PUBLIC PLAN. IF $\pi^D$ IS EXTENSIBLE BY ALL AGENTS IN A THEN $\pi^D$ IS A PUBLIC PROJECTION OF SOLUTION $\pi$ OF $\Pi$
  - GENERAL IDEA
    - FIND PUBLIC PLAN EXTENSIBLE BY ALL AGENTS
    - FIND RESPECTIVE EXTENDING PLAN FOR EACH AGENT
  - PLANNING FIRST
    - DISTRIBUTED CSP TO FIND EXTENSIBLE PUBLIC PLAN
      - ONE VARIABLE PER AGENT
      - VALUES
        - ALL POSSIBLE SEQUENCES OF PUBLIC ACTIONS FROM $O^D$ AND PLACEHOLDERS FOR PRIVATE ACTIONS
      - CONSTRAINTS
        - BINARY, SEQUENCES MUST MATCH ON PUBLIC ACTIONS
        - UNARY: THE VALUE (PUBLIC PLAN) MUST BE EXTENSIBLE BY RESPECTIVE

- PSM
  - EACH AGENT $i$
    - GENERATE PLAN FOR $\pi^{D_i}$
    - CREATE PUBLIC PROJECTIONS
    - SHARE AND FIND INTERSECTION
  - IF INTERSECTION NONEMPTY, SOLUTION FOUND
    - ELSE ITERATE

- STATE BASED COORDINATION
  - MULTI AGENT FORWARD SEARCH
    - SEPARATE SEARCH SPACES
    - EACH AGENT SEARCH USING OPERATORS IN $O_i$
    - IF STATE $s$ WAS REACHED BY PUBLIC OPERATOR $o \in O_i^{PUB}$, SEND $s$ TO ALL OTHER AGENTS
      - ENCRIPT PRIVATE PARTS OF STATE
  - MAD-A*
    - HOW ENSURE GLOBAL OPTIMALITY?
      - BETTER STATE MIGHT BE IN OTHER AGENTS' OPEN LIST
      - WE NEED TO CHECK GLOBAL STATE OF DISTRIBUTED SYSTEM

- HEURISTIC IN MULTI AGENT PLANNING

  - PROJECTED HEURISTIC

    - COMPUTED ON THE AGENT PROJECTED PROBLEM IN ESTIMATING ONLY AGENTS PROBLEM SOLUTION COST

    - MAINTAINS ADMISSIBILITY

      - AGENTS PROJECTION IS ABSTRACTION
      - TAKING MAX $(h_i(s), h_j(s))$ IS ADMISSIBLE

    - STRONG PRIVACY

      - NO INFORMATION EXCHANGES
      - UNLESS HEURISTIC ITSELF LEAKS INFO

  - DISTRIBUTED HEURISTIC

    - COMPUTED IN DISTRIBUTED WAY
    - ESTIMATES GLOBAL PROBLEM SOLUTION COST
    - BETTER INFORMED

    ~~- ADDITIVE~~

    - POTENTIAL HEURISTIC

      - $h_{POT}(s) = \sum_{v \in V} pot(\langle v, s[v] \rangle)$

      - IN MAP

$$h^{PUB}_{POT}(s) = \sum_{v \in V^{PUB}} pot(\langle v, s[v] \rangle)$$

$$h^{PRIV_i}_{POT}(s) = \sum_{v \in V^{PRIV}_i} pot(\langle v, s[v] \rangle)$$

$$h^{G}_{POT}(s) = h^{PUB}_{POT}(s) + \sum_{i \in A} h^{PRIV_i}_{POT}(s)$$

- NO ADDITIONAL COMMUNICATION
  - PRIVATE PARTS $R_{POT}^{PRIV_i}(s)$ OF OTHER AGENTS $i \in A \setminus \{j\}$ ARE NOT CHANGED BY AGENT $j$

- PROBABILISTIC PLANNING
  - ACTIONS TAKES SOME TIME
  - ACTIONS HAVE NON-DETERMINISTIC RESULTS
  - ENVIRONMENT MAY CHANGE
  - AGENT DOESN'T KNOW WHOLE SITUATION
  - NOT-PRECISE SENSORS

  - CLASSICAL PLANNING

    $\langle S, s_0, S_G, A, f, c \rangle$

    - $f$ ... TRANSITION FUNCTION $f: S \times A \to S$

  - PROBABILISTIC PLANNING
    - PROBABILISTIC TRANSITION FUNCTION

      $T: S \times A \times S \to [0,1]$

      $\sum_{s' \in S} T(s, a, s') = 1$

- SOLUTION IN CLASSICAL PLANNING IS SET OF ACTIONS
- SOLUTION IN PROBABILISTIC PLANNING IS POLICY

  - $\pi: H \times A \to [0,1]$ ($H$: HISTORY)

    $h = s_1 a_1 s_2 a_2 \dots s_t$

- EVALUATION
  - COST ASSIGNED TO $(s, a, s')$
  - "REWARD"
  - EXECUTING AN POLICY YIELDS A SEQUENCE OF REWARDS
  - POLICY VALUE
    - $w(R_1, R_2, \ldots) = R_1 + \gamma R_2 + \gamma^2 R_3 + \ldots$
    - $w(\pi(s_0)) = E[w(R_1, \ldots)]$
  - THERE EXIST POLICY THAT IS OPTIMAL AT EVERY TIME STEP
  - WE SEARCH FOR
    $\pi^*$ s.t. $w(\pi^*) \geq w(\pi)$ FOR ALL OTHER $\pi$

- MARKOV DECISION PROCESS
  - $\langle S, A, D, T, R \rangle$
    - $S$ - FINITE SET OF STATES
    - $A$ - FINITE SET OF ACTIONS
    - $D$ - HORIZONT, FINITE/INFINITE SET OF TIMESTEPS
    - $T$ - TRANSITION FUNCTION
      $T : S \times A \times S \to [0, 1]$ ; $\sum_{s' \in S} T(s, a, s') = 1$
    - $R$ - REWARD FUNCTION
      $R : S \times A \times S \to \mathbb{R}$
      - TYPICALLY BOUNDED

- MDP POLICY

  - HISTORY DEPENDENT POLICY

    - $\pi : H \times A \rightarrow [0, 1]$

    $$\sum_{a \in A} \pi(h, a) = 1$$

  - BUT FOR SIMPLE CASES WE DON'T NEED HISTORY AND RANDOMIZATION

    - MARKOV ASSUMPTION

  - POLICY IS ASSIGNMENT OF ACTION IN EACH STATE AND TIME

    - $\pi : S \rightarrow A$

  - STATIONARY POLICY

    - POLICY IS SAME EVERY TIME S IS VISITED

  - POSITIONAL POLICY

    - DETERMINISTIC AND STATIONARY POLICY

- FF - REPLAN

  - 1. DETERMINIZE THE DOMAIN (REMOVE STOCHASTICITY)
  - 2. SYNTHETIZE PLAN
  - 3. EXECUTE PLAN
  - 4. IF UNEXPECTED STATE OCCUR, REPLAN

  - WAYS OF DETERMINIZATION
    - KEEP ONLY MOST PROBABLE OUTCOME OF ACTION
    - KEEP ALL OUTCOMES BUT GENERATE SEPARATE ACTION FOR EACH

- SIMPLE, NOT SOUND, NOT OPTIMAL, BUT ENOUGH FOR SIMPLE DOMAINS

- ROBUST -FF
  - GENERALIZES FF-REPLAN
  - 1. DETERMINIZE PROBLEM
  - 2. USE CLASICAL PLANNER TO FIND PARTIAL PLANS
  - 3. AGGREGATE THESE PLANS INTO PARTIAL POLICY
  - 4. CONTINUE UNTIL PROBABILITY OF REPLANNING IS BELOW GIVEN THRESHOLD
  - SEVERAL OPTIONS
    - DETERMINIZATION
    - SELECTING GOAL
    - CALCULATING PROBABILITY OF REACHING TERMINAL STATE
  - IT IS SOUND ONLY WITH SELECTED OPTIONS
  - NOT OPTIMAL IN GENERAL

- FF- HINDSIGHT
  - APROXIMATE VALUE OF STATE
    - SAMPLE SET OF DETERMINIZED PROBLEMS ORIGINATING FROM STATE
    - SOLVE THESE PROBLEMS AND COMBINE THEIR VALUE
  - OPTIMAL VALUE

$$V^*(s,T) = \max_{\pi} E[R(s,F,\pi)]$$

  - $s$ - STATE
  - $T$ - HORIZONT
  - $\pi$ - POLICY
  - $F$ - RANDOM VARIABLE
  - $R$ - REWARD FUNCTION

- HOP VALUE APPROXIMATION

$$V^*(s,T) = E\left[\max_{\pi} R(s,F,\pi)\right]$$

- VALUE OF POLICY

$$V_\pi^\lambda(s) = E\left[\sum_{t=0}^{\lambda} \gamma^t \cdot R(s_t, a_t, s_{t+1}) \mid s_0 = s,\ a_t = \pi(s_t)\right]$$

- OPTIMAL POLICY

$$\pi^{*\lambda}(s) = \operatorname*{ARGMAX}_{\pi} V_\pi^\lambda(s)$$

- FOR LARGE OR INFINITE $\lambda$ WE CAN APPROXIMATE VALUE BY DYNAMIC PROGRAMMING

- $V_\pi^0(s) = 0$

- $V_\pi^\lambda(s) = \sum_{s' \in S} T(s,a,s')\left[R(s,a,s) + \gamma V_\pi^{\lambda-1}(s)\right] \quad a = \pi(s)$

- VALUE ITERATION

- $V^0(s) = 0 \quad \forall s \in S$

- $V^\lambda(s) = \max_{a \in A} \underbrace{\sum_{s' \in S} T(s,a,s')\left[R(s,a,s') + \gamma V^{\lambda-1}(s')\right]}_{\text{Q-FUNCTION } Q(s,a)}$

- FOR $\lambda \to \infty$ VALUES CONVERGE TO OPTIMUM $V^\lambda \to V^*$

- IT CONVERGES

- WE CAN MEASURE RESIDUAL $r$ AND STOP IF IT IS SMALL ENOUGH

$$\left(r \le \varepsilon(1-\gamma)/\gamma\right)$$

- $r = \max_{s \in S} |V_{i+1}(s) - V_i(s)|$

- CONVERGENCE DEPENDS ON $\gamma$

- VALUE ITERATION CALCULATES ONLY VALUES OF STATES
- OPTIMAL POLICY IS EXRACTED BY GREEDY APPROACH
- $\pi^{\lambda}(s) = \text{ARG MAX}_{a \in A} \sum_{s' \in S} T^{\lambda}(s,a,s') \left[ R^{\lambda}(s,a,s') + \gamma V^{\lambda}(s') \right]$

- POLICY ITERATION
  - START WITH ARBITRARY POLICY
  - TWO STEPS
    - POLICY EVALUATION
      - RECALCULATES VALUE OF STATES GIVEN CURRENT POLICY $\pi^{\lambda}$
    - POLICY IMPROVEMENT
      - CALCULATE NEW MAXIMUM EXPECTED UTILITY POLICY $\pi^{\lambda+1}$
  - ITERATE UNTILL VALUE CHANGES

- ASYNCHRONOUS VALUE ITERATION
  - YOU DONT HAVE TO UPDATE ALL STATES AT ONCE
  - YOU CAN UPDATE ONE STATE AT TIME
  - THIS LOWERS THE MEMORY CONSUMPTION

- WE CAN INITIALIZE VALUES OF STATES MORE INTELIGENTLY THAN BY 0
  - WE CAN RUN FF-REPLAN OR THAT FOR EXAMPLE

- MONTE CARLO METHODS
  - REPEATED SAMPLING TO DETERMINE THE PROPERTIES OF SOME PHENOMENON
- MONTE CARLO PLANNING
  - COMPUTE GOOD POLICY FOR MDP BY INTERACTING WITH MDP SIMULATOR

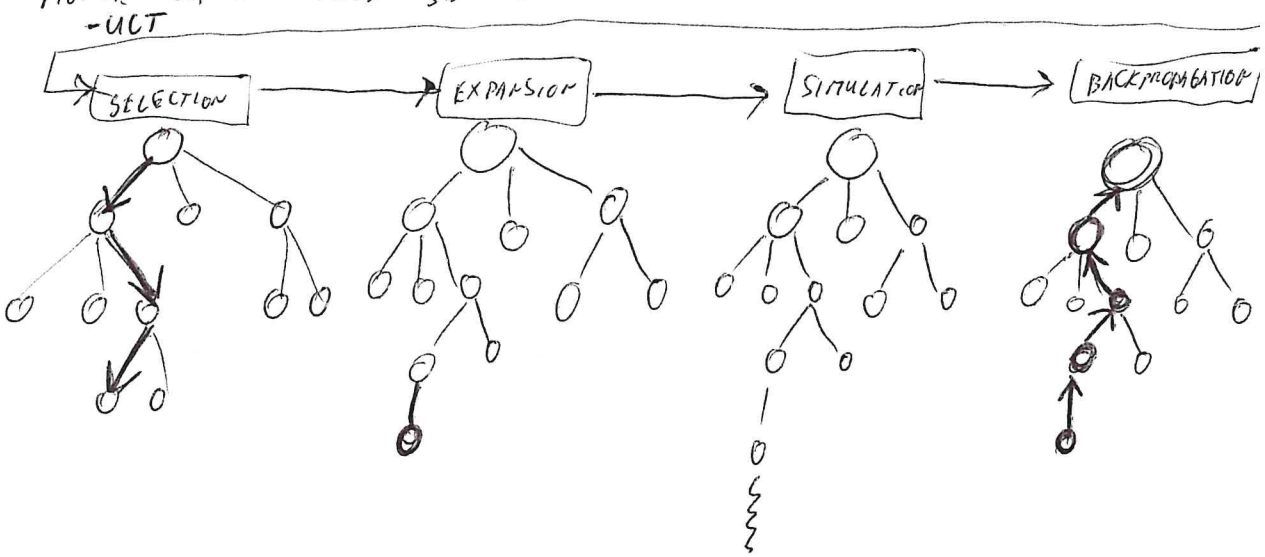
- UCB 1
  - SELECTION ACTION FORMULA
    - SELECT ACTION $a_i$ WHICH MAXIMIZES

$$\bar{x}_i + C\sqrt{\frac{\ln n}{n_i}}$$

AND UPDATE $\bar{x}_i$

- $n$ ... NUMBER OF TIMES THE STATE IS VISITED
- $n_i$ ... NUMBER OF TIMES THE ACTION IS VISITED
- $\bar{x}_i$ ... AVERAGE REWARD FROM PREVIOUS PLAYS
- $C$ ... EXPLORATION FACTOR, ENSURES THAT RARELY EVALUATED ACTIONS ARE EVALUATED

- MONTE CARLO TREE SEARCH
- UCT



- VANILA UCT DOESN'T WORK WELL IN PRACTICE
  - HUGE BRANCHING FACTOR
  - LONG HORIZONT
  - DIFFICULT TO FIND CORRECT PLAY BY RANDOM ROLLOUTS
- PROST
  - SEARCH DEPTH LIMITATION
  - PRUNING OUT UNREASONABLE ACTIONS
  - HEURISTIC VALUE INITIALIZATION

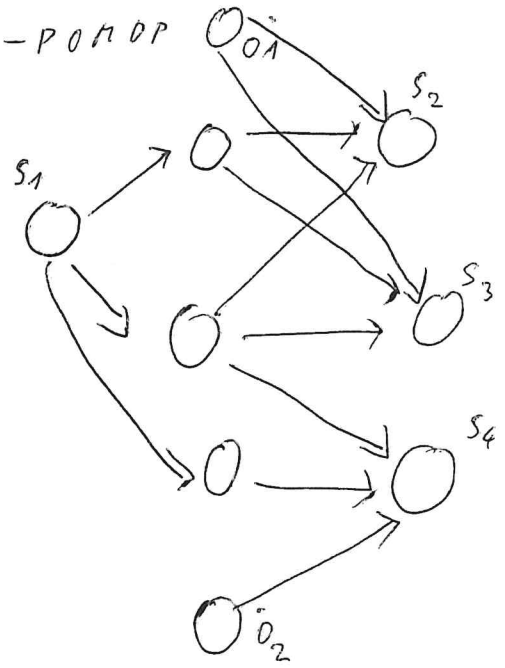
- ONLINE PLANNING
  - SELECTING BEST ACTION IN CURRENT SITUATION IN LIMITED TIME
  - SIMPLE REGRET
    - WE DON'T WANT TO REGRET NOT SELECTING DIFFERENT ACTION IN CURRENT STATE
    - BUT UCB 1 OPTIMIZES CUMULATIVE REGRET
      - SELECTING BEST ARM OVER ALL ATTEMPTS
  - BRUE
    - TWO CONFLICTING TASKS
      - SELECT THE BEST ACTION IN S (REACHING S')
      - EXPLORING AND FINDING THE BEST CONTINUATION AFTER S' IS REACHED
    - TO SATISFY SECOND TASK
      - WE HAVE TO SELECT THE BEST ACTION SUFFICIENTLY OFTEN
      - BUT TO DO THAT WE NEED TO KNOW OPTIMAL CONTINUATION
    - BRUE USES TWO ACTION SELECTION METHODS
      - SELECTION PHASE
        - ACTION IS CHOOSED UNIFORMLY

- IF UPDATE PHASE
  - ACTION IS SELECTED USING GREEDY STRATEGY

- TRIAL-BASED HEURISTIC TREE SEARCH
  - COMMON FRAMEWORK WITH FIVE PARTS
    - HEURISTIC FUNCTION
    - BACKUP FUNCTION
    - ACTION SELECTION
    - OUTCOME SELECTION
    - TRIAL LENGTH
  - MAINTAINS TREE OF ALTERNATING DECISION AND CHANCE NODES
  - SELECTION PHASE
    - ALTERNATING VISIT_DECISION_NODE AND VISIT_CHANCE_NODE
    - SELECTING ACTION BY SELECT_ACTION AND SELECT_OUTCOME
    - TRAVERSING TREE DOWN
  - EXPANSION PHASE
    - UNVISITED NODE IS ENCOUNTERED
    - ADD CHILD FOR EACH ACTION
    - USE HEURISTIC TO INITIALIZE THE ESTIMATE
    - ALLOWS SELECTION PHASE FOR NEW NODES
  - SELECTION AND EXPANSION ALTERNATE UNTIL TRIAL LENGTH
  - BACKUP PHASE
    - BACKUP_DECISION_NODE
    - BACKUP_CHANCE_NODE
    - ALL SELECTED NODES ARE UPDATED IN REVERSE ORDER
    - TRIAL ENDS WHEN BACKUP IS CALLED ON ROOT NODE

- PARTIALLY OBSERVABLE MARKOV DECISION PROCESSES

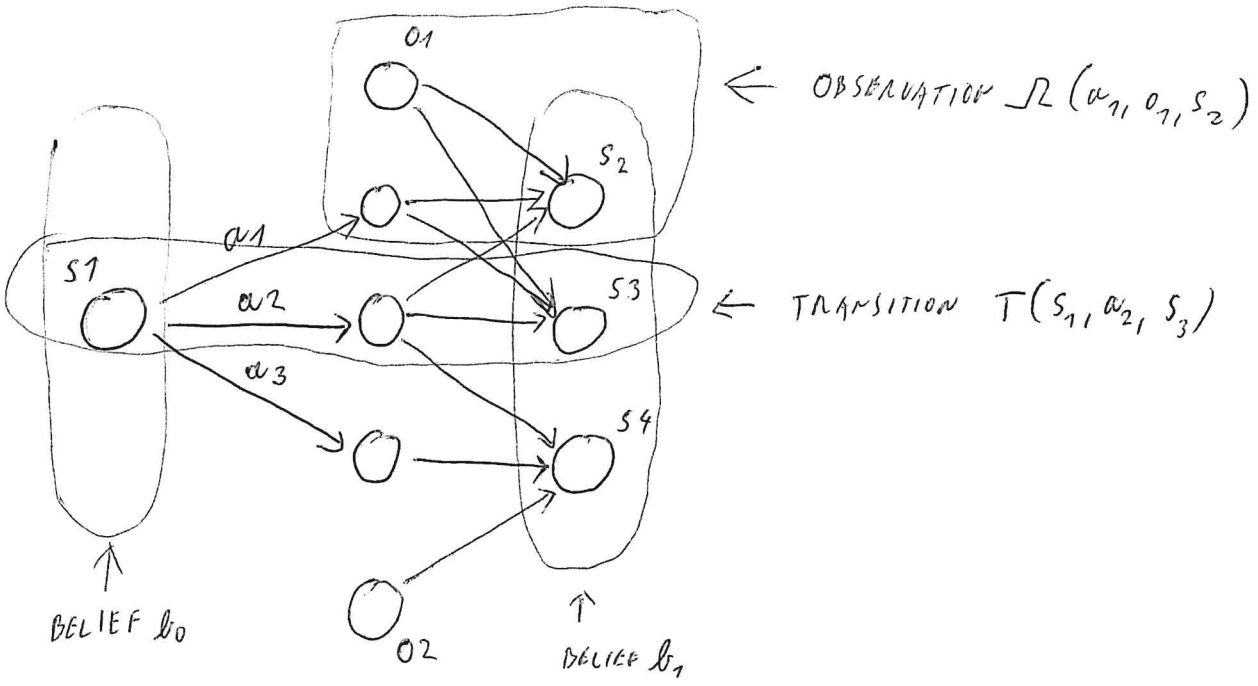
- POMDP




- MODEL FOR SCENARIOS WITH UNCERTAIN OBSERVATIONS

- $\langle S, A, D, O, b_0, T, \Omega, R, \gamma \rangle$

  - $S$ ... STATES (FINITE SET)
  - $A$ ... FINITE SET OF ACTIONS
  - $D$ ... TIME STEPS
  - $O$ ... FINITE SET OF POSSIBLE OBSERVATIONS
  - $b_0$ ... INITIAL BELIEF FUNCTION $b_0 : S \to [0,1]$
  - $T$ ... TRANSITION FUNCTION $T : S \times A \times S \to [0,1]$
  - $\Omega$ ... OBSERVATION PROBABILITY : $\Omega : A \times O \times S \to [0,1]$
  - $R$ ... REWARD FUNCTION $R : S \times A \to \mathbb{R}$
  - $\gamma$ ... DISCOUNT FACTOR $\gamma$ $0 \leq \gamma \leq 1$




- BELIEF IS PROBABILITY DISTRIBUTION OVER STATES
- BELIEFS ARE UNIQUELY IDENTIFIED BY HISTORY
  - $b_1$ - PROBABILITY DISTRIBUTION OVER STATES AFTER PLAYING ONE ACTION
  - $b_t \leftarrow Pr(s_t \mid b_0, a_0, o_1, \dots, o_{t-1}, a_{t-1}, o_t)$
  - BY DYNAMIC PROGRAMMING

$$b_t(s') = \eta\, \Omega(o, s', a) \cdot \sum_{s \in S} T(s, a, s')\, b_{t-1}(s)$$

    - $o$ ... LAST OBSERVATION
    - $a$ ... LAST ACTION
    - $\eta$ ... NORMALIZING CONSTANT

  - BELIEFS DETERMINE NEW VALUES

$$V(b) = \max_{a \in A} \left[ R(b, a) + \gamma \sum_{b' \in B} T(b, a, b')\, V(b') \right]$$

    - THIS WAY WE TRANSFORMED POMDP TO MDP
- IN THEORY WE CAN USE ANY MDP ALGORITHM
  - BUT B IS INFINITE

- SOLVING CONTINUOUS STATE MDPs

  - VALUES CAN BE COMPACTLY REPRESENTED AS FINITE SET OF $\alpha$ VECTORS

    $V = \{\alpha_0, \ldots, \alpha_m\}$

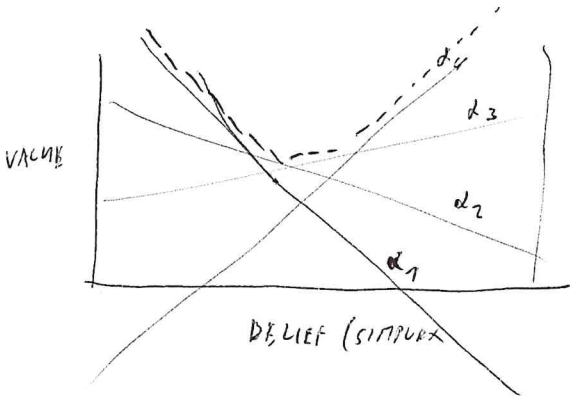
  - $\alpha$ VECTOR IS $|S|$ DIMENSIONAL HYPER-PLANE

    - LINEAR FUNCTION REPRESENTING UTILITY VALUES AFTER SELECTING SOME FIXED ACTION

  - DEFINES VALUE FUNCTION OVER BOUNDED REGION OF BELIEF

  - $V(b) = \max_{\alpha \in V} \sum_{s \in S} \alpha(s) b(s)$

  - $V$ IS PICE-WISE LINEAR CONVEX FUNCTION




  - MODIFICATION OF ITERATION ALGORITHM TO WORK WITH $\alpha$

$$V^t(b) = \max_{a \in A} \left[ \sum_{s \in S} R(s,a) b(s) + \gamma \sum_{o \in O} \max_{\alpha' \in V^{t-1}} \sum_{s \in S} \sum_{s' \in S} T(s,a,s') \Omega(o,s',a) \cdot \alpha'(s') b(s) \right]$$

  - FORMULA TO COMPUTE $\alpha$-VECTORS

$$\alpha^{a,*}(s) = R(s,a)$$

$$\alpha_i^{a,o}(s) = \gamma \sum_{s' \in S} T(s,a,s') \Omega(o,s',a) \alpha_i'(s') \quad \forall \alpha_i' \in V'$$

$$V^a = \alpha^{a,*} \oplus \alpha^{a,o_1} \oplus \alpha^{a,o_2} \oplus \ldots$$

$$V = \bigcup_{a \in A} V^a$$

- BUT THIS HAS SEVERAL DISADVANTAGES

  - COMPLEXITY

    - EXPONENTIAL IN $|V|^{|O|}$

  - SO USEFUL FOR SMALL DOMAINS

- POINT BASED VALUE ITERATION

  - USE ONLY LIMITED SET OF BELIEFS

    $B = \{b_0, \dots, b_q\}$

  - KEEP ONLY SINGLE ALPHA VECTOR FOR ONE BELIEF POINT

  - ANYTIME ALGORITHM ALTERNATES

    - BELIEF POINT VALUE UPDATE

    - BELIEF POINT SET EXPANSION

  - BELIEF VALUE UPDATE

    $$V_b^a = r^{a,*} + \gamma \sum_{o \in O} \arg\max_{\alpha \in \alpha_i^{a,o}} (\alpha \cdot b)$$

    $$V \leftarrow \arg\max_{V_b^a, \forall a \in A} V_b^a \cdot b \quad \forall b \in B$$

  - THIS REMOVES EXPONENTIAL COMPLEXITY

  - VALUE ITERATION ENDS AFTER $k$ ITERATIONS

  - BELIEF POINT SET EXPANSION

    - SAMPLING NEW BELIEFS FROM EXISTING BELIEFS

    - TRYING TO UNIFORMLY COVER REACHABLE BELIEF SPACE