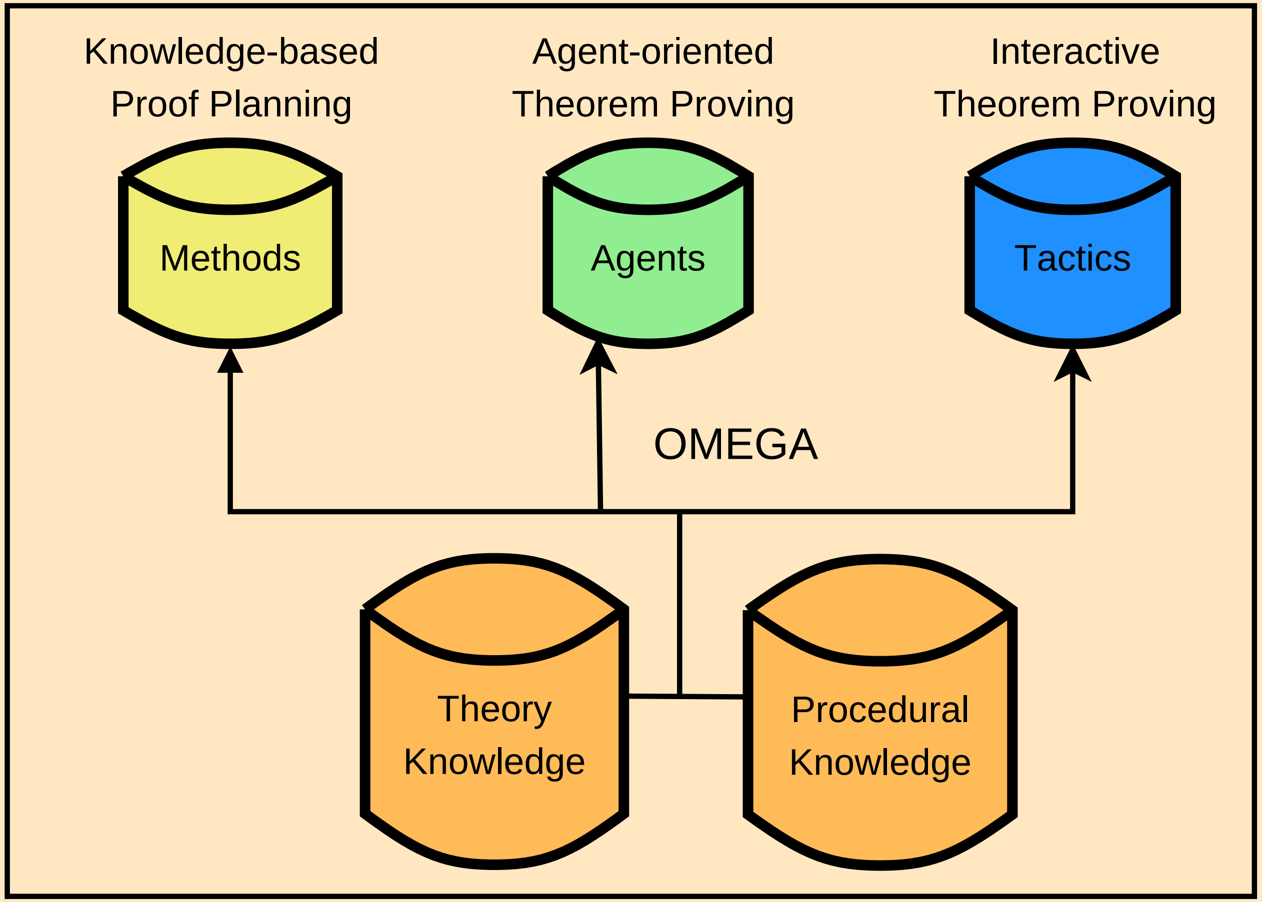
Synthesizing Proof Planning Methods and ΩANTS Agents from Mathematical Knowledge

Serge Autexier, Dominik Dietrich

dietrich@ags.uni-sb.de

Saarland University, Saarbrücken

Motivation



- Problem 1: different representations for tactics and methods
- Problem 2: manual encoding of theory knowledge and procedural knowledge

ΩMEGA-Tactics

- Example: Substitution $\dfrac{P_1 : F[U] \quad P_2 : U = V}{C : F[V]}$
  - Having P_1 and P_2, we call tactic subst-m-fw
  - Having C and P_2, we call tactic subst-m-bw1
  - Having C and P_1, we call tactic subst-m-bw2
  - Having P_1, P_2 and C, we call tactic subst-m-close

$$\dfrac{P_1 : F \qquad P_2 : U = V}{C : G}\ \textit{subst-m-bw1}(\pi)$$

Appl. Cond.: $(G|_\pi = U)$

Outline: $\langle P_1,$ `compute-subst-m`$(P_2, C, \pi)\rangle$

$\langle \pi,$ `compute-pos` $(P_2, C)\rangle$

$$\dfrac{P_1 : F \qquad P_2 : U = V}{c : G}\ \textit{subst-m-fw}(\pi)$$

Appl. Cond.: $(F|_\pi = U)$

Outline: $\langle c,$ `compute-subst-m`$(P_1, P_2, \pi)\rangle$

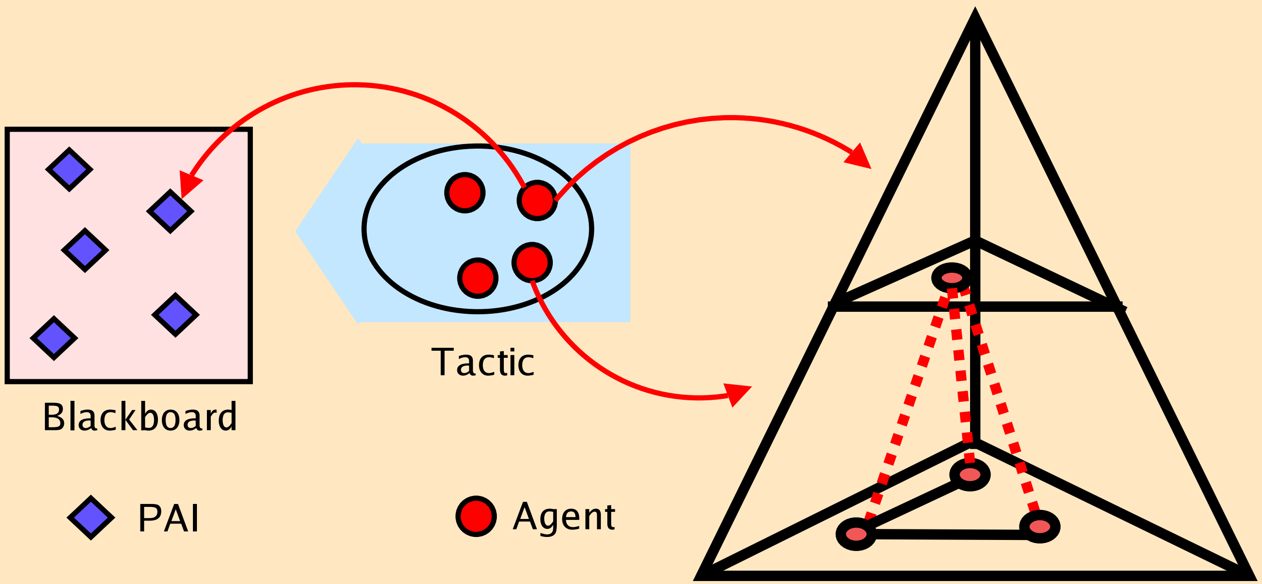
$\langle \pi,$ `compute-pos` $(P_1, P_2)\rangle$

ΩMEGA-Methods

method	subst-m-bw1
premises	$\oplus L1, L2$
conclusions	$\ominus L3$
appl. cond	$F\|_{\pi} = U$
proof schema	$L1 : \Delta \quad \vdash F$ (open) $L2 : \quad \vdash U = V$ $L3 : \Delta \quad \vdash G$ (open)

Argument Agents in ΩANTS...

- Society of agents for each tactic
- Their job: finding instantiations for the tactic (PAI)
- Single agent: try to add a subset of the arguments to a PAI



- Example: agent$_1$ triest to find an instantiation of P_1, agent$_2$ tries to add instantiation for P_2, agent$_3$ tries to add instantiation of C

I. Inferences

Example of Inferences (1)

$$\frac{P_1 : F \quad P_2 : U = V}{C : G} \; \textit{subst-m}(\pi)$$

Appl. Cond.: $(F|_\pi = U \wedge G|_{\pi \leftarrow V} = F) \vee (G|_\pi = U \wedge F|_{\pi \leftarrow V} = G)$

Outline:

$\langle C, \texttt{compute-subst-m}(P_1, P_2, \pi)\rangle$

$\langle P_1, \texttt{compute-subst-m}(P_2, C, \pi)\rangle$

$\langle \pi, \texttt{compute-pos}\ (P_1, P_2)\rangle$

$\langle \pi, \texttt{compute-pos}\ (P_2, C)\rangle$

Examples of Inferences (2)

$$[\epsilon > 0, D > 0, 0 <\mid x - A \mid, \mid x - A \mid < D]$$

$$\vdots$$

$$\frac{P :\mid F(x) - L \mid < \epsilon}{C : \lim_A F = L} \; Limit(\epsilon, x)$$

Appl. Cond.: $\mathsf{EV}(\epsilon, \{F, A, L\}) \wedge \mathsf{EV}(x, \{F, A, L, D\})$

Outline: -

General Form for Inferences

$$\frac{\begin{array}{ccc}[H_1^1,\ldots,H_{n_1}^1] & & [H_1^k,\ldots,H_{n_k}^k] \\ \vdots & & \vdots \\ P_1 : F_1 & \ldots & P_k : F_k\end{array}}{C_1 : G_1 \quad \ldots \quad C_m : G_m}\ \mathit{Name}(\omega_1,\ldots,\omega_n)$$

Outline Functions: $\langle i^1, f^1(i_1^1,\ldots,i_{j_1}^1)\rangle \ldots \langle i^l, f^l(i_1^l,\ldots,i_{j_l}^l)\rangle$

Application Conditions: $P(i_1^0,\ldots,i_m^0)$

parameter variables: $\mathcal{P}$

names for premises and conclusions: $\mathcal{N}$

Applicability of Inferences (1)

- Goal: maintain state of the instantiation explicitly (PAI) with respect to a given proof situation T
- Idea: start with an empty PAI, instantiate arguments by updating the PAI
- PAI applicable if outline functions can provide missing values
- Formally PAI: $PAI_I^T := \langle \sigma_{\mathcal{P}}, \sigma_{\mathcal{N}}, \sigma_{\mathcal{X}} \rangle$
  - empty: $dom(PAI_{\mathcal{I}}^{T}) = \emptyset$
  - initial: $\sigma_{\mathcal{N}}(a_j) \notin$ *wff*$_{\Sigma,\mathcal{V}}^{\mathcal{X}}$ and for all $p \in \mathcal{P}$ holds $\sigma_{\mathcal{P}}(p) = \perp$
  - fully specified: $P \cup C \subseteq dom(\sigma)$
  - complete: sequence of outline functions $\langle v_l, f(I_{k_l}^{\vec{k}}) \rangle$, $0 \leq k \leq m$ such that $PAI_I^T \oplus \langle v_1, f(i_{k_1}^{\vec{1}}) \rangle \oplus \ldots \oplus \langle v_m, f(i_{k_m}^{\vec{m}}) \rangle$ is a fully specified

Applicability of Inferences (2)

Example:

- Given a proof situation $T := 2 * 3 = 6 \vdash 2 * 3 < 7$
- First step: instantiate arguments (2 updates):
  - gives initial PAI: $PAI = \{P_2 \mapsto_T 2 * 3 = 6, C \mapsto_T 2 * 3 < 7\}$
- Steps: use outline functions to calculate missing values

$\langle C, \texttt{compute-subst-m}(P_1, P_2, \pi)\rangle$

$\langle P_1, \texttt{compute-subst-m}(P_2, C, \pi)\rangle$

$\langle \pi, \texttt{compute-pos}\ (P_1, P_2)\rangle$

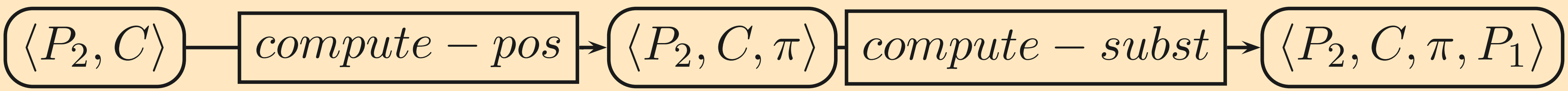
$\langle \pi, \texttt{compute-pos}\ (P_2, C)\rangle$

  - Update 1: compute π using P_2, C
  - Update 2: compute P_1 using P_2, C, π
  - gives: $PAI = \{P_2 \mapsto_T 2 * 3 = 6, C \mapsto_T 2 * 3 < 7, \pi \mapsto 6, P_1 \mapsto 6 < 7\}$

Determing the Application Directions

- Consider equivalence classes of PAIs, called PAI-status
  - function $\mathcal{S}_{\mathcal{I}} : \mathcal{A} \rightarrow \{TERM, POS, \perp\}$. Its domain is $\mathrm{dom}(\mathcal{S}_{\mathcal{I}}) := \{a \mid \mathcal{S}_{\mathcal{I}}(a) \neq \perp\}$.

  and PAI-status updates
- Application directions = complete PAI-statuses
- Is $\langle P_2, C\rangle$ complete?



II. Generating Inferences

Generating the Inferences

- Inspired by technique presented by Benjamin Wack in his PhD
- The idea: apply ND rules to given formula C, 2 phases
  - First: apply ND elimination rules
  - Second: simplify premises $[\mathcal{H}]P$ in Γ
- Yielding a set of inferences $\mathcal{R}$
- Collect eigenvariable conditions in $\mathcal{C}$
- Handle equality via Leibniz definition

Generating the Inferences

$$\frac{R \star \mathcal{C}; \Gamma \vdash_R A \Leftrightarrow B}{R \star \mathcal{C}; \Gamma \vdash_R A \Rightarrow B \star \mathcal{C}; \Gamma \vdash_R B \Rightarrow A} \Leftrightarrow_E$$

$$\frac{R \star \mathcal{C}; \Gamma \vdash_R s = t}{R \star \mathcal{C}; \Gamma \vdash_R X(s) \Rightarrow X(t) \star \mathcal{C}; \Gamma \vdash_R X(t) \Rightarrow X(s)} =_E$$

where X new wrt. $\mathcal{C}; \Gamma \vdash_R s = t$

$$\frac{R \star \mathcal{C}; \Gamma \vdash_R A \Rightarrow B}{R \star \mathcal{C}; \Gamma, []A \vdash_R B} \Rightarrow_E$$

$$\frac{R \star \mathcal{C}; \Gamma \vdash_R A \wedge B}{R \star \mathcal{C}; \Gamma \vdash_R A \star \mathcal{C}; \Gamma \vdash_R B} \wedge_E$$

$$\frac{R \star \mathcal{C}; \Gamma \vdash_R \forall x A}{R \star \mathcal{C}; \Gamma \vdash_R A[X/x]} \forall_E$$

where X new wrt. $\mathcal{C}; \Gamma \vdash_R \forall x A$

$$\frac{R \star \mathcal{C}; \Gamma \vdash_R \exists x A}{R \star \mathcal{C}, c \notin \Gamma, \exists x A; \Gamma \vdash_R A[c/x]} \exists_E$$

$$\frac{R \star \mathcal{C}; \Gamma, [\Delta]A \Rightarrow B \vdash_R F}{R \star \mathcal{C}; \Gamma, [\Delta, A]B \vdash_R F} \Rightarrow_I$$

$$\frac{R \star \mathcal{C}; \Gamma, [\Delta]A \wedge B \vdash_R F}{R \star \mathcal{C}; \Gamma, [\Delta]A, [\Delta]B \vdash_R F} \wedge_I$$

$$\frac{R \star \mathcal{C}; \Gamma, [\Delta]A \vee B \vdash_R F}{R \star \mathcal{C}; \Gamma, [\Delta]A \vdash_R F \star \mathcal{C}; \Gamma, [\Delta]B \vdash_R F} \vee_I$$

$$\frac{R \star \mathcal{C}; \Gamma, [\Delta]\forall x A \vdash_R F}{R \star \mathcal{C}, c \notin [\Delta]\forall x A; \Gamma, [\Delta]A[c/x] \vdash_R F} \forall_I$$

$$\frac{R \star \mathcal{C}; \Gamma, [\Delta]\exists x A \vdash_R F}{R \star \mathcal{C}; \Gamma, [\Delta]A[X/x] \vdash_R F} \exists_I$$

where X new wrt. $\mathcal{C}; \Gamma, [\Delta]\exists x A \vdash_R F$

$$\frac{R \star \mathcal{C}; \Gamma, [\Delta, A \wedge B]G \vdash_R F}{R \star \mathcal{C}; \Gamma, [\Delta, A, B]G \vdash_R F} \wedge_E^H$$

Example (1)

- Start with
  $.;. \vdash_R \forall f, a, l.(\forall \epsilon.\epsilon > 0 \Rightarrow \exists \delta.\delta > 0 \Rightarrow \forall x.(0 <\mid x - a \mid \wedge \mid x - a \mid < \delta) \Rightarrow \mid f(x) - l \mid < \epsilon) \Rightarrow \lim_a f = l$
- Apply elimination rules:
  $\{.;[]\forall \epsilon.\epsilon > 0 \Rightarrow \exists \delta.\delta > 0 \Rightarrow \forall x.(0 <\mid x - A \mid \wedge \mid x - A \mid < \delta) \Rightarrow \mid F(x) - L \mid < \epsilon \vdash_R \lim_A F = L\}$
- Apply introduction rules:

$$\left\{ \begin{array}{l} \mathsf{EV}(\epsilon, \{F, A, L\}), \mathsf{EV}(x, \{F, A, L, D\}); \\ [\epsilon > 0, D > 0, 0 <\mid x - A \mid, \mid x - A \mid < D] \mid F(x) - L \mid < \epsilon \vdash_R \lim_A F = L \end{array} \right\}$$

- $\mathsf{EV}(\epsilon, \{F, A, L\})$ means ϵ must not occur in F, A, L

Example (2)

$$\frac{\begin{array}{c}[\epsilon > 0, D > 0, 0 <\mid x - A \mid, \mid x - A \mid < D] \\ \vdots \\ P :\mid F(x) - L \mid < \epsilon\end{array}}{C : \lim_{A} F = L}$$

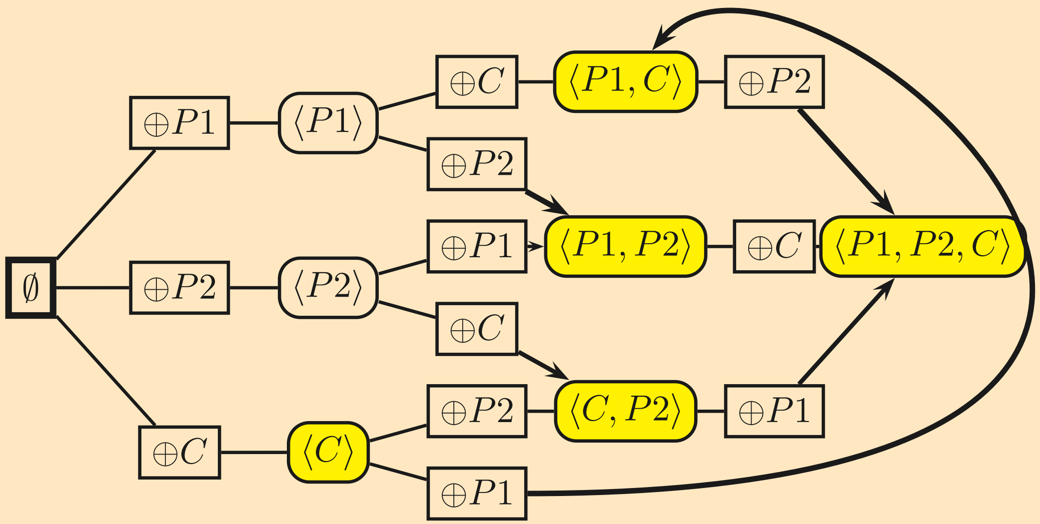
Application Condition: $\mathsf{EV}(\epsilon, \{F, A, L\}) \wedge \mathsf{EV}(x, \{F, A, L, D\})$

Parameters: ϵ, x

III. Generation of Argument Agents

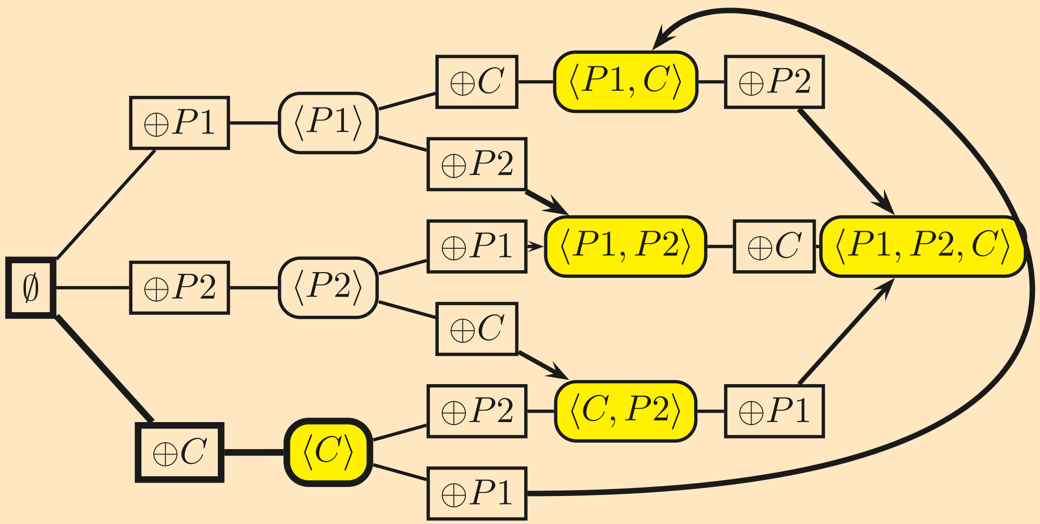
Generation of Argument Agents

- Known the application directions (yellow marked)
- Maintain a set of computable PAI-status
- Round boxes represents initial PAI-statuses
- Squared boxes represent updateses
- Apply stepwise Dijkstra algorithm and add agent



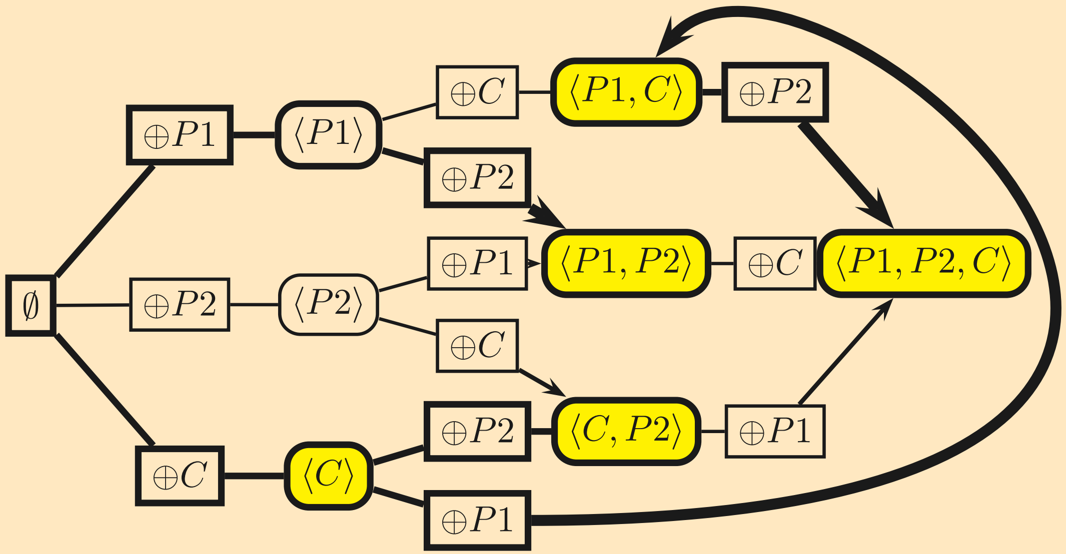
Generation of Argument Agents

- Known the application directions (yellow marked)
- Maintain a set of computable PAI-status
- Round boxes represents initial PAI-statuses
- Squared boxes represent updateses
- Apply stepwise Dijkstra algorithm and add agent



Generation of Argument Agents

- Known the application directions (yellow marked)
- Maintain a set of computable PAI-status
- Round boxes represents initial PAI-statuses
- Squared boxes represent updateses
- Apply stepwise Dijkstra algorithm and add agent



Summary

	Old ΩMEGA	New ΩMEGA
tactics for math. knowledge	manually	automatically
tactics for procedures	manually	manually
application directions	manually	automatically
agent specification	manually	automatically
control knowledge	manually	manually
assertion level	simulated	directly
OR parallelism	no	yes