CS256/Spring 2008 — Lecture #15

Zohar Manna

Particle Tableau

Particle Tableau: Motivation

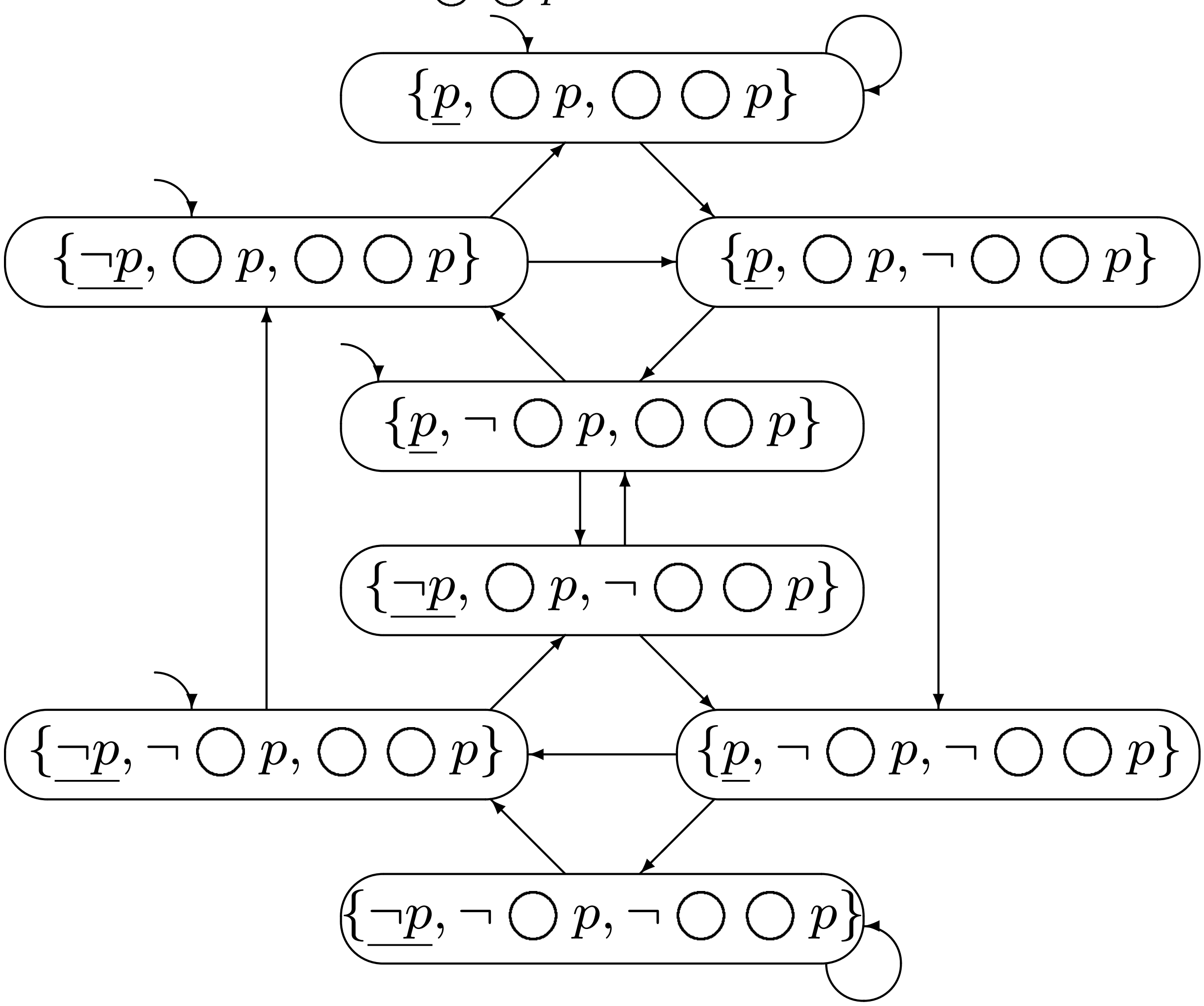
Consider $\boxed{\varphi : \bigcirc \bigcirc p}$

The closure Φ_φ has three basic formulas:

$p, \bigcirc p, \bigcirc \bigcirc p.$

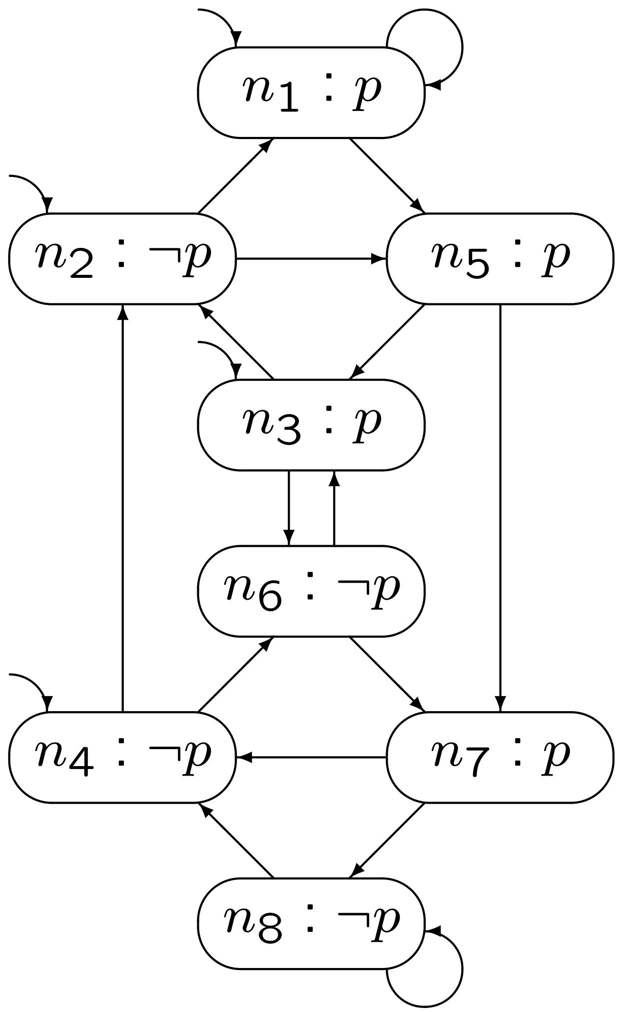
Thus, it has eight atoms.

The atom tableau $T_{\bigcirc \bigcirc p}$ is



Particle Tableau: Motivation

The ω-automaton $A_{\bigcirc \bigcirc p}$:



$$\begin{aligned} \mathcal{F}_M &= \{ \text{ all SCS's } \} \\ \mathcal{F}_S &= \{\} \end{aligned}$$

Note: No promising formulas.

Particle Tableau: Motivation

Because of the atom construction rule:

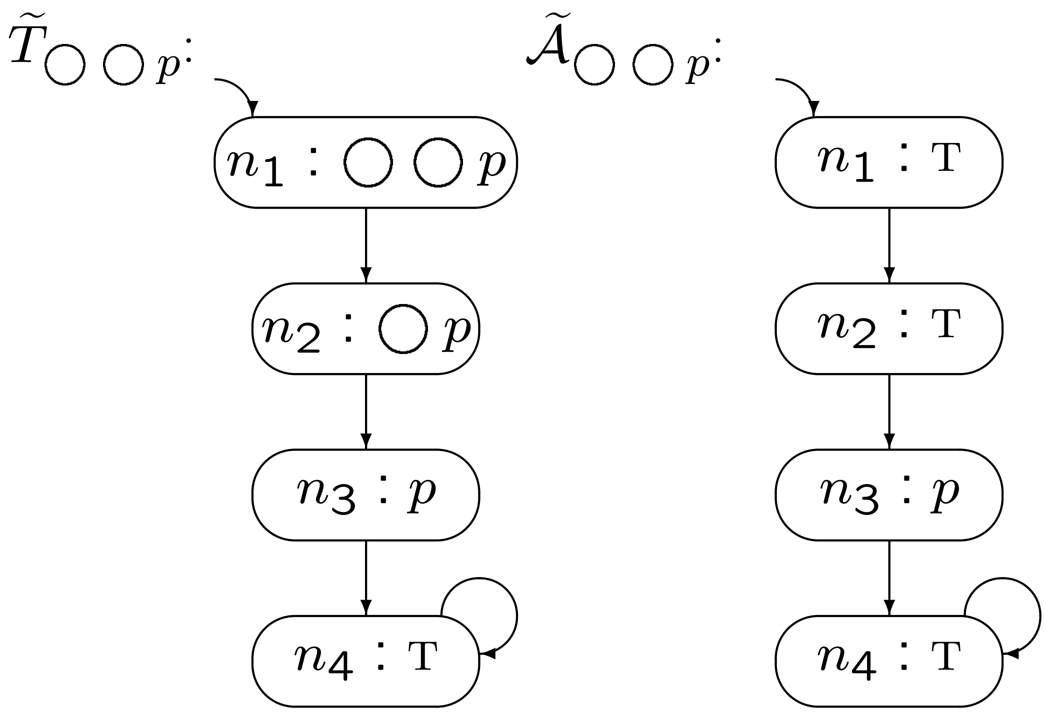
for every $\psi \in \Phi_\varphi$,
$\psi \in A$ iff $\neg\psi \notin A$,

every atom makes a commitment about every formula in the closure.

Clearly, some of these commitments are
irrelevant in determining the satisfiability of the formula.

Particle Tableau: Motivation (Cont'd)

Intuitively, the tableau below should suffice to determine satisfiability. The truth value of p at the first two positions is irrelevant:



If we change the offending rule to

$$\text{if } \psi \in A \text{ then } \neg\psi \notin A$$

we get the particle tableau, which is usually considerably smaller than the atom tableau.

Particles

The idea of a particle is to assert what needs to be true, not what needs to be false, except for state formulas.

Thus, if $\psi \in A$, ψ needs to be true;
if $\psi \notin A$, ψ can be true or false.

Step 0: Push negations inside φ
We push all negations inside the formula such that negations only appear at the state level. This can be done with the help of the following congruences:

$$\begin{array}{rcl} \neg \Diamond p & \approx & \Box \neg p \\ \neg \bigcirc p & \approx & \bigcirc \neg p \\ \neg \Box p & \approx & \Diamond \neg p \\ \neg(p \mathcal{U} q) & \approx & (\neg q) \mathcal{W} (\neg p \land \neg q) \\ \neg(p \mathcal{W} q) & \approx & (\neg q) \mathcal{U} (\neg p \land \neg q) \end{array}$$

Thus, the closure only needs to contain positive formulas and the negation of state formulas.

Closure $\widetilde{\Phi}_\varphi$

- $\varphi \in \widetilde{\Phi}_\varphi$

- for every $\psi \in \widetilde{\Phi}_\varphi$ and χ a subformula of ψ, $\chi \in \widetilde{\Phi}_\varphi$

- for every ψ of the form

$$\Box\psi_1,\ \Diamond\psi_1,\ \psi_1\mathcal{U}\psi_2,\ \psi_1\mathcal{W}\psi_2,$$

if $\psi \in \widetilde{\Phi}_\varphi$,
then $\bigcirc\psi \in \widetilde{\Phi}_\varphi$

Particles: Definition

A particle of φ is any set $P \subseteq \widetilde{\Phi}_\varphi$ that satisfies the following requirements:

- R_{sat}: $\boldsymbol{state}(P)$ is satisfiable

- R_α: for every α-formula $\psi \in \widetilde{\Phi}_\varphi$,
  $$\psi \in P \text{ iff } \kappa(\psi) \in P$$

- R_β: for every β-formula $\psi \in \widetilde{\Phi}_\varphi$,
  $$\psi \in P \text{ iff } \begin{array}{l} \kappa_1(\psi) \in P \\ \text{or } \kappa_2(\psi) \subseteq P \quad \text{(or both)} \end{array}$$

Note: The empty set $\{\}$ is always a particle, denoted by $P_\varnothing$.

Examples:

$$\boxed{\varphi : \ \Diamond \Box p}$$

$\widetilde{\Phi}_\varphi$: $\{ \Diamond \Box p,\ \bigcirc \Diamond \Box p,\ \Box p,\ \bigcirc \Box p,\ p \}$

Particle: $\{ \Diamond \Box p,\ \bigcirc \Diamond \Box p \}$

Atom: $\{ \Diamond \Box p,\ \bigcirc \Diamond \Box p,\ \neg p,$
$\neg \bigcirc \Box p,\ \neg \Box p \}$

$$\boxed{\varphi : \ \bigcirc \bigcirc p}$$

$\widetilde{\Phi}_\varphi$: $\{ \bigcirc \bigcirc p,\ \bigcirc p,\ p \}$

Particle: $\{ \bigcirc \bigcirc p \}$

Atom: $\{ \bigcirc \bigcirc p,\ \bigcirc p,\ \neg p \}$

Cover of a Formula Set

Given a set of formulas $B \subseteq \widetilde{\Phi}_\varphi$, we give a procedure for constructing the cover of B, a set of particles of φ that contain B.

Recursive function $\mathit{cover}_\varphi(B\text{: set of formulas})$: set of particles

- if $\mathit{state}(B)$ is not consistent, then return $\{\}$

- α-expansion
  if for some α-formula $\psi \in \widetilde{\Phi}_\varphi$,
  $\psi \in B$ but $\kappa(\psi) \not\subseteq B$,
  then return
  $\mathit{cover}_\varphi(B \cup \kappa(\psi))$

- α^{-1}-expansion
  if for some α-formula $\psi \in \widetilde{\Phi}_\varphi$,
  $\kappa(\psi) \subseteq B$ but $\psi \notin B$,
  then return
  $\mathit{cover}_\varphi(B \cup \{\psi\})$

- $\underline{\beta\text{-expansion}}$
  if for some β-formula $\psi \in \widetilde{\Phi}_\varphi$,
  $$\psi \in B, \text{ but } \kappa_1(\psi) \notin B \text{ and } \kappa_2(\psi) \not\subseteq B,$$
  then return
  $$\mathit{cover}_\varphi(B \cup \{\kappa_1(\psi)\}) \cup \mathit{cover}_\varphi(B \cup \kappa_2(\psi))$$

- $\underline{\beta^{-1}\text{-expansion}}$
  if for some β-formula $\psi \in \widetilde{\Phi}_\varphi$,
  $$\psi \notin B, \text{ but } \kappa_1(\psi) \in B \text{ or } \kappa_2(\psi) \subseteq B,$$
  then return
  $$\mathit{cover}_\varphi(B \cup \{\psi\})$$

- return $\{B\}$

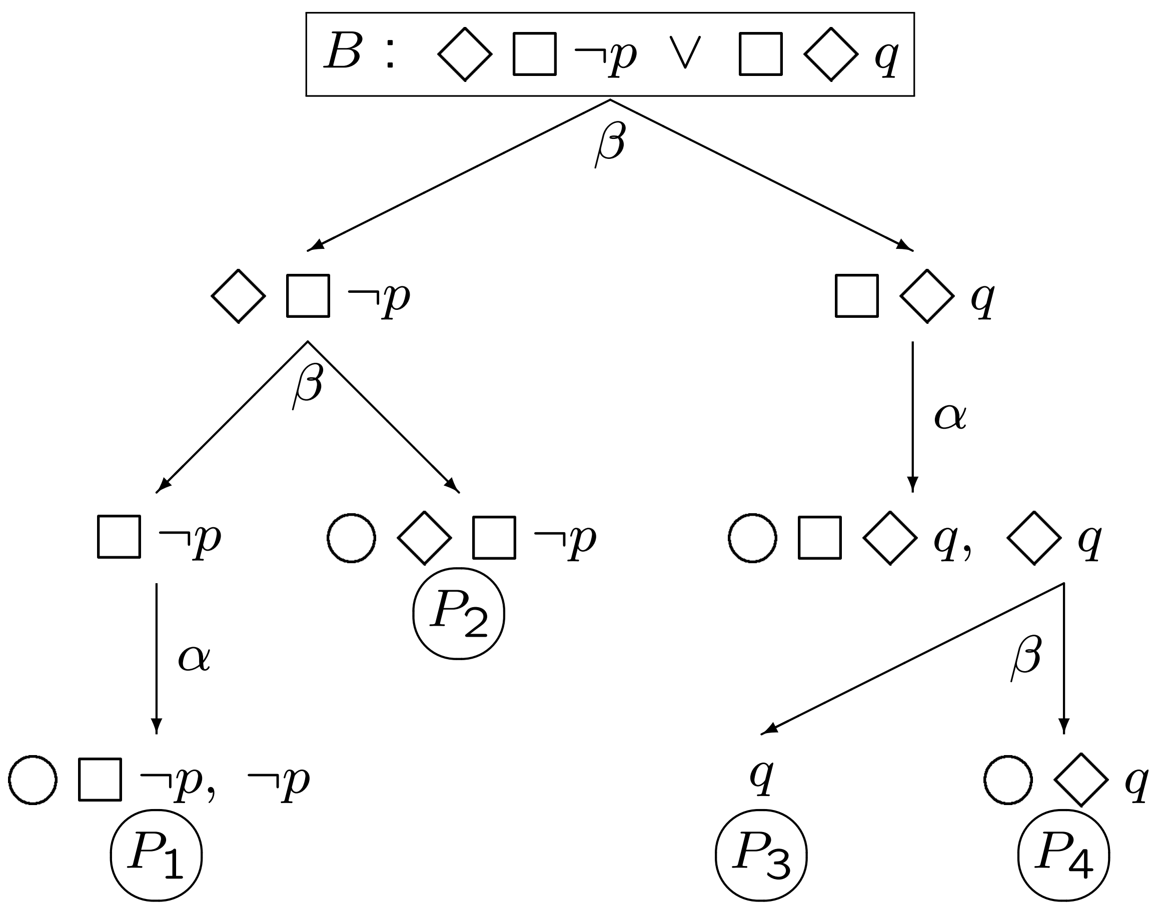
Note: $\mathit{cover}_\varphi(\underbrace{\{\}}_{B}) = \{P_\emptyset\}$

Tree Representation of the Procedure

Example: To find all particles covering

$$B = \varphi : \Diamond \Box \neg p \vee \Box \Diamond q$$

construct the tree:



`Example` (Cont'd): Particles

Thus,

$$cover_{\varphi}(\underbrace{\{\varphi\}}_{B}) = \{P_1, P_2, P_3, P_4\},$$

where

$$P_1 : \ \{\ \varphi,\ \Diamond \Box \neg p,\ \Box \neg p,\ \bigcirc \Box \neg p,\ \neg p\ \}$$

$$P_2 : \ \{\ \varphi,\ \Diamond \Box \neg p,\ \bigcirc \Diamond \Box \neg p\}$$

$$P_3 : \ \{\ \varphi,\ \Box \Diamond q,\ \bigcirc \Box \Diamond q,\ \Diamond q,\ q\ \}$$

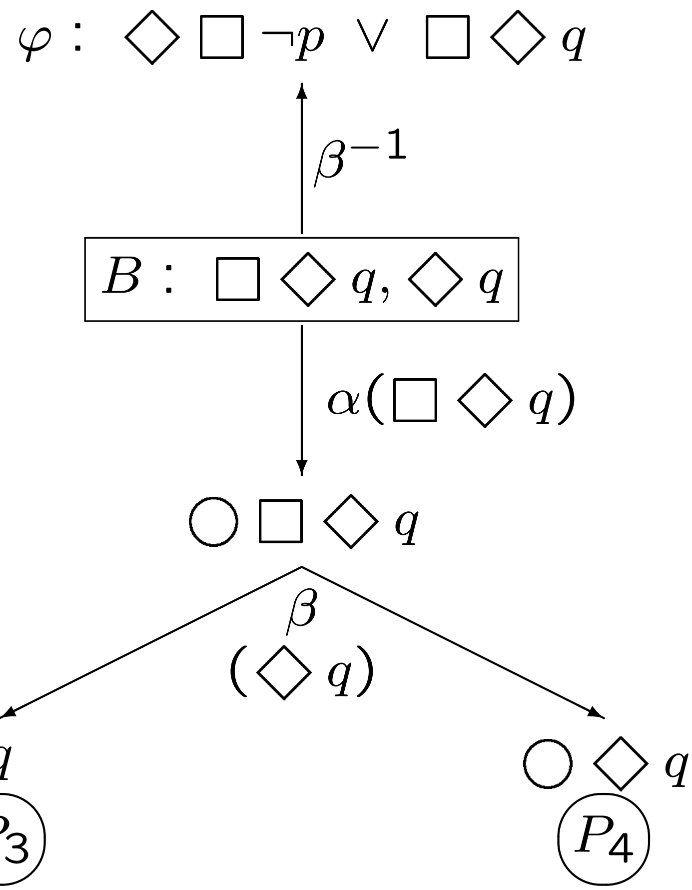
$$P_4 : \ \{\ \varphi,\ \Box \Diamond q,\ \bigcirc \Box \Diamond q,\ \Diamond q,\ \bigcirc \Diamond q\ \}$$

__`Example`__: $\varphi: \ \Diamond \Box \neg p \ \vee \ \Box \Diamond q$

To find all particles of φ covering

$$B: \ \{\Box \Diamond q, \Diamond q\}$$

construct the tree:



Thus, $cover_{\varphi}(\underbrace{\{\Box \Diamond q, \Diamond q\}}_{B}) = \{P_3, P_4\}$

Incremental Particle Tableau Construction

Idea: Start with initial φ-particles and only construct particles that are reachable from
previously constructed particles.

Implied successors $\boldsymbol{imps}(P)$ of particle P:

if $\bigcirc\,\psi \in P$, then $\psi \in \boldsymbol{imps}(P)$

Successors of particle P:

$$\boldsymbol{succ}(P) = \boldsymbol{cover}_\varphi(\boldsymbol{imps}(P))$$

Algorithm for constructing $\widetilde{T}_\varphi$:

- initially, $\widetilde{T}_\varphi = \boldsymbol{cover}_\varphi(\{\varphi\})$
  these are the initial nodes.
- for each particle $P \in \widetilde{T}_\varphi$,
  let $S = \boldsymbol{succ}(P)$
  for each $Q \in S$,
  if $Q \notin \widetilde{T}_\varphi$, then add it
  draw an edge from P to Q

Example: Construct $\widetilde{T}_\varphi$ for $\varphi: \ \diamondsuit \square \neg p \ \vee \ \square \diamondsuit q$

Particles:

$P_1: \ \{ \varphi, \ \diamondsuit \square \neg p, \ \square \neg p, \ \underline{\bigcirc \square \neg p}, \ \neg p \ \}$

$P_2: \ \{ \varphi, \ \diamondsuit \square \neg p, \ \underline{\bigcirc \diamondsuit \square \neg p} \ \}$

$P_3: \ \{ \varphi, \ \square \diamondsuit q, \ \underline{\bigcirc \square \diamondsuit q}, \ \diamondsuit q, \ q \ \}$

$P_4: \ \{ \varphi, \ \square \diamondsuit q, \ \underline{\bigcirc \square \diamondsuit q}, \ \diamondsuit q, \ \underline{\bigcirc \diamondsuit q} \ \}$

$$\begin{aligned} imps(P_1) &= \ \{ \ \square \neg p \ \} \\ succ(P_1) &= \ cover(\{ \square \neg p \ \}) \ = \ \{P_1\} \end{aligned}$$

$$\begin{aligned} imps(P_2) &= \ \{ \ \diamondsuit \square \neg p \ \} \\ succ(P_2) &= \ cover(\{ \diamondsuit \square \neg p \ \}) \ = \ \{P_1, P_2\} \end{aligned}$$

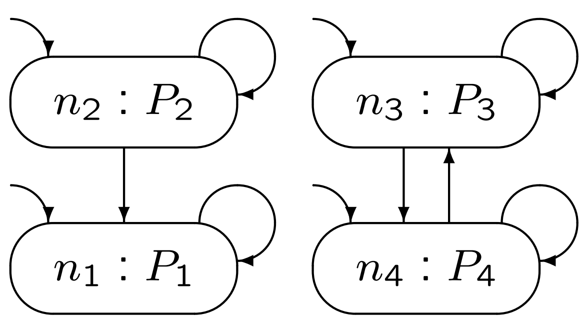
$$\begin{aligned} imps(P_3) &= \ \{ \ \square \diamondsuit q \ \} \\ succ(P_3) &= \ cover(\{ \square \diamondsuit q \ \}) \ = \ \{P_3, P_4\} \end{aligned}$$

$$\begin{aligned} imps(P_4) &= \ \{ \ \square \diamondsuit q, \ \diamondsuit q \ \} \\ succ(P_4) &= \ cover(\{ \square \diamondsuit q, \ \diamondsuit q\}) \ = \ \{P_3, P_4\} \end{aligned}$$

Example (Cont'd):

Particle	$imps$	$succ$
P_1	$\Box\neg p$	P_1
P_2	$\Diamond\Box\neg p$	P_1, P_2
P_3	$\Box\Diamond q$	$P_3,\ P_4$
P_4	$\Box\Diamond q,\ \ \Diamond q$	P_3, P_4

$\widetilde{T}_\varphi$:



Fulfillment

A particle P fulfills formula $\psi \in \widetilde{\Phi}_\varphi$, which promises r, if

$$\psi \notin P \quad \text{or} \quad r \in P.$$

An SCS S is fulfilling if every promising formula $\psi \in \widetilde{\Phi}_\varphi$ is fulfilled by some particle $P \in S$.

Proposition:
An LTL formula φ is satisfiable

iff

$\widetilde{T}_\varphi$ has a fulfilling SCS that is reachable from an initial node.

Example: $\varphi : \diamondsuit \square \neg p \;\vee\; \square \diamondsuit q$

Promising formulas:

$$\diamondsuit \square \neg p \;\text{ promises }\; \square \neg p$$
$$\diamondsuit q \;\text{ promises }\; q$$

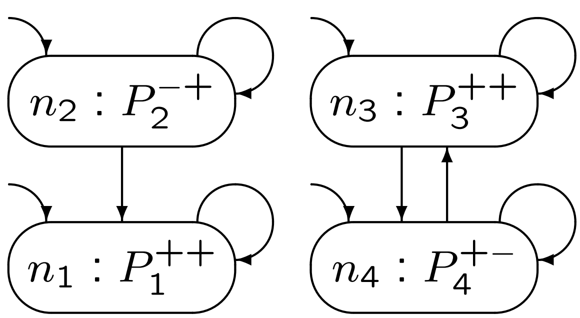
P_1^{++} : $\{\, \varphi,\ \diamondsuit \square \neg p,\ \square \neg p,\ \bigcirc \square \neg p,\ \neg p \,\}$

P_2^{-+} : $\{\, \varphi,\ \diamondsuit \square \neg p,\ \bigcirc \diamondsuit \square \neg p \}$

P_3^{++} : $\{\, \varphi,\ \square \diamondsuit q,\ \bigcirc \square \diamondsuit q,\ \diamondsuit q,\ q \,\}$

P_4^{+-} : $\{\, \varphi,\ \square \diamondsuit q,\ \bigcirc \square \diamondsuit q,\ \diamondsuit q,\ \bigcirc \diamondsuit q \,\}$

Particle Tableau $\widetilde{T}_\varphi$:



Fulfilling SCS's :$\{n_1\}, \{n_3\}, \{n_3, n_4\}$
Hence, φ is satisfiable.

From Particle Tableau $\widetilde{T}_\varphi$ to ω-Automaton $\mathcal{A}_\varphi$

For temporal formula φ, construct the ω-automaton

$$\mathcal{A}_\varphi : \ \langle \underbrace{N,\ N_0,\ E,}_{\substack{\text{Same as} \\ \widetilde{T}_\varphi}}\ \mu,\ \mathcal{F} \rangle$$

where

- Node labeling μ:
  For node $n \in N$ labeled by particle P in $\widetilde{T}_\varphi$,

  $$\mu(n) \ = \ state(P).$$

- Acceptance condition $\mathcal{F}$:

  Muller:

  $$\mathcal{F} \ = \ \{\text{SCS } S \mid S \text{ is fulfilling } \}$$

  Street:

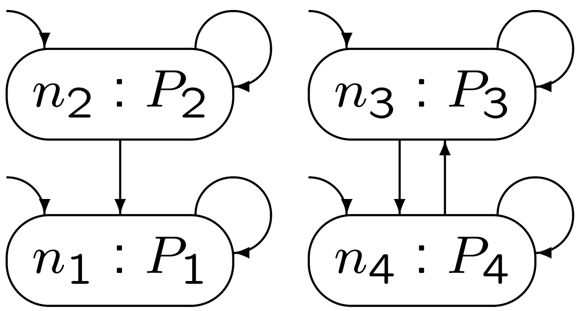
  $$\mathcal{F} \ = \ \{(P_\psi, R_\psi) \mid \psi \in \widetilde{\Phi}_\varphi \text{ promises } r\},$$

  where

  $$\begin{array}{lcl} P_\psi & = & \{ \ P \mid \psi \not\in P \ \} \quad \Leftarrow \\ R_\psi & = & \{ \ P \mid r \in P \ \} \end{array}$$

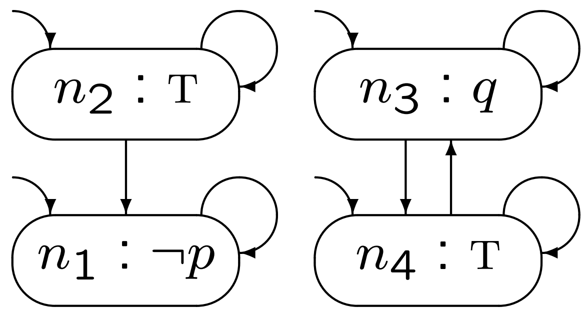
Example (Cont'd): $\varphi: \ \diamondsuit \square \neg p \ \vee \ \square \diamondsuit q$

Particle Tableau $\widetilde{T}_\varphi$



with fulfilling SCS's
$\{n_1\}, \{n_3\}, \{n_3, n_4\}$

Corresponding
ω-Automaton $\mathcal{A}_\varphi$



$$\mathcal{F}_M = \{\{n_1\}, \{n_3\}, \{n_3, n_4\}\}$$

$$\mathcal{F}_S = \{(P_{\diamondsuit \square \neg p}, R_{\diamondsuit \square \neg p}), (P_{\diamondsuit q}, R_{\diamondsuit q})\}$$

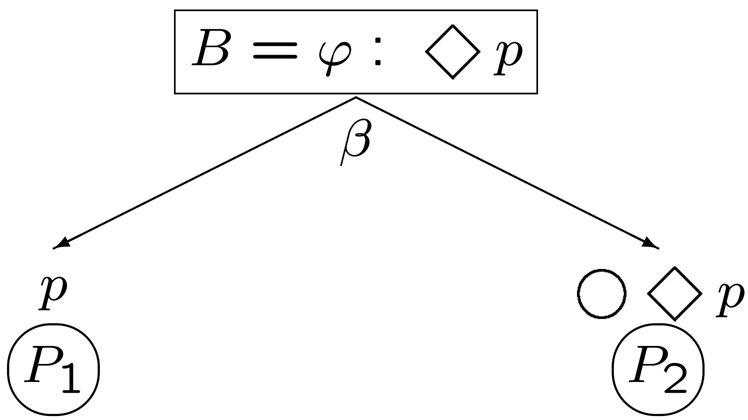
with

$P_{\diamondsuit \square \neg p}: \ \{n_3, n_4\}$
$R_{\diamondsuit \square \neg p}: \ \{n_1\}$
$P_{\diamondsuit q}: \ \{n_1, n_2\}$
$R_{\diamondsuit q}: \ \{n_3\}$

`Example`: To find all particles covering

$$\varphi : \Diamond p$$

construct the tree:



Thus, $cover_{\varphi}(\underbrace{\{\varphi\}}_{B}) = \{P_1, P_2\}$, where

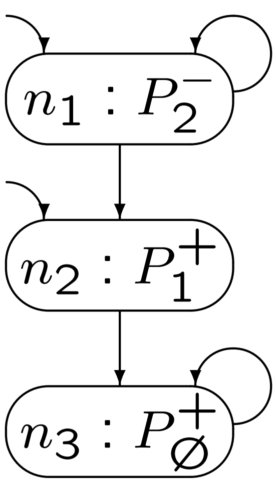
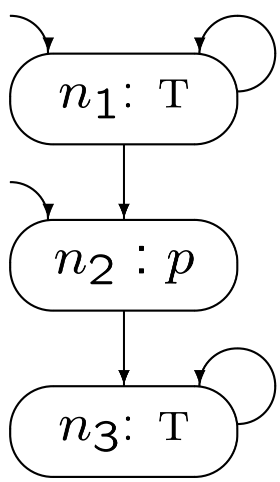
$$P_1 : \{ \varphi, \; p \}$$

$$P_2 : \{ \varphi, \; \bigcirc \Diamond p \}$$

Example (Cont'd): $\varphi : \Diamond p$

$P_1 : \{\varphi, p\}$	$imps(P_1) = \{\}$	$succ(P_1) = \{P_\emptyset\}$
$P_\emptyset$	$imps(P_\emptyset) = \{\}$	$succ(\{\}) = \{P_\emptyset\}$
$P_2 : \{\varphi, \bigcirc \Diamond p\}$	$imps(P_2) = \{\varphi\}$	$succ(P_2) = \{P_1, P_2\}$

$P_1^+, P_\emptyset^+$ fulfilling $\quad P_2^-$ not fulfilling.

Particle Tableau $\widetilde{T}_\varphi$	Corresponding ω-Automaton $\mathcal{A}_\varphi$
	
with fulfilling SCS's $\{n_3\}$	$\mathcal{F}_M = \{\{n_3\}\}$ $\mathcal{F}_S = \{(\{n_3\}, \{\cancel{n_2}\})\}$

Hence, φ is satisfiable.

Example: To find all particles covering

$$\varphi: \ \bigcirc \bigcirc p$$

construct the (trivial) tree:

$$\boxed{B = \varphi : \bigcirc \bigcirc p}$$

(only one node)

Thus,

$$cover_\varphi(\underbrace{\{\bigcirc \bigcirc p\}}_{B}) = \{P_1\},$$

where

$$P_1 : \ \{ \ \varphi \ \}.$$

<u>`Example`</u> (Cont'd): $\varphi : \bigcirc \bigcirc p$

$P_1 : \{\varphi\}$ $\quad imps(P_1) = \{\bigcirc p\}$ $\quad succ(P_1) = \{\underbrace{\bigcirc p}_{P_2}\}$

$P_2 : \{\bigcirc p\}$ $\quad imps(P_2) = \{p\}$ $\quad succ(P_2) = \{\underbrace{p}_{P_3}\}$

$P_3 : \{p\}$ $\quad imps(P_3) = \{\}$ $\quad succ(P_3) = \{P_\emptyset\}$

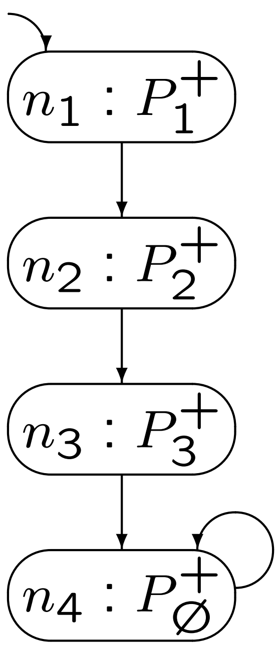
$P_\emptyset$ $\quad imps(P_\emptyset) = \{\}$ $\quad succ(P_\emptyset) = \{P_\emptyset\}$

No promising formulas

$$P_1^+ : \{\varphi\}$$
$$P_2^+ : \{\bigcirc p\}$$
$$P_3^+ : \{p\}$$
$$P_4^+ : P_\emptyset$$

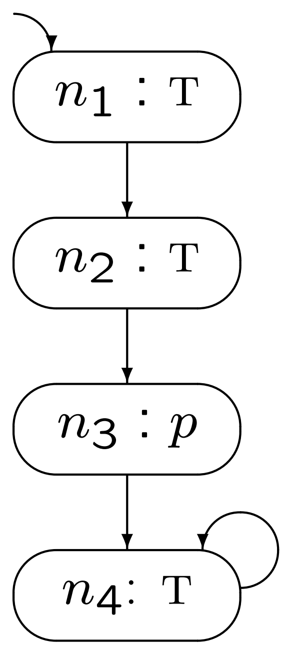
Example (Cont'd): $\varphi : \bigcirc \bigcirc p$

Particle Tableau $\widetilde{T}_\varphi$



with fulfilling SCS $\{n_4\}$

Corresponding ω-Automaton $\mathcal{A}_\varphi$



$\mathcal{F}_M = \{\{n_4\}\}$

$\mathcal{F}_S = \{\}$

Hence, φ is satisfiable.