CS256/Winter 2009 Lecture #15

Zohar Manna

Particle Tableau

Particle Tableau: Motivation

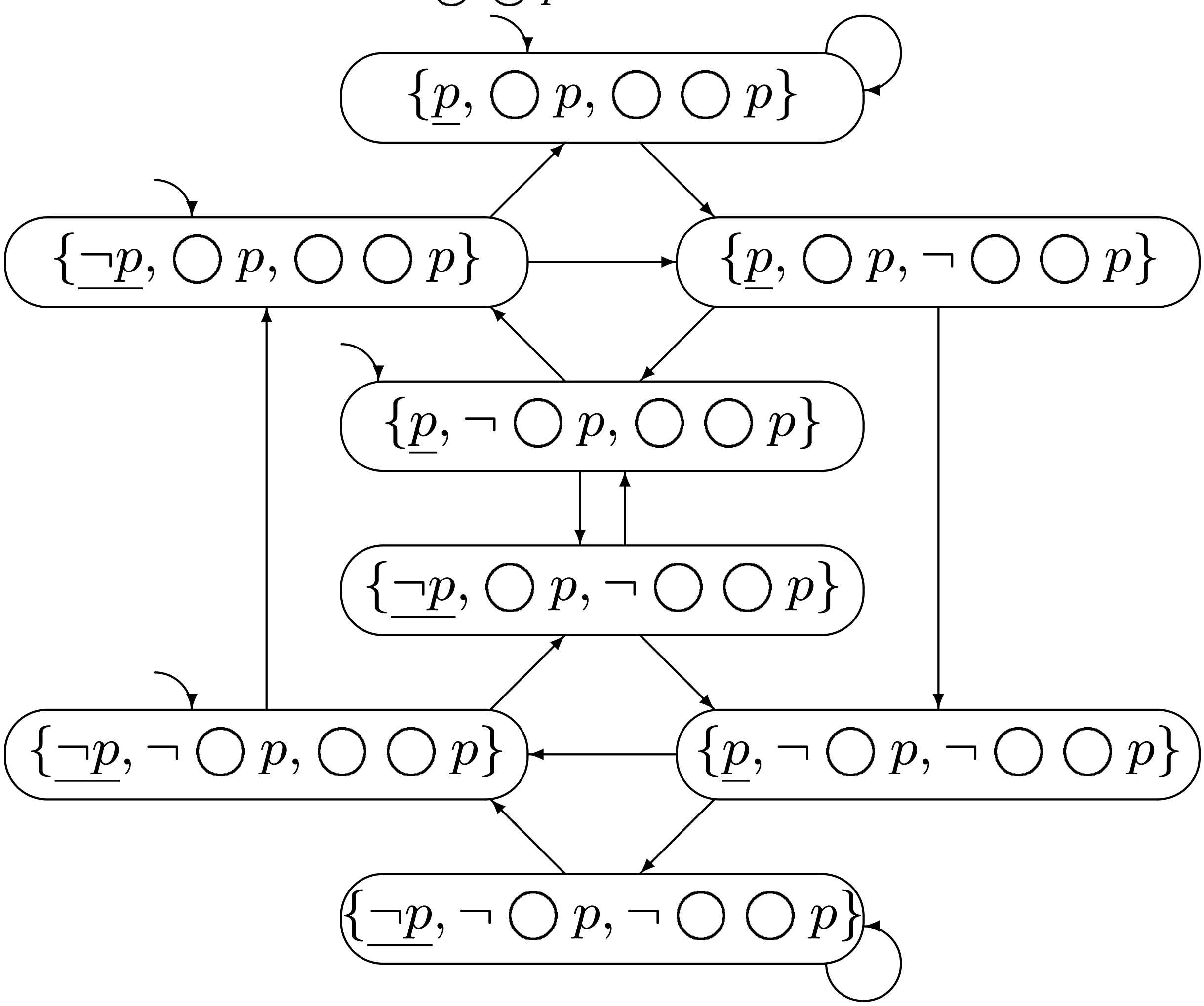
Consider $\boxed{\varphi:\ \bigcirc\bigcirc p}$

The closure Φ_φ has three basic formulas:

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

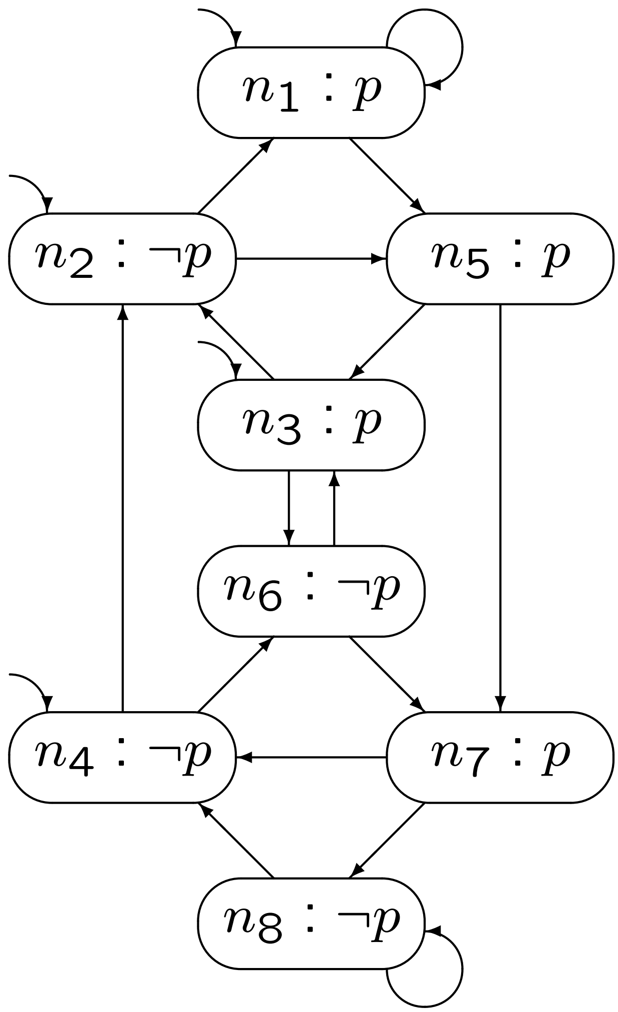
Thus, it has eight atoms.

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



Particle Tableau: Motivation

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



$$\mathcal{F}_M = \{ \text{ all SCS's } \}$$
$$\mathcal{F}_S = \{\}$$

Note: No promising formulas.

Particle Tableau: Motivation

Because of the atom construction rule:

for every $\psi \in \Phi_\varphi$,

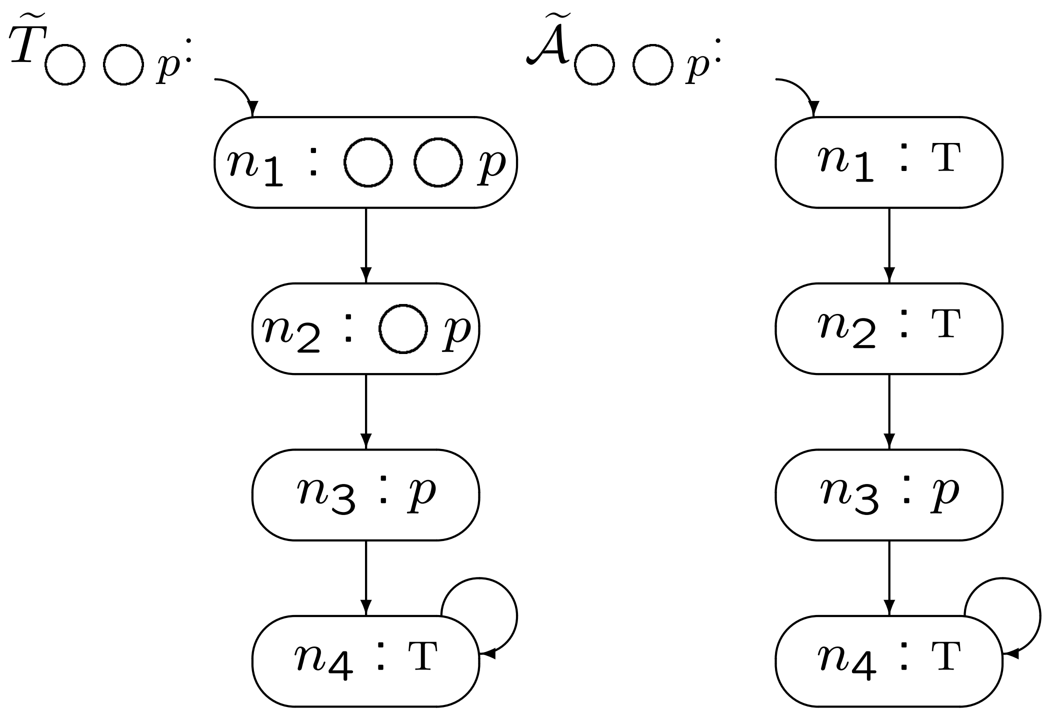
$\psi \in A$ iff $\neg\psi \notin A$,

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:

$$
\begin{aligned}
\neg \Diamond p &\approx \Box \neg p \\
\neg \bigcirc p &\approx \bigcirc \neg p \\
\neg \Box p &\approx \Diamond \neg p \\
\neg(p \mathcal{U} q) &\approx (\neg q) \mathcal{W} (\neg p \wedge \neg q) \\
\neg(p \mathcal{W} q) &\approx (\neg q) \mathcal{U} (\neg p \wedge \neg q)
\end{aligned}
$$

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 $P_\emptyset$.

Examples:

$$\boxed{\varphi : \ \diamond \square p}$$

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

Particle: $\{ \diamond \square p,\ \bigcirc \diamond \square p \}$

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

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

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

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 $\boldsymbol{cover}_\varphi(B$: set of formulas): set of particles

- if $\boldsymbol{state}(B)$ 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
  $\boldsymbol{cover}_\varphi(B \cup \kappa(\psi))$

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

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

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

- return $\{B\}$

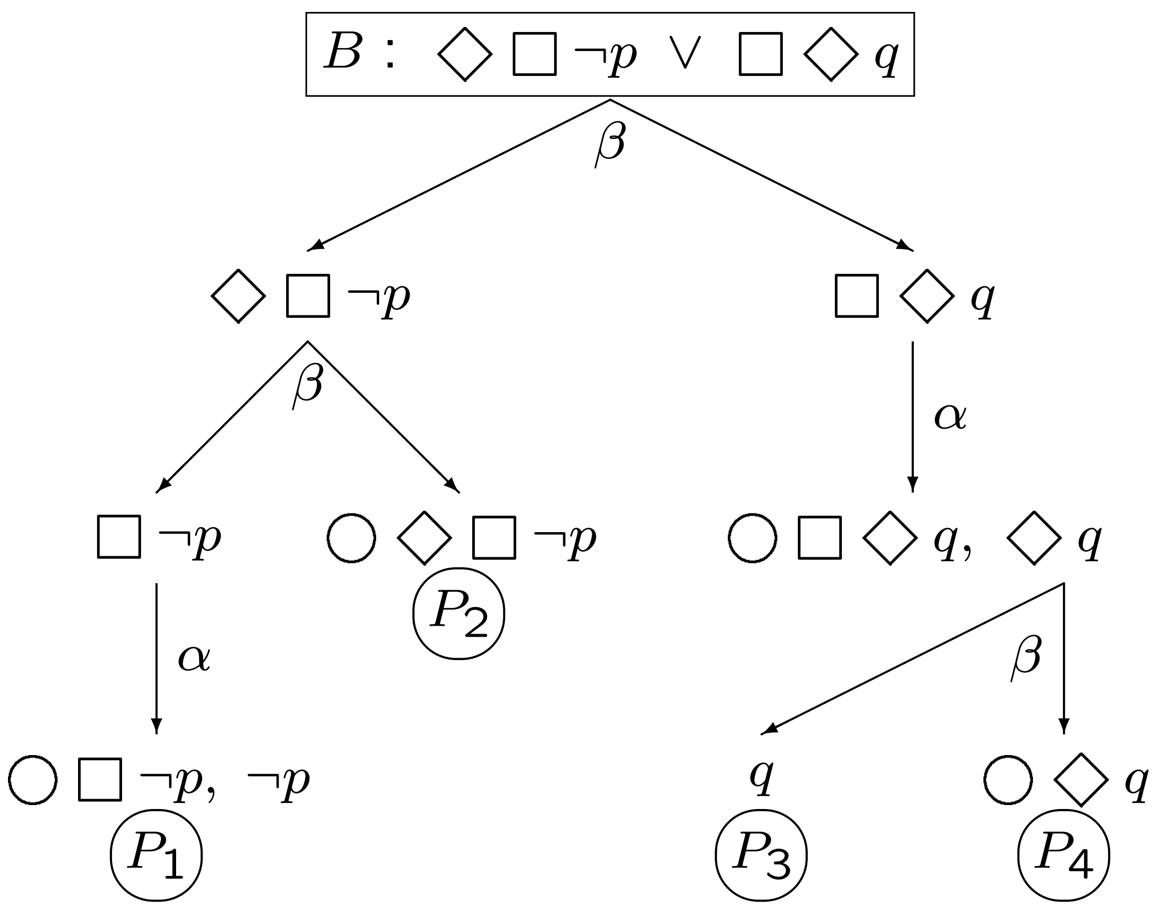
Note: $cover_\varphi(\underbrace{\{\}}_{B}) = \{P_\varnothing\}$

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

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

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

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

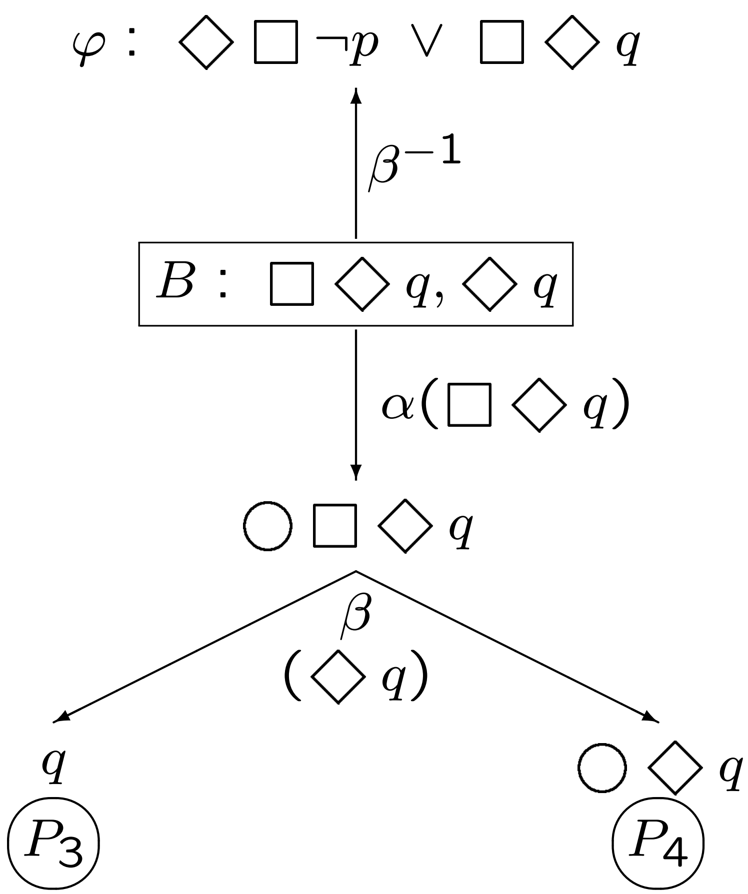
$$P_4: \ \{\ \varphi,\ \Box\Diamond q,\ \bigcirc\Box\Diamond q,\ \Diamond q,\ \bigcirc\Diamond q\ \}$$

<u>`Example`</u>: $\varphi:\ \Diamond\Box\neg p \ \vee\ \Box\Diamond q$

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 $imps(P)$ of particle P:

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

Successors of particle P:

$$succ(P) = cover_{\varphi}(imps(P))$$

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

- initially, $\widetilde{T}_{\varphi} = cover_{\varphi}(\{\varphi\})$
  these are the initial nodes.
- for each particle $P \in \widetilde{T}_{\varphi}$,
  let $S = succ(P)$
  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:

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

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

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

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

$$imps(P_1) = \{\ \square\,\neg p\ \}$$
$$succ(P_1) = cover(\{\square\,\neg p\ \}) = \{P_1\}$$

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

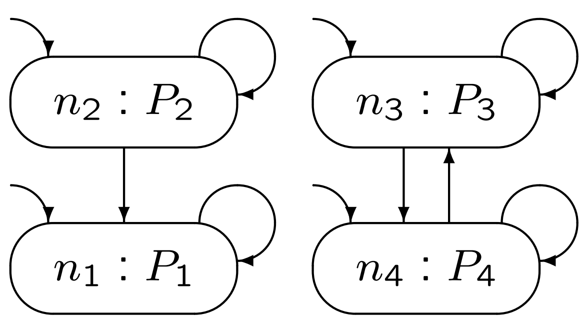
$$imps(P_3) = \{\ \square\,\diamondsuit\,q\ \}$$
$$succ(P_3) = cover(\{\square\,\diamondsuit\,q\ \}) = \{P_3, P_4\}$$

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

<u>`Example`</u> (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`: $\varphi : \diamond \Box \neg p \;\vee\; \Box \diamond q$

Promising formulas:

$$\diamond \Box \neg p \;\; \text{promises} \;\; \Box \neg p$$
$$\diamond q \;\; \text{promises} \;\; q$$

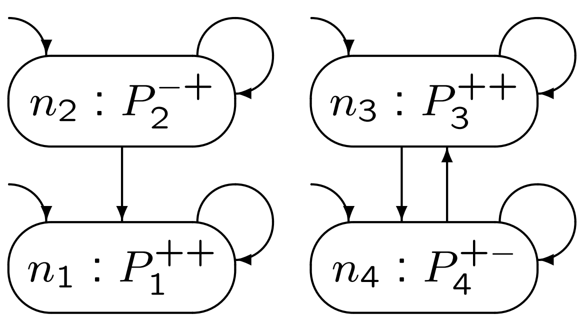
P_1^{++} : $\{ \varphi,\ \diamond \Box \neg p,\ \Box \neg p,\ \bigcirc \Box \neg p,\ \neg p \}$

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^{+-} : $\{ \varphi,\ \Box \diamond q,\ \bigcirc \Box \diamond q,\ \diamond q,\ \bigcirc \diamond q \}$

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 $\mathcal{F}$:

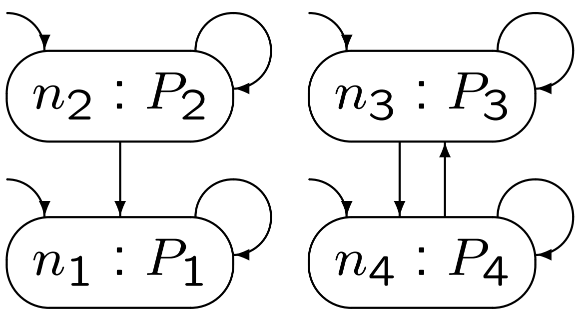
  Muller:
  $$\mathcal{F} \ = \ \{\text{SCS } S \mid S \text{ is fulfilling } \}$$

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

  $$\begin{array}{lcll} P_\psi & = & \{ \ P \mid \psi \notin P \ \} & \Leftarrow \\ R_\psi & = & \{ \ P \mid r \in P \ \} & \end{array}$$

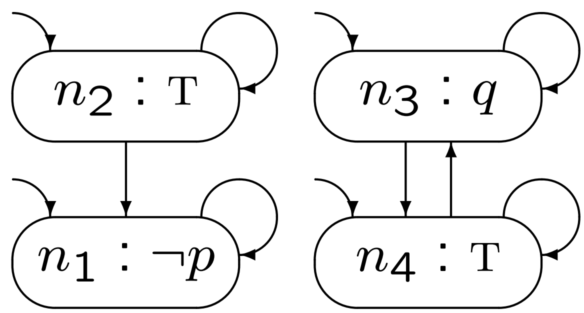
<u>`Example`</u> (Cont'd): $\varphi: \ \diamondsuit \square \neg p \ \vee \ \square \diamondsuit q$

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

$P_{\diamondsuit \square \neg p}: \ \{n_3, n_4\}$

$R_{\diamondsuit \square \neg p}: \ \{n_1\}$

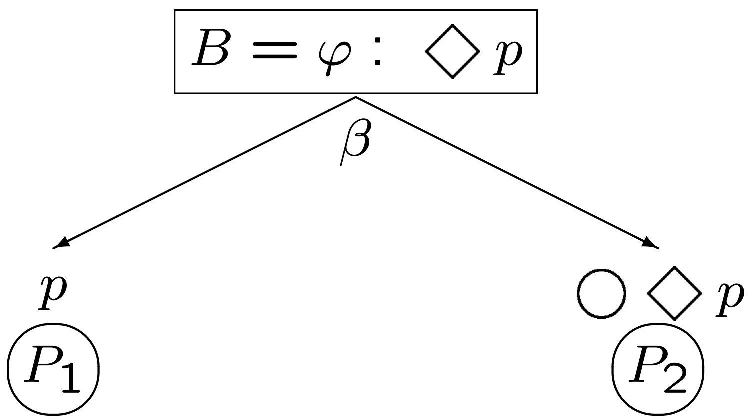
$P_{\diamondsuit q}: \ \{n_1, n_2\}$

$R_{\diamondsuit q}: \ \{n_3\}$

Example: To find all particles covering

$$\varphi: \ \Diamond p$$

construct the tree:



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

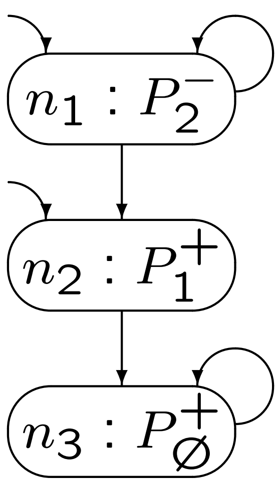
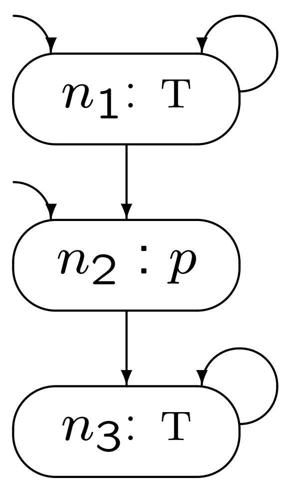
$$P_1: \ \{ \ \varphi, \ p \ \}$$

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

<u>`Example`</u> (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

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

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

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

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

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

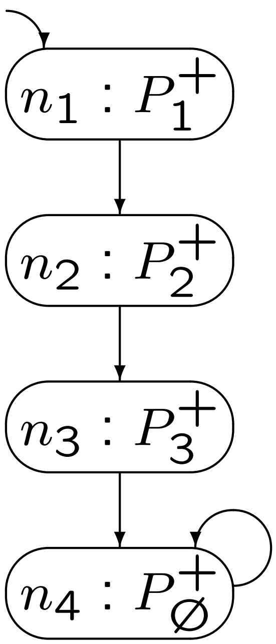
$$P_\emptyset \qquad imps(P_\emptyset) = \{\} \qquad succ(P_\emptyset) = \{P_\emptyset\}$$

No promising formulas

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

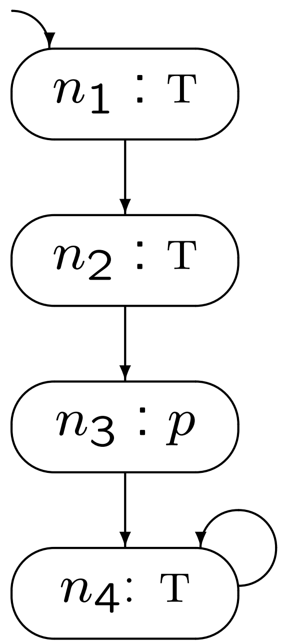
Example (Cont'd): $\varphi : \bigcirc \bigcirc p$

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.