重み付き有限オートマトンに対する曖昧性解消演算の一般化とその応用

†林 克彦　　永田 昌明

NTT CS Lab
†hayashi.katsuhiko@lab.ntt.co.jp

オートマトンの曖昧性

決定化演算の問題

一般化曖昧性解消演算

重み付き木オートマトンへの一般化

実験

自然言語処理タスクへの適用

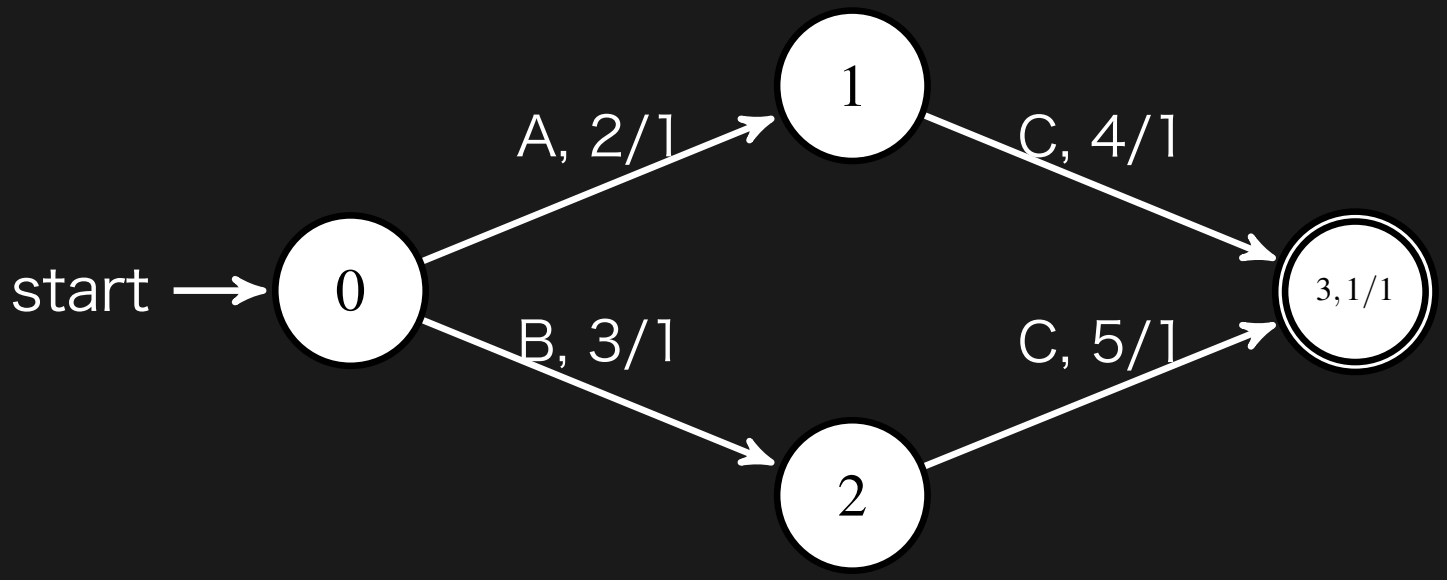
まとめ

今後の課題

オートマトンの曖昧性

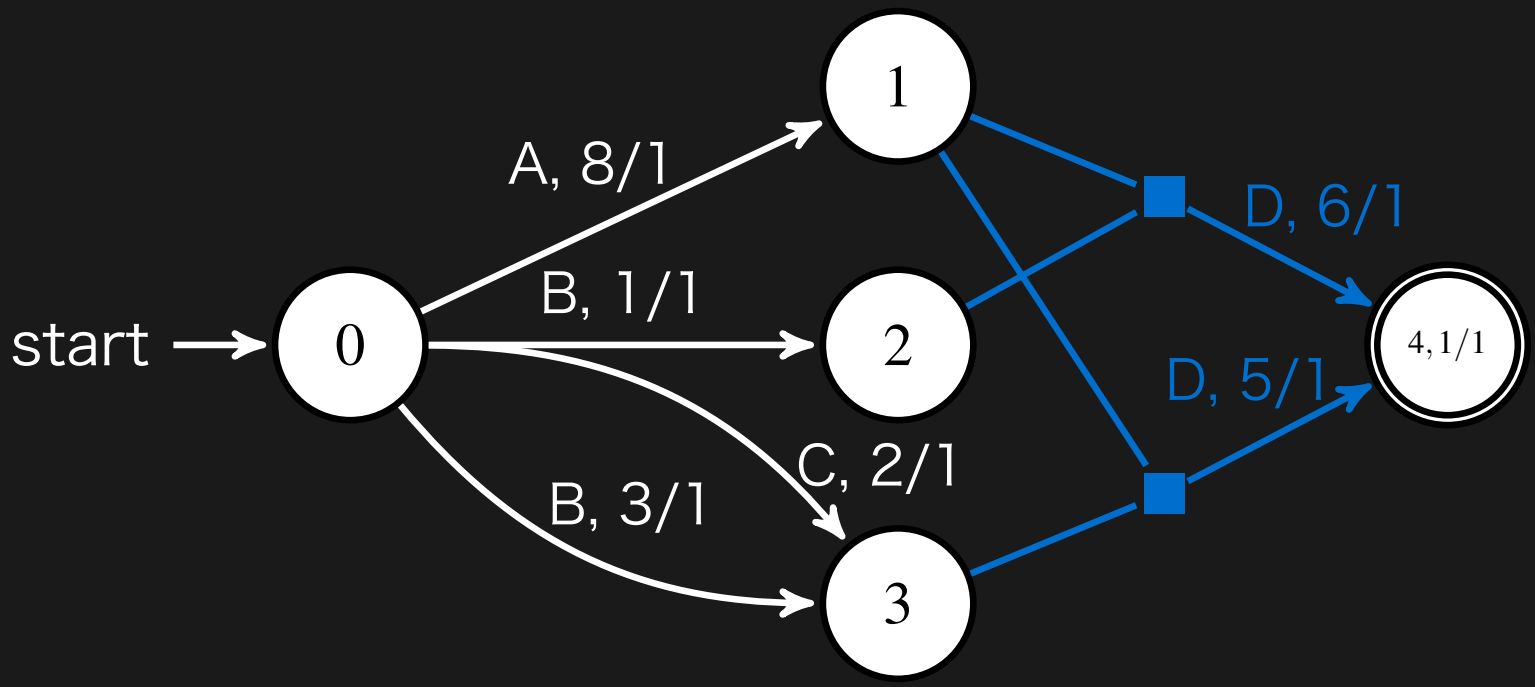
決定化演算の問題

重み付き有限オートマトン



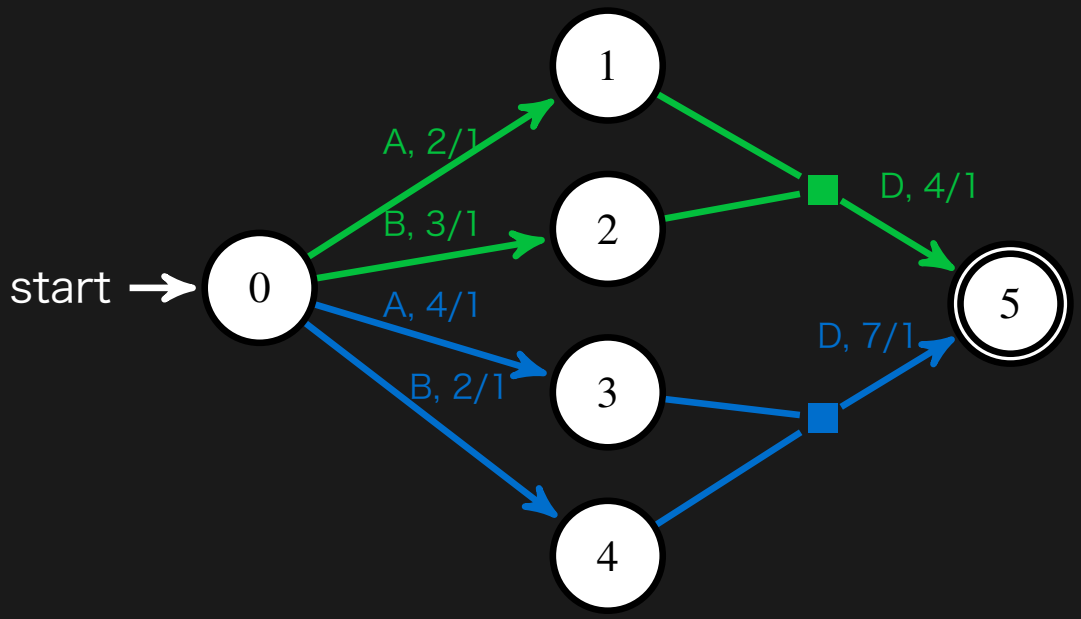
- $A = \{\Sigma, Q, i, F, E, \rho\}$
  - 記号集合Σ : {A,B,C}
  - 状態集合Q : $\{0,1,2,3\}$, 初期状態i : 0, 終了状態集合F : $\{3\}$
  - 遷移集合E : $\{\mathrm{A}(0) \xrightarrow{2/1} 1, \mathrm{B}(0) \xrightarrow{3/1} 2, \mathrm{C}(1) \xrightarrow{4/1} 3, \mathrm{C}(2) \xrightarrow{5/1} 3\}$
  - 終了重み関数ρ : $\{\rho(3) = 1/1\}$
- 言語$L(A)$: {AC,BC}

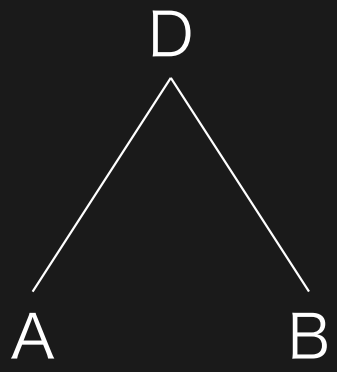
重み付き木オートマトン (Thatcher, 68)



- $A = \{\Sigma, Q, i, F, E, \rho\}$
  - ランク付き記号集合Σ : $\{\mathrm{A}^{(1)}, \mathrm{B}^{(1)}, \mathrm{C}^{(1)}, \mathrm{D}^{(2)}\}$
  - 遷移集合E : $\{\mathrm{A}(0) \xrightarrow{8/1} 1, \mathrm{B}(0) \xrightarrow{1/1} 2, \mathrm{B}(0) \xrightarrow{2/1} 3, \mathrm{C}(0) \xrightarrow{3/1} 3,$ $\mathrm{D}(1,2) \xrightarrow{6/1} 4,\ \mathrm{D}(1,3) \xrightarrow{5/1} 4\}$
- 木言語$L(A)$: $\{$  ,  $\}$

オートマトンが曖昧性を持つとは?





- 同じ木に対応する2つ以上の受理経路が存在
  - ある木に対する重みが複数の受理経路に分割
    - 真の1-bestやK-best解を効率的に求められない (NP困難: Sima'an 96)
    - K-best解に同じ木が存在 (疑似曖昧性の問題)

曖昧性解消の目的と従来のアプローチ

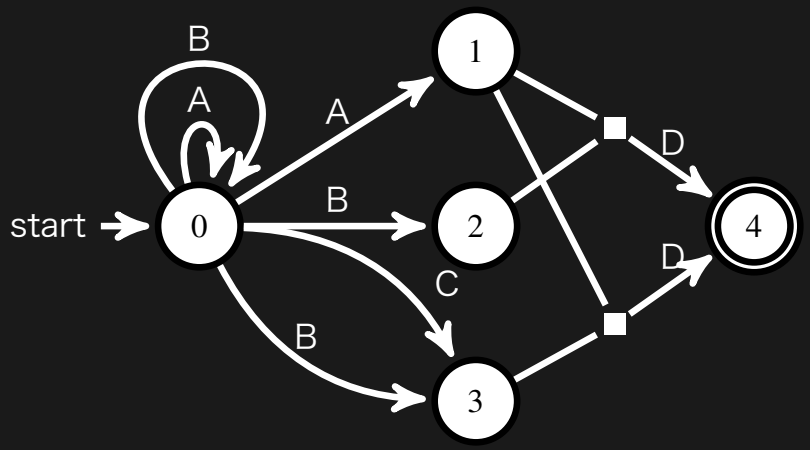
- 曖昧性解消の目的
  - ある木に対する複数の受理経路を1つに統合
  - その統合した受理経路に各受理経路の重みを合計

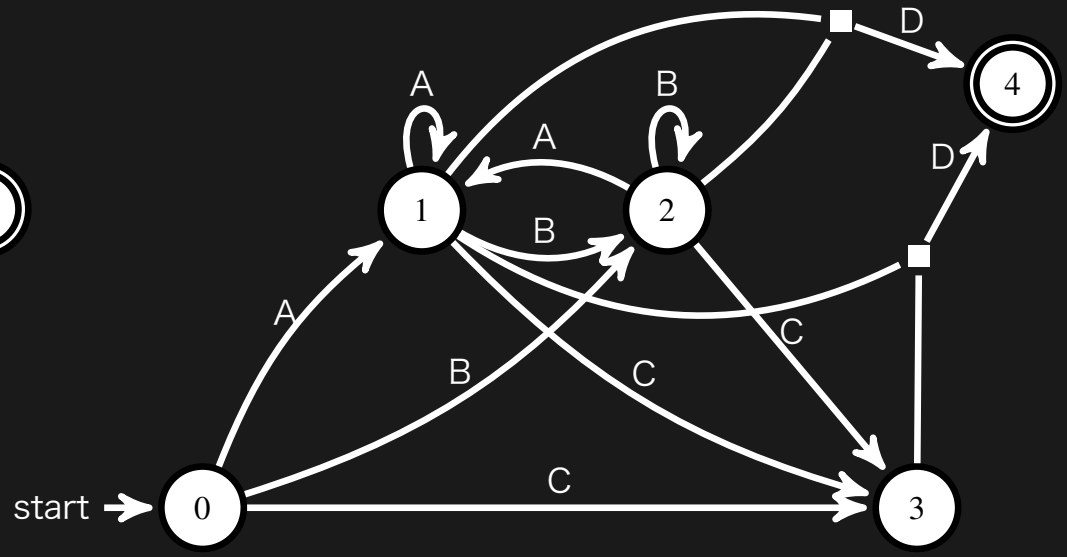
曖昧性解消の目的と従来のアプローチ

- 曖昧性解消の目的
  - ある木に対する複数の受理経路を1つに統合
  - その統合した受理経路に各受理経路の重みを合計
- 従来のアプローチ: 決定化演算 (Mohri 96, May 06)
  - ある状態(のベクトル)から同じ記号で
    の遷移を許さない等価なオートマトンへ変換 ⇒ 決定性, 無曖昧性

決定性オートマトンの問題

- 重み付き木オートマトンの決定化演算 (閉路無し: May 06, 閉路有り: Büchse 09)

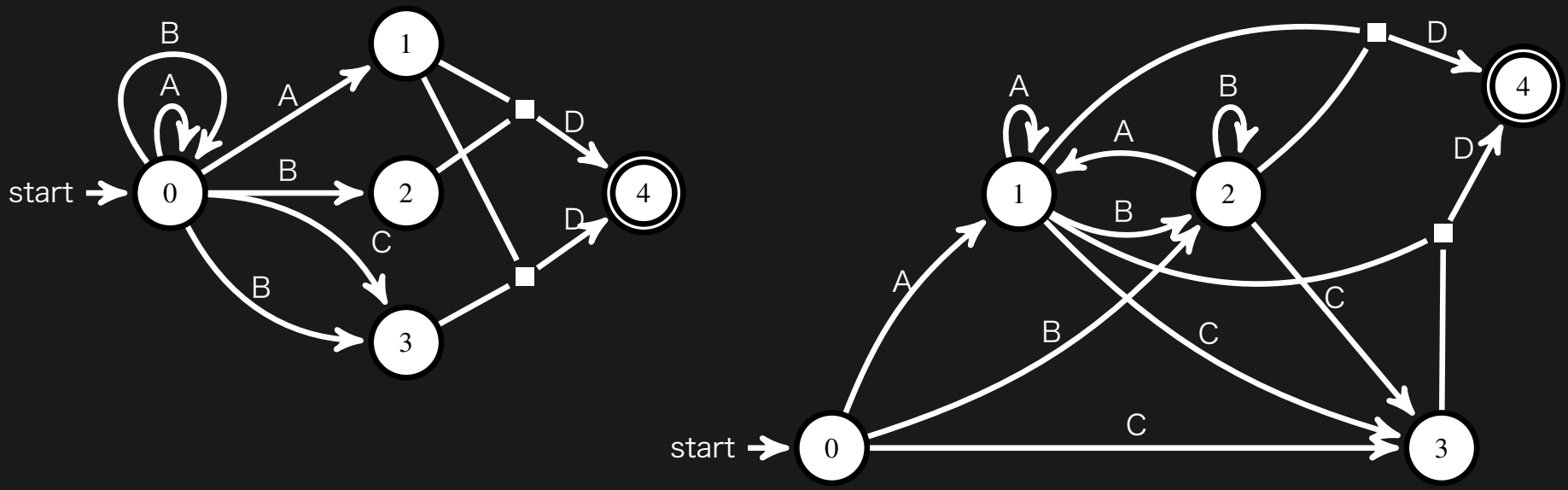




- オートマトンのサイズが増大する
  - 元のオートマトンの状態数nに対して, 最悪の場合2^n

決定性オートマトンの問題

- 重み付き木オートマトンの決定化演算 (閉路無し: May 06, 閉路有り: Büchse 09)



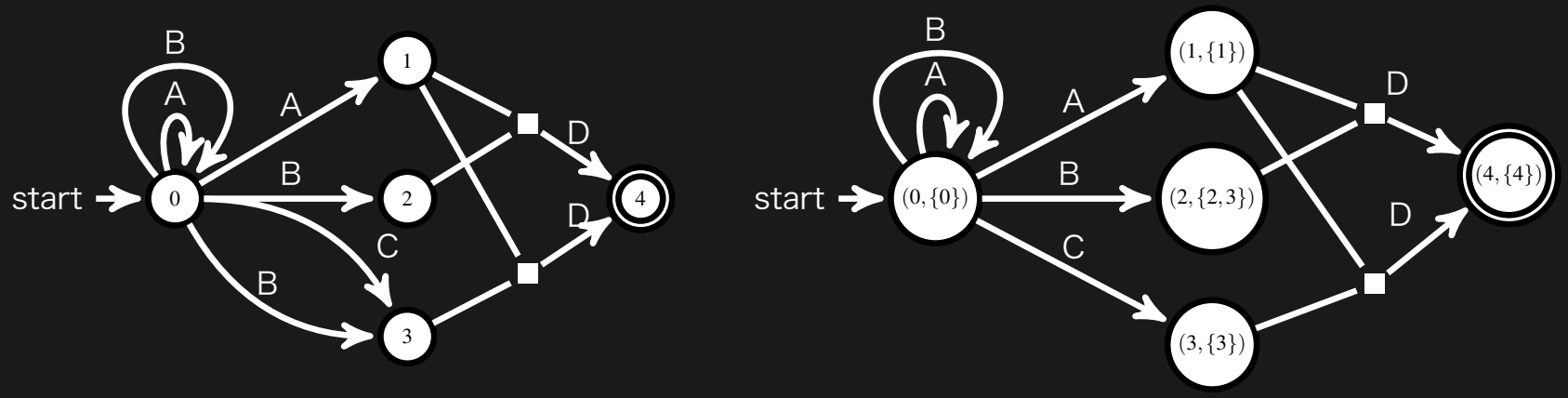
- オートマトンのサイズが増大する
  - 元のオートマトンの状態数nに対して, 最悪の場合2^n

無曖昧性の木オートマトンを直接構築する演算が必要

一般化曖昧性解消演算

重み付き木オートマトンへの一般化

木オートマトンに対する曖昧性解消演算

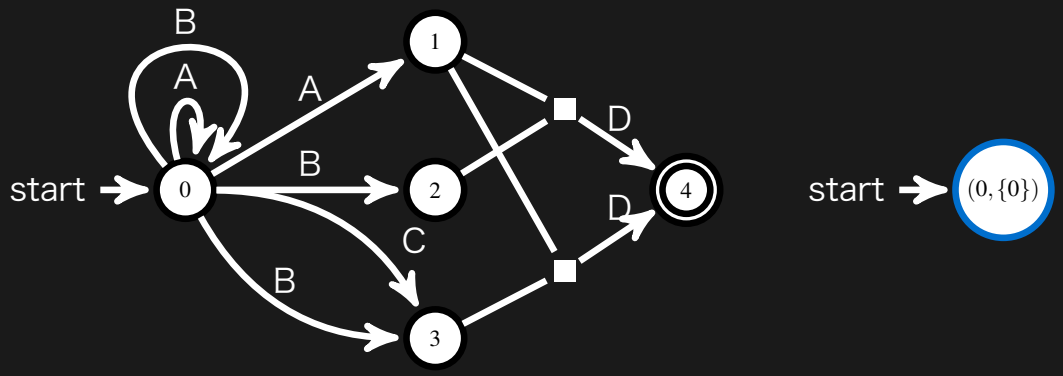


- 曖昧性解消演算の2つのキーポイント
  - 状態間の同値関係 $(key1)$
    - 初期状態から同じ木で到達可能
  - 2つの状態が将来を共有する $(key2)$
    - 同じ木で終了状態へ到達する経路を持つ $\Rightarrow trim(A \cap A)$でチェック
- 新しい状態$(p, \underline{s(p,t)})$: $p \in Q,\ t \in T_\Sigma$

$$\underline{s(p,t)} = \{q \mid q \in \underbrace{\delta(\{i\},t)}_{key1} \wedge \underbrace{(p,q) \in trim(A \cap A)}_{key2}\}$$

実行例 (1/7)

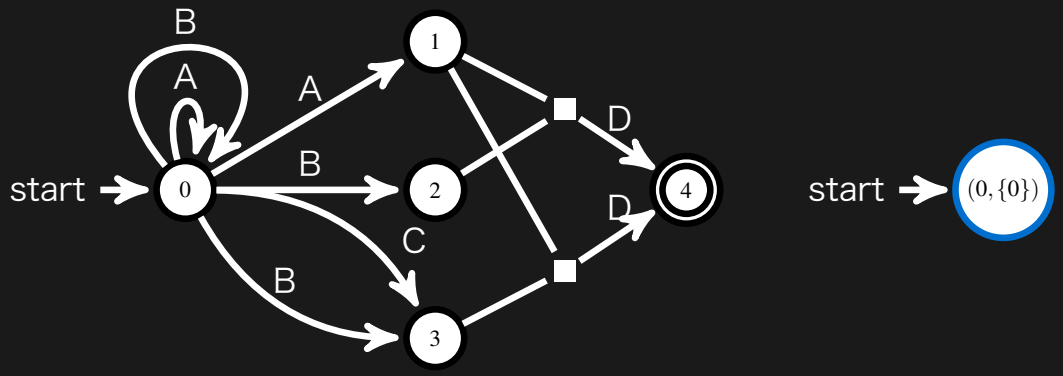
- 初期状態を構築



- 状態$(0,\underline{\{0\}})$: $\underline{s(0,\varepsilon)} = \{q | q \in \delta(\{\},\varepsilon) \wedge (0,q) \in trim(A \cap A)\} = \{0\}$

実行例 (1/7)

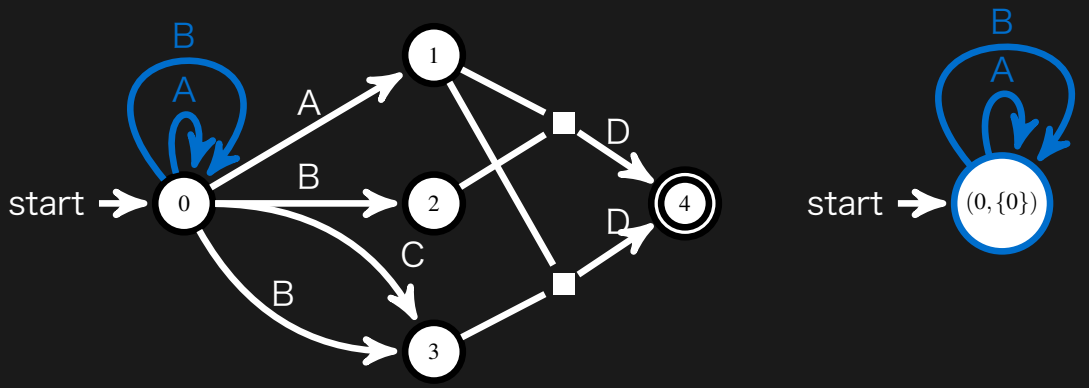
- 初期状態を構築



- 状態$(0,\underline{\{0\}})$: $\underline{s(0,\varepsilon)} = \{q | q \in \delta(\{\},\varepsilon) \wedge (0,q) \in trim(A \cap A)\} = \{0\}$
  * 同じ木で到達できる状態同士の同値関係Rを構築 $(0,\{0\})R(0,\{0\})$

実行例 (1/7)

- 遷移A$(0) \to 0$, B$(0) \to 0$に対する遷移を構築

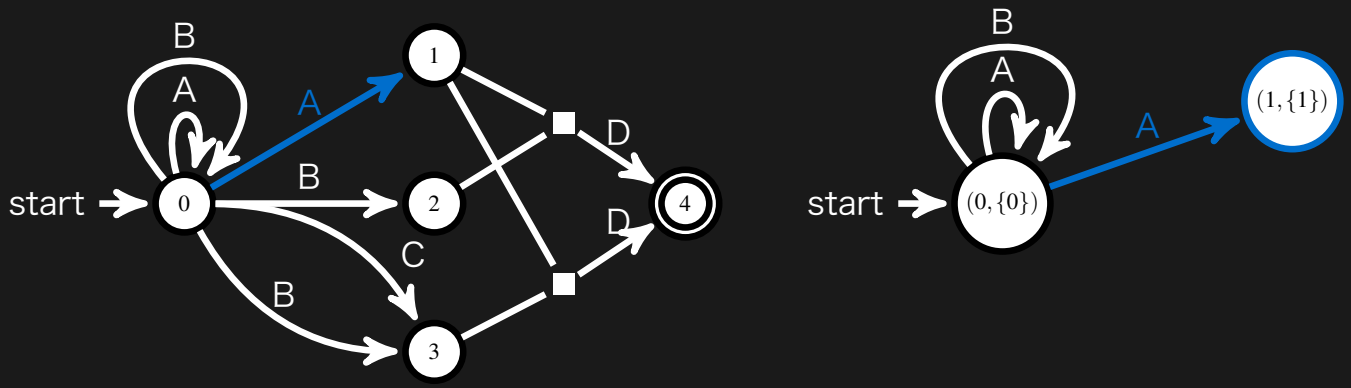


- 状態$(0, \underline{\{0\}})$: $\underline{s(0, \varepsilon)} = \{q | q \in \delta(\{\}, \varepsilon) \wedge (0, q) \in trim(A \cap A)\} = \{0\}$

  * 同じ木で到達できる状態同士の同値関係Rを構築 $(0, \{0\}) R (0, \{0\})$

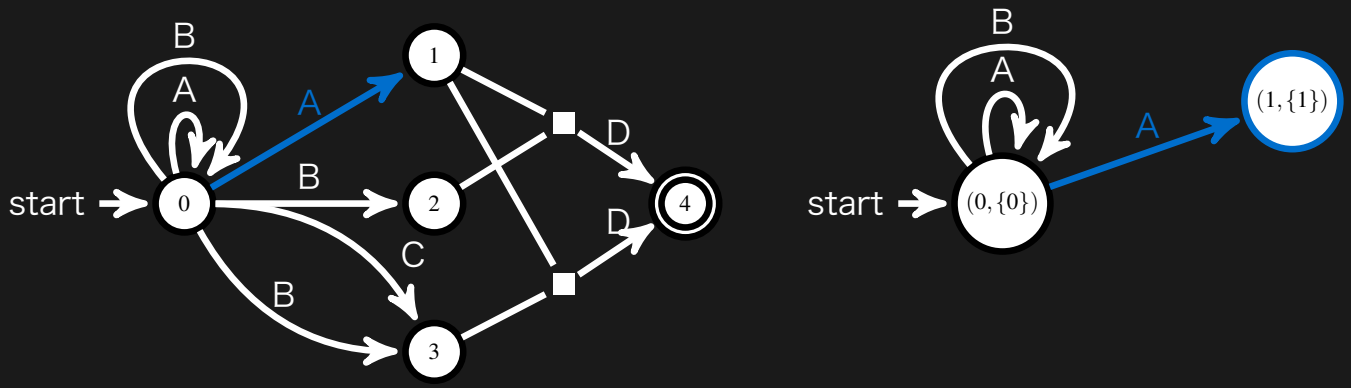
実行例 (2/7)

- 遷移A$(0) \to 1$に対する遷移と状態の構築



実行例 (2/7)

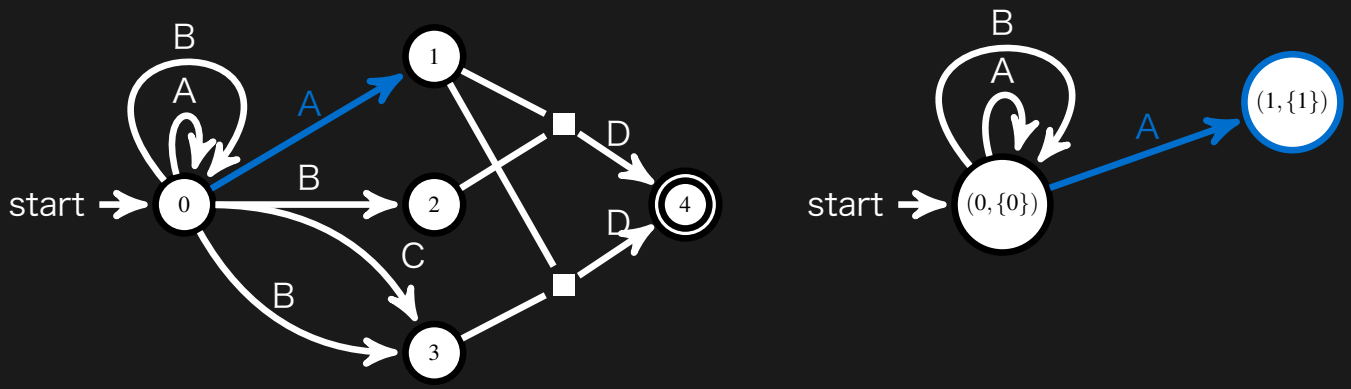
- 遷移A$(0) \rightarrow 1$に対する遷移と状態の構築



- 状態$(1, \underline{\{1\}})$: $\underline{s(1, \mathrm{A})} = \{q | q \in \delta(\{0\}, \mathrm{A}) \wedge (1, q) \in trim(A \cap A)\} = \{1\}$

実行例 (2/7)

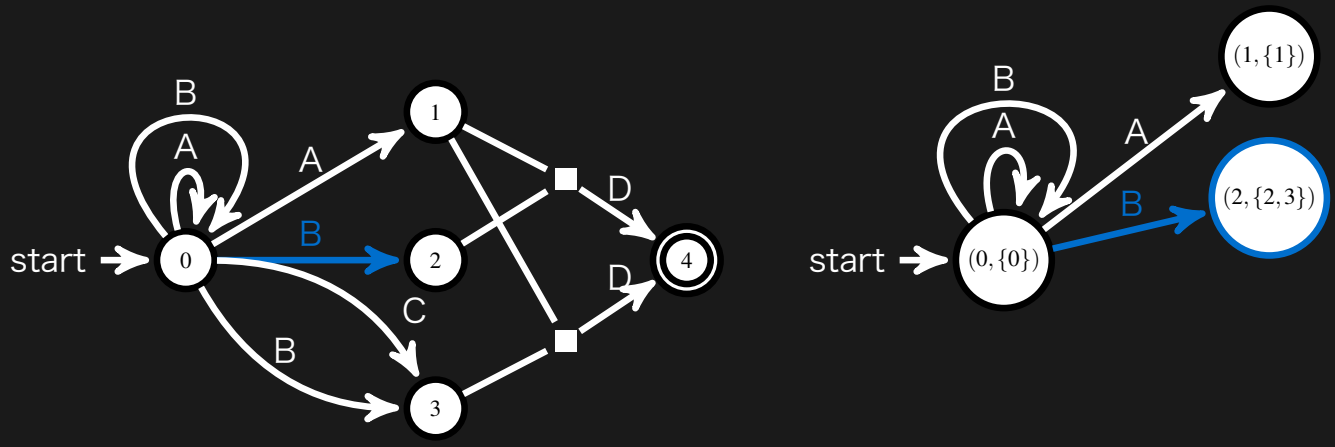
- 遷移A$(0) \to 1$に対する遷移と状態の構築



- 状態$(1, \underline{\{1\}})$: $\underline{s(1, \mathrm{A})} = \{q | q \in \delta(\{0\}, \mathrm{A}) \wedge (1, q) \in trim(A \cap A)\} = \{1\}$
- 関係: $(0, \{0\}) R (0, \{0\}), (1, \{1\}) R (1, \{1\})$

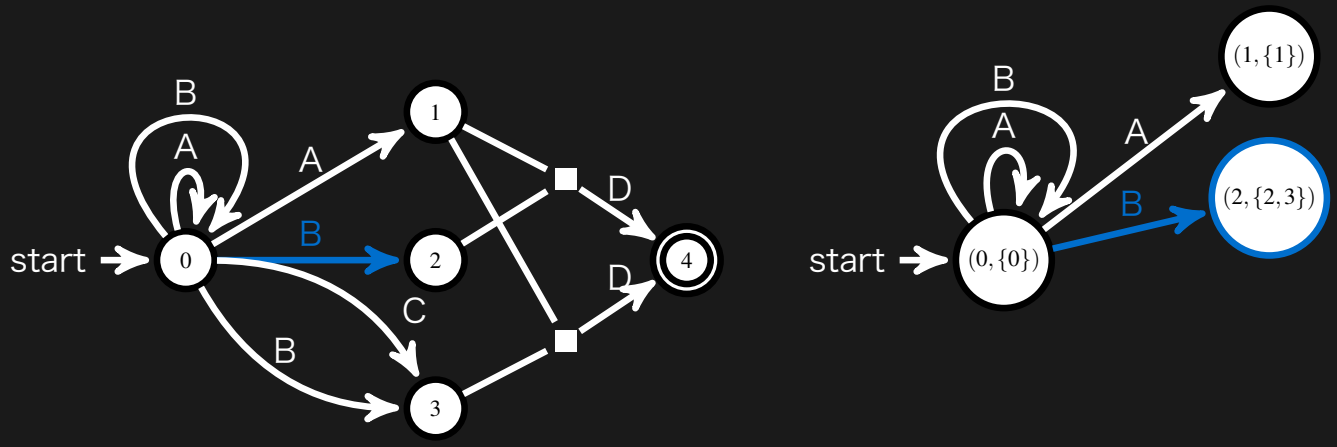
実行例 (3/7)

- 遷移B$(0) \to 2$に対する遷移と状態の構築



実行例 (3/7)

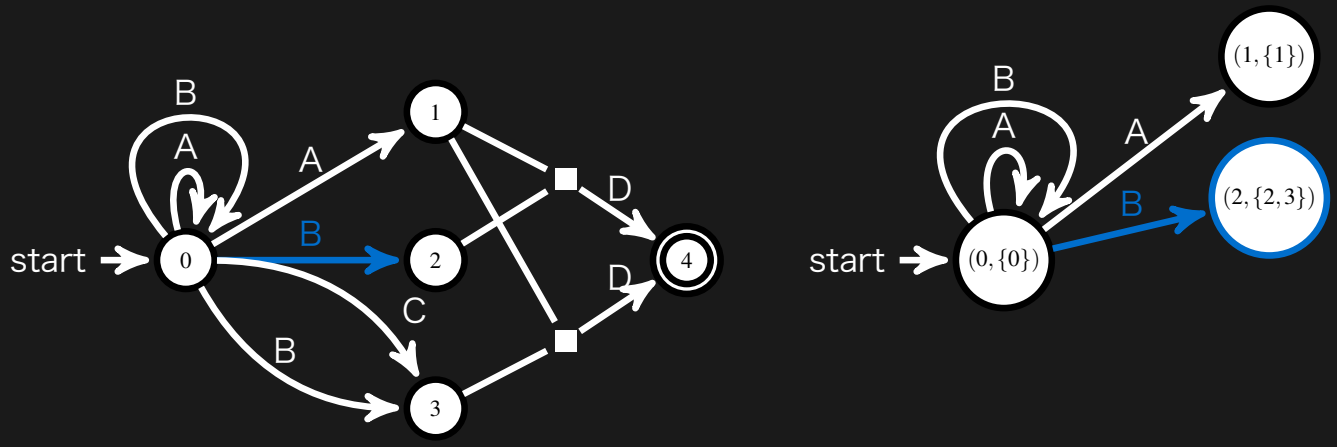
- 遷移B$(0) \rightarrow 2$に対する遷移と状態の構築



- 状態$(2,\underline{\{2,3\}})$: $\underline{s(2,\mathrm{B})} = \{q | q \in \delta(\{0\},\mathrm{B}) \wedge (2,q) \in trim(A \cap A)\} = \{2,3\}$

実行例 (3/7)

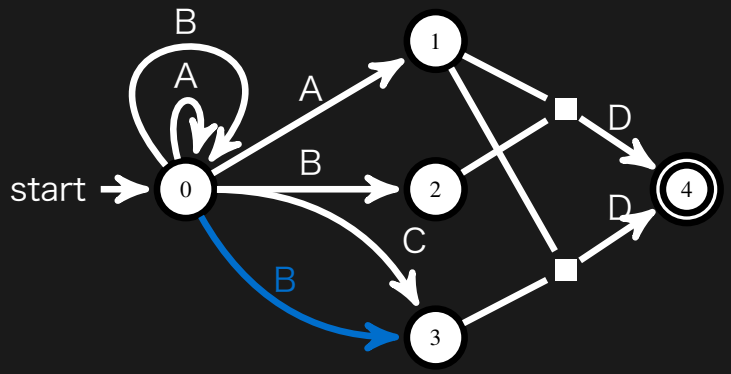
- 遷移B$(0) \rightarrow 2$に対する遷移と状態の構築

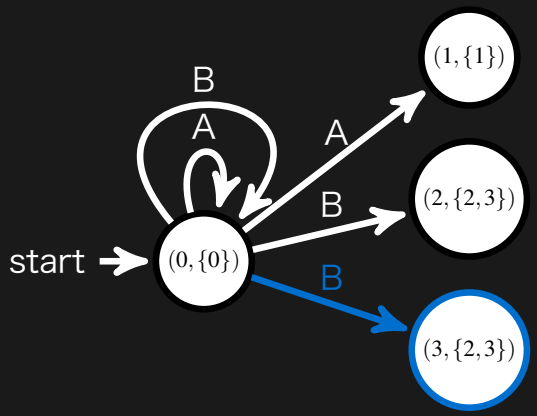


- 状態$(2,\underline{\{2,3\}})$: $\underline{s(2,\mathrm{B})} = \{q | q \in \delta(\{0\},\mathrm{B}) \wedge (2,q) \in trim(A \cap A)\} = \{2,3\}$
- 関係: $(0,\{0\})R(0,\{0\}), (1,\{1\})R(1,\{1\}), (2,\{2,3\})R(2,\{2,3\})$

実行例 (4/7)

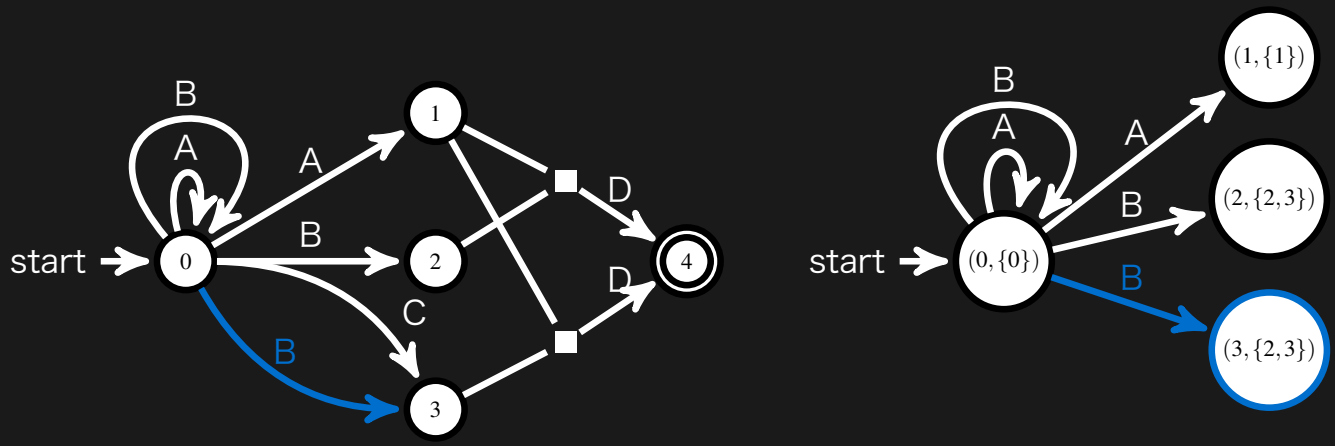
- 遷移B$(0) \rightarrow 3$に対する遷移と状態の構築





実行例 (4/7)

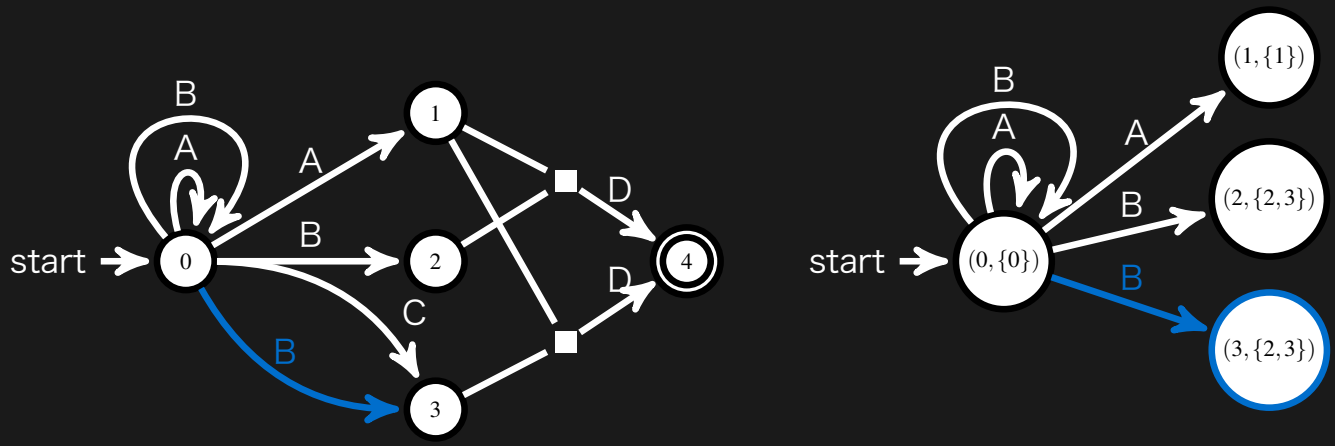
- 遷移B$(0) \rightarrow 3$に対する遷移と状態の構築



- 状態$(3,\underline{\{2,3\}})$: $\underline{s(3,\mathrm{B})} = \{q | q \in \delta(\{0\},\mathrm{B}) \wedge (3,q) \in trim(A \cap A)\} = \{2,3\}$

実行例 (4/7)

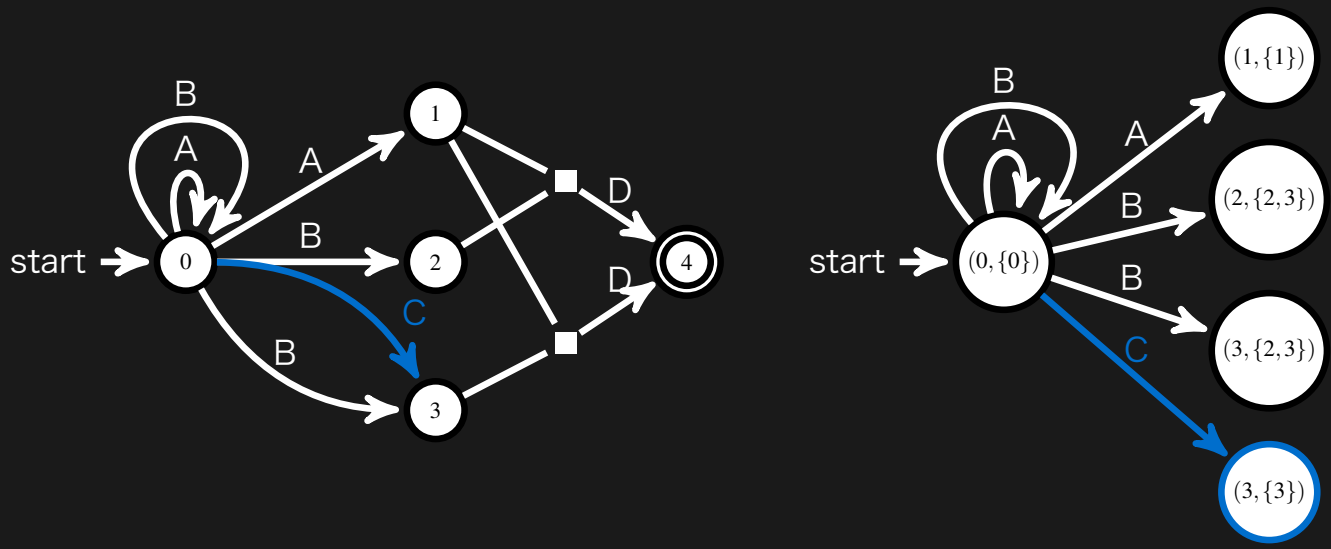
- 遷移B$(0) \rightarrow 3$に対する遷移と状態の構築



- 状態$(3,\underline{\{2,3\}})$: $\underline{s(3,\mathrm{B})} = \{q | q \in \delta(\{0\},\mathrm{B}) \wedge (3,q) \in trim(A \cap A)\} = \{2,3\}$
- 関係: $(0,\{0\})R(0,\{0\}), (1,\{1\})R(1,\{1\}), (2,\{2,3\})R(2,\{2,3\}),$
  $(3,\{2,3\})R(3,\{2,3\}), (2,\{2,3\})R(3,\{2,3\})$

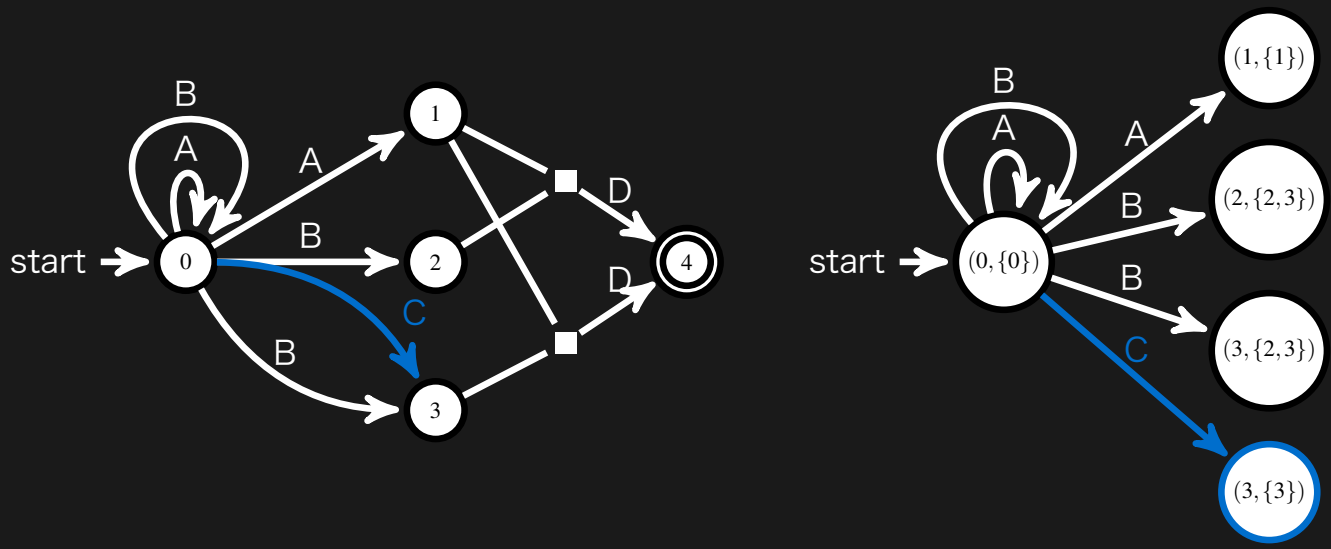
実行例 (5/7)

- 遷移C$(0) \to 3$に対する遷移と状態の構築



実行例 (5/7)

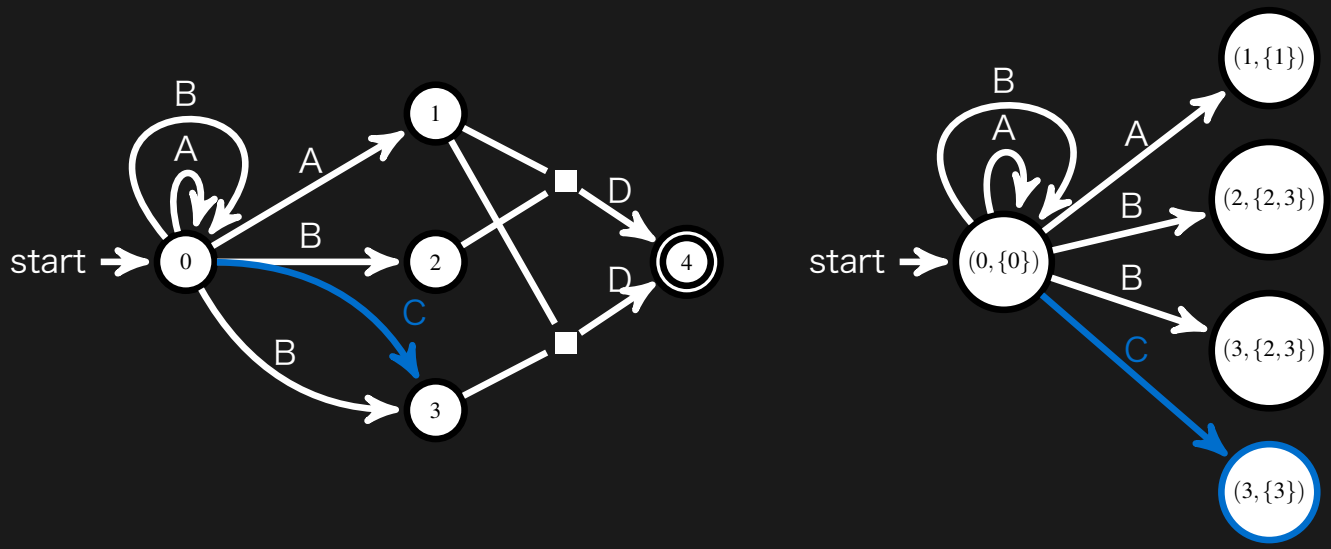
- 遷移C$(0) \rightarrow 3$に対する遷移と状態の構築



- 状態$(3,\underline{\{3\}})$: $\underline{s(3,\mathsf{C})} = \{q | q \in \delta(\{0\},\mathsf{C}) \wedge (3,q) \in trim(A \cap A)\} = \{3\}$

実行例 (5/7)

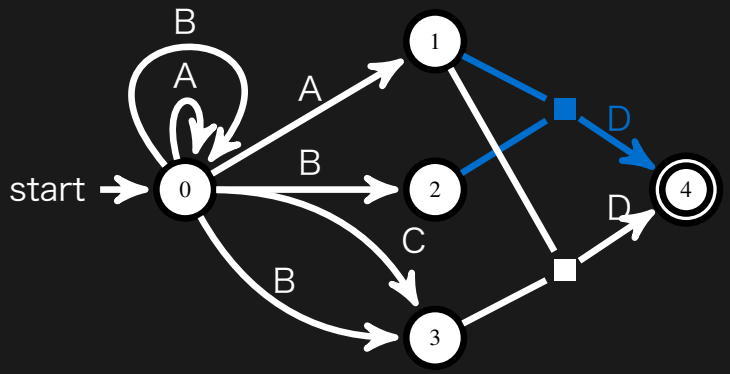
- 遷移C$(0) \rightarrow 3$に対する遷移と状態の構築

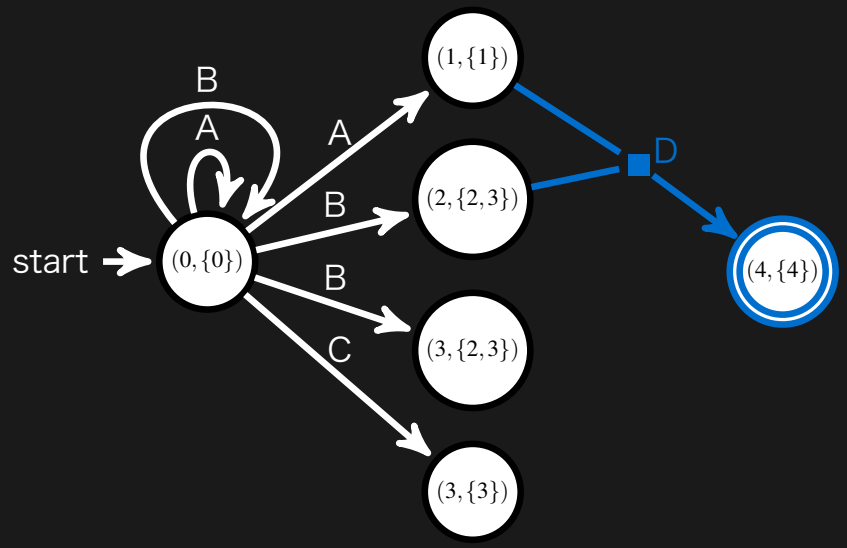


- 状態$(3,\underline{\{3\}})$: $\underline{s(3,\mathsf{C})} = \{q | q \in \delta(\{0\},\mathsf{C}) \wedge (3,q) \in trim(A \cap A)\} = \{3\}$
- 関係: $(0,\{0\})R(0,\{0\}), (1,\{1\})R(1,\{1\}), (2,\{2,3\})R(2,\{2,3\}),$
$(3,\{2,3\})R(3,\{2,3\}), (2,\{2,3\})R(3,\{2,3\}), (3,\{3\})R(3,\{3\})$

実行例 (6/7)

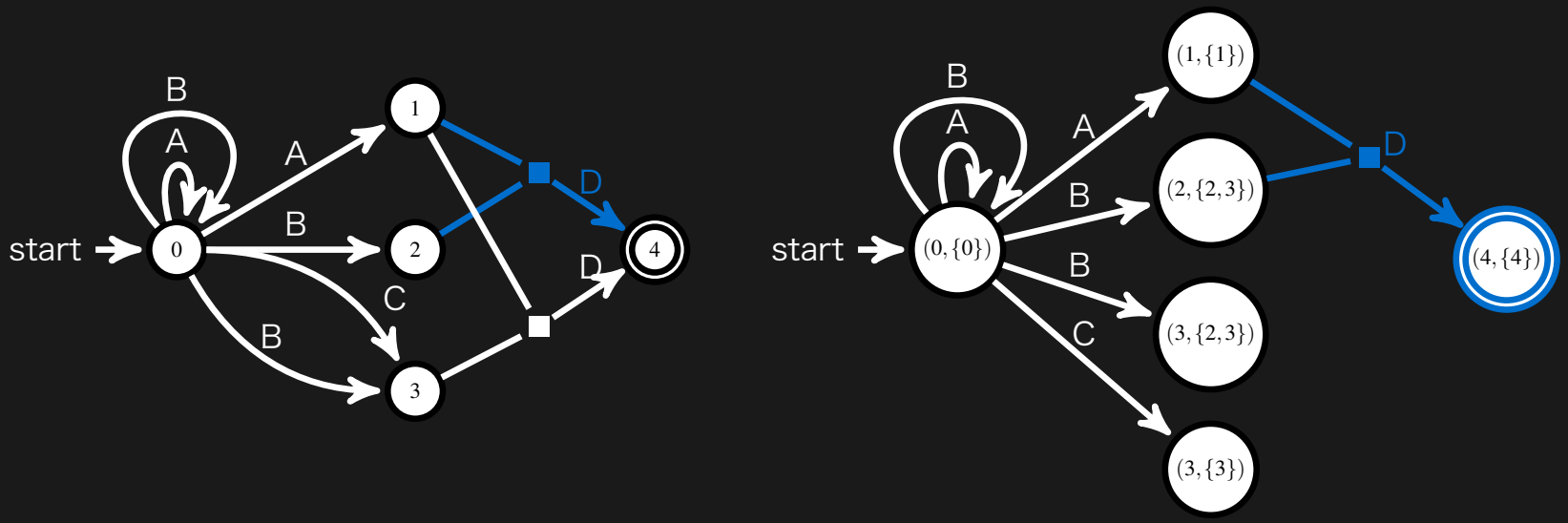
- 遷移D$(1,2) \to 4$に対する遷移と状態の構築





実行例 (6/7)

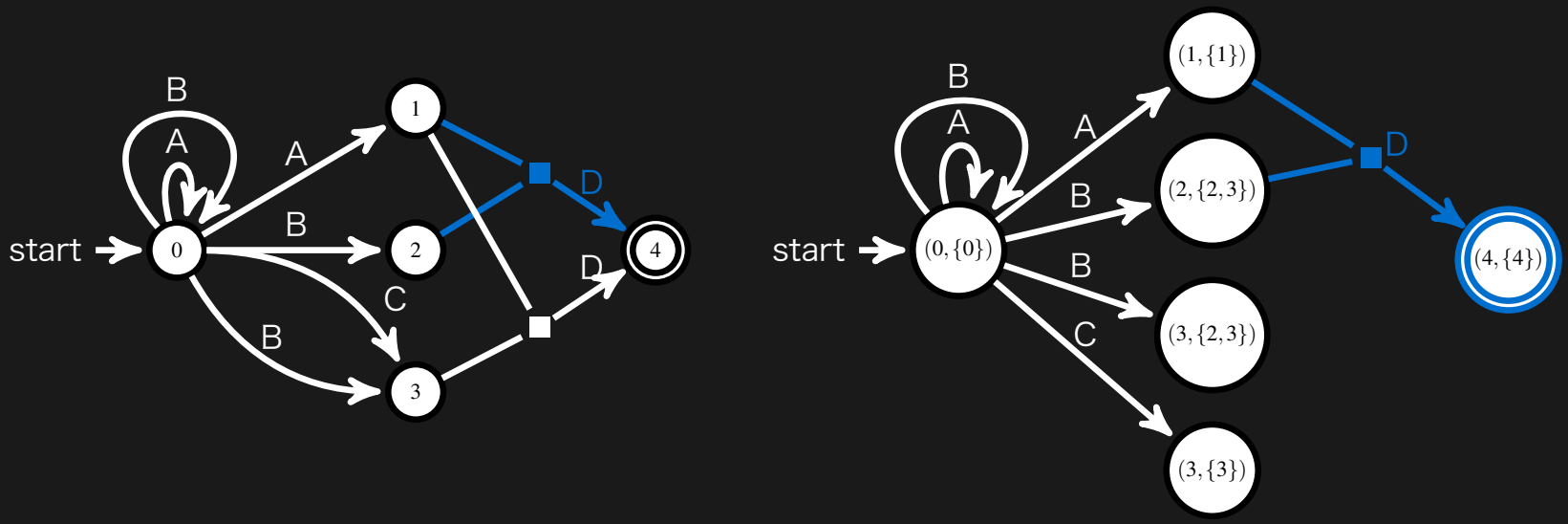
- 遷移D$(1,2) \to 4$に対する遷移と状態の構築



- 状態$(4,\{4\})$:
  $\underline{s(4,\mathsf{D}(\overline{\mathsf{A}\ \mathsf{B}}))} = \{q | q \in \delta(\{\{1,2\},\{1,3\}\},\mathsf{D}) \wedge (4,q) \in trim(A \cap A)\} = \{4\}$

実行例 (6/7)

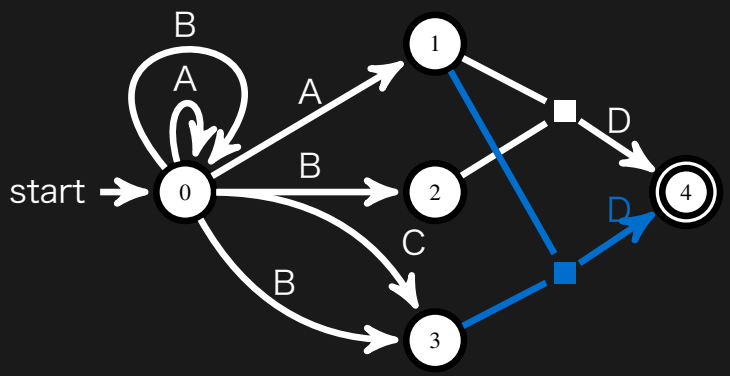
- 遷移D$(1,2) \rightarrow 4$に対する遷移と状態の構築

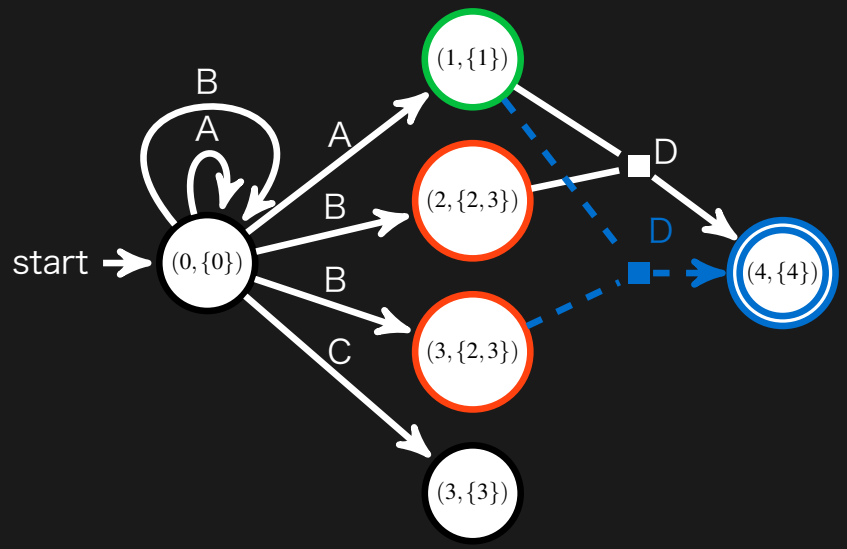


- 状態$(4,\{4\})$:
  $s(4,\mathrm{D}(\overline{\mathrm{A}\ \mathrm{B}})) = \{q | q \in \delta(\{\{1,2\},\{1,3\}\},\mathrm{D}) \wedge (4,q) \in trim(A \cap A)\} = \{4\}$
- 関係: $(0,\{0\})R(0,\{0\}), (1,\{1\})R(1,\{1\}), (2,\{2,3\})R(2,\{2,3\}),$
  $(3,\{2,3\})R(3,\{2,3\}), (2,\{2,3\})R(3,\{2,3\}), (3,\{3\})R(3,\{3\}),$
  $(4,\{4\})R(4,\{4\})$

実行例 (7/7)

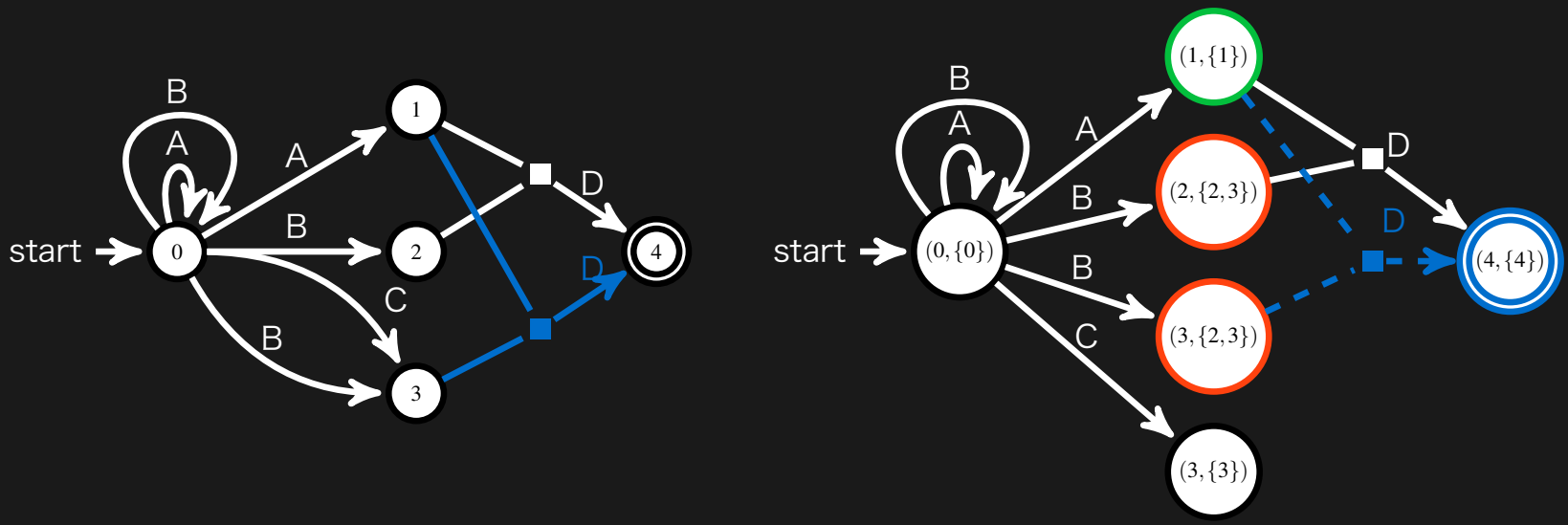
- 遷移D$(1,3) \to 4$に対する遷移と状態の構築





実行例 (7/7)

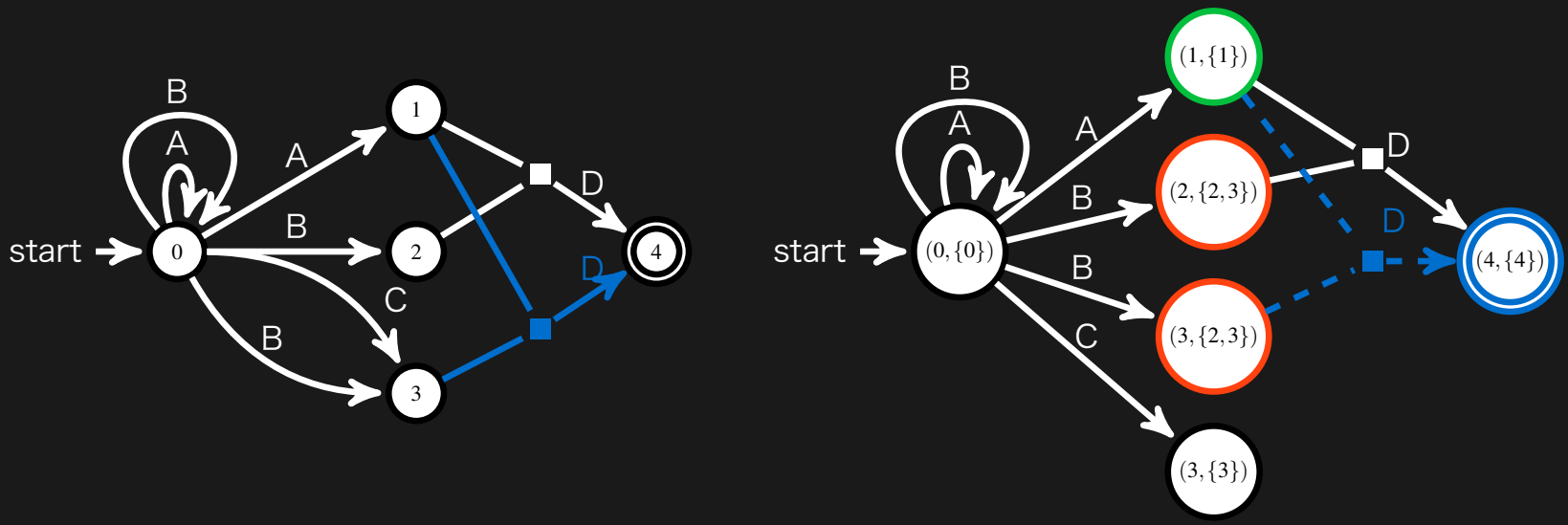
- 遷移D$(1,3) \to 4$に対する遷移と状態の構築



- 状態$(4,\{4\})$:
  $s(4,\mathsf{D}(\overline{\mathsf{A}\ \mathsf{B}})) = \{q | q \in \delta(\{\{1,2\},\{1,3\}\},\mathsf{D}) \wedge (4,q) \in trim(A \cap A)\} = \{4\}$

実行例 (7/7)

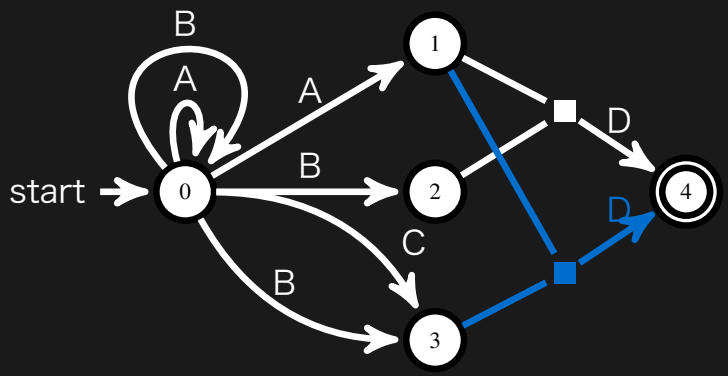
- 遷移D$(1,3) \to 4$に対する遷移と状態の構築

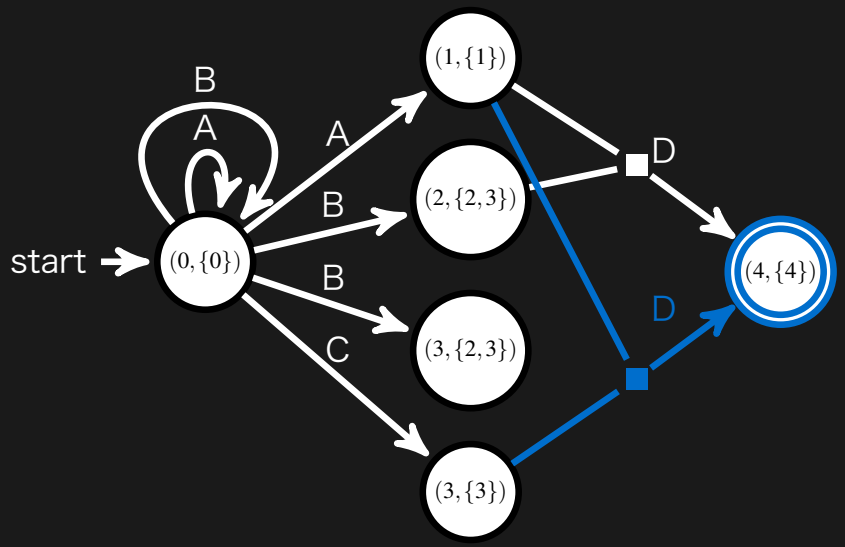


- 状態$(4,\{4\})$:
  $s(4,\mathrm{D}(\overline{\mathrm{A}\ \mathrm{B}})) = \{q | q \in \delta(\{\{1,2\},\{1,3\}\},\mathrm{D}) \wedge (4,q) \in trim(A \cap A)\} = \{4\}$
- 関係$(1,\{1\})R(1,\{1\}), (2,\{2,3\})R(3,\{2,3\})$を持つ状態から$(4,\{4\})$への遷移はすでに存在するので，この遷移は構築されない

実行例 (7/7)

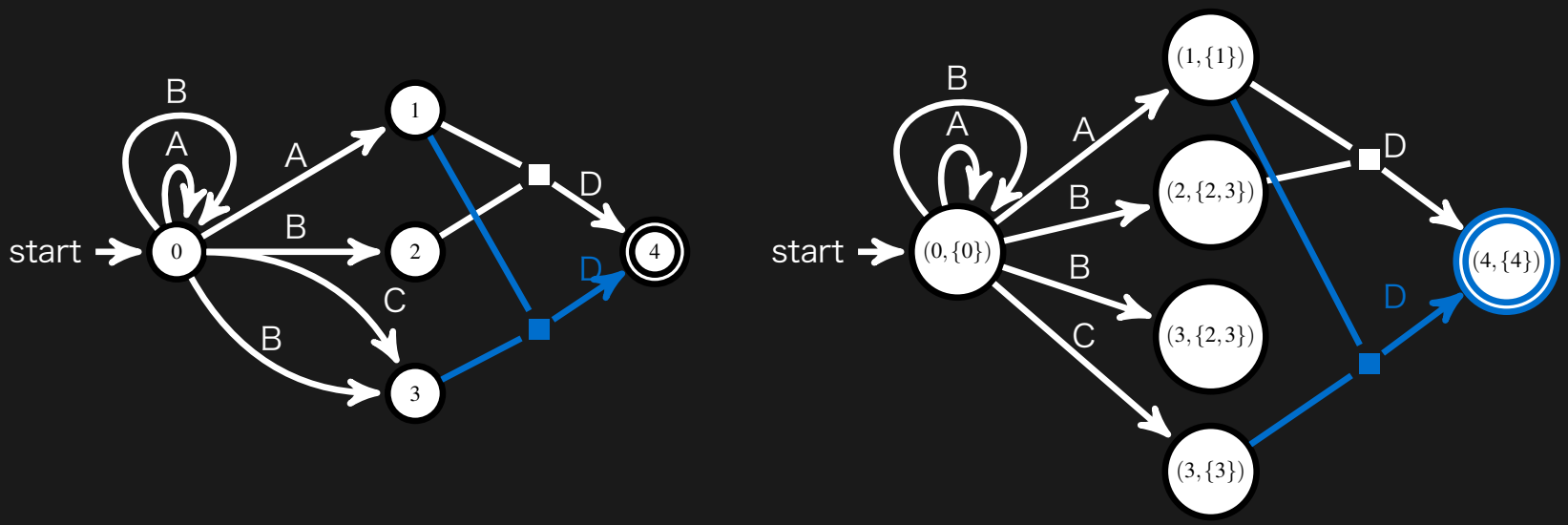
- 遷移D$(1,3) \to 4$に対する遷移と状態の構築





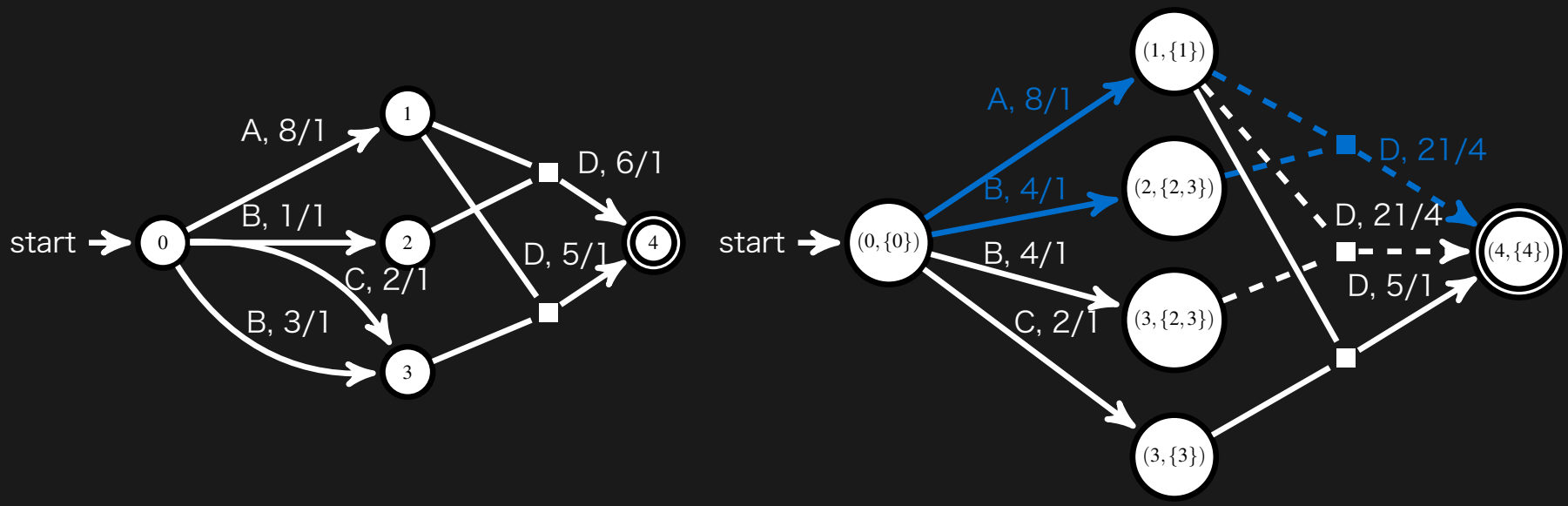
実行例 (7/7)

- 遷移D$(1,3) \rightarrow 4$に対する遷移と状態の構築



- 状態$(4,\{4\})$:
  $\underline{s(4,\mathsf{D}(\overline{\mathsf{A}\ \mathsf{C}}))} = \{q | q \in \delta(\{\{1,2\},\{1,3\}\},\mathsf{D}) \wedge (4,q) \in trim(A \cap A)\} = \{4\}$

重み付きの場合の結果



♪ 同じ木に対応する全受理経路の重みの合計

例えば，$w($  $) = 48 + 120 = 168$

提案法のいくつかの重要な性質

- 一般化曖昧性解消演算の正当性
  - 木オートマトンAに対する曖昧性解消演算の結果をA'とする.
    そのとき, A'は無曖昧で, かつ, $L(A) = L(A')$である.

提案法のいくつかの重要な性質

- 一般化曖昧性解消演算の正当性
  - 木オートマトンAに対する曖昧性解消演算の結果をA'とする.
    そのとき, A'は無曖昧で, かつ, $L(A) = L(A')$である.
- 重み計算の正当性
  - Aによって受理可能な木tに対して, その経路の集合を$Run(\{i\}, t, F)$と書く.
    また, A'で木に対する受理経路をp'とするとき,

$$w(p') = \sum_{p \in Run(\{i\}, t, F)} w(p).$$

提案法のいくつかの重要な性質

- 一般化曖昧性解消演算の正当性
  - 木オートマトンAに対する曖昧性解消演算の結果をA'とする.
    そのとき, A'は無曖昧で, かつ, $L(A) = L(A')$である.
- 重み計算の正当性
  - Aによって受理可能な木tに対して, その経路の集合を$Run(\{i\}, t, F)$と書く.
    また, A'で木に対する受理経路をp'とするとき,

$$w(p') = \sum_{p \in Run(\{i\}, t, F)} w(p).$$

- 適用十分条件 (閉路を含む場合)
  - トロピカル半環$(R_+^{+\infty}, \min, +, +\infty, 0)$上で定義された$A$が
    弱双子定理 (weak twins property) を満たすとき, 演算は適用可能
    - 決定化は双子定理 (twins property)を満たすときに
      適用可能 (Büchse 09) $\Rightarrow$ 決定化より適用範囲が広い

実験

自然言語処理タスクへの適用

実験設定

- 自然言語処理の文圧縮 (Cohn 09)と機械翻訳 (Graehl 08)
  - 拡張型木トランスデューサ (Graehl 08)でモデル化
    - 入力された文の木構造を別の木構造へと変換
    - 出力空間は重み付き木オートマトンで表現
  - 出力空間を決定化, 無曖昧化したときのオートマトンのサイズ ($|A| = |Q| + |E|$)を比較

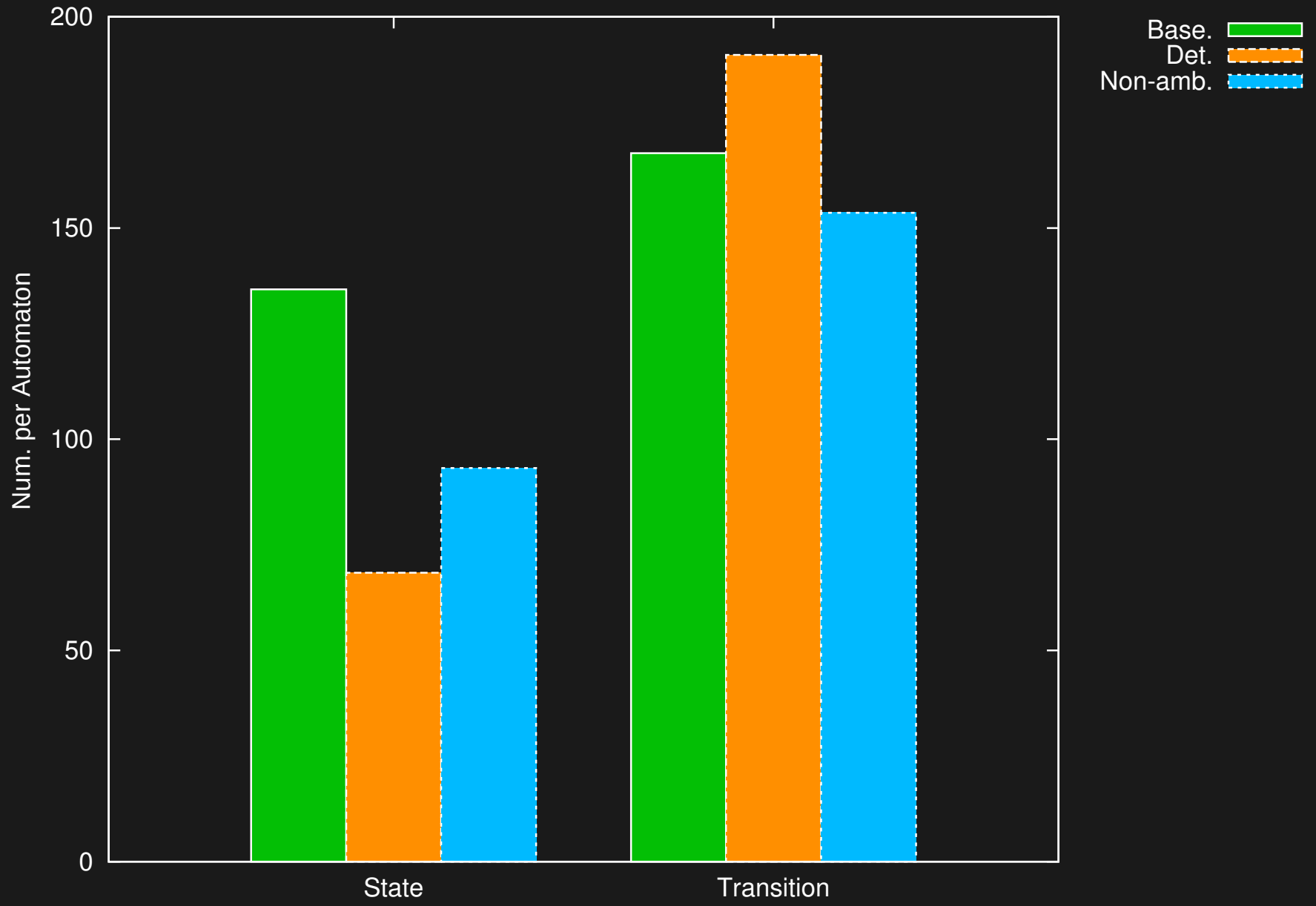
実験設定

- 自然言語処理の文圧縮 (Cohn 09)と機械翻訳 (Graehl 08)
  - 拡張型木トランスデューサ (Graehl 08)でモデル化
    - 入力された文の木構造を別の木構造へと変換
    - 出力空間は重み付き木オートマトンで表現
  - 出力空間を決定化, 無曖昧化したときのオートマトンのサイズ ($|A| = |Q| + |E|$)を比較
- 実験データ, ツール

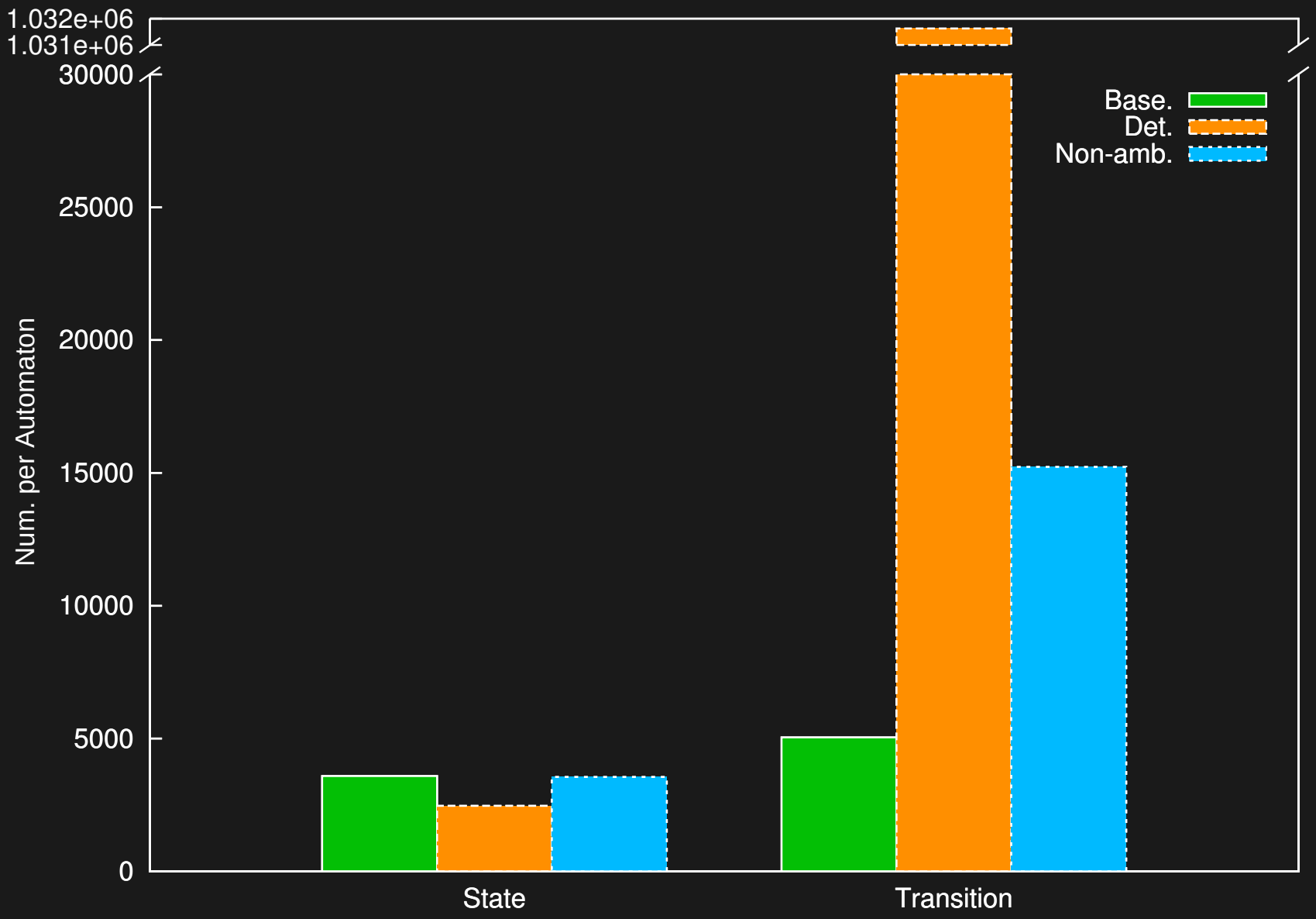
		文圧縮 (Cohn 08)	機械翻訳 (Goto 13)
データ	訓練サイズ	480文対	5,000文対
	テストサイズ	††480文対	278文対
ツール	木トランスデューサ	Tiburon	Tiburon
	規則抽出	T3	T3
	構文解析器	内製システム	内製システム
	単語アライメント	GIZA++	GIZA++

††データ量が少ないため, 訓練データと同じデータを使用

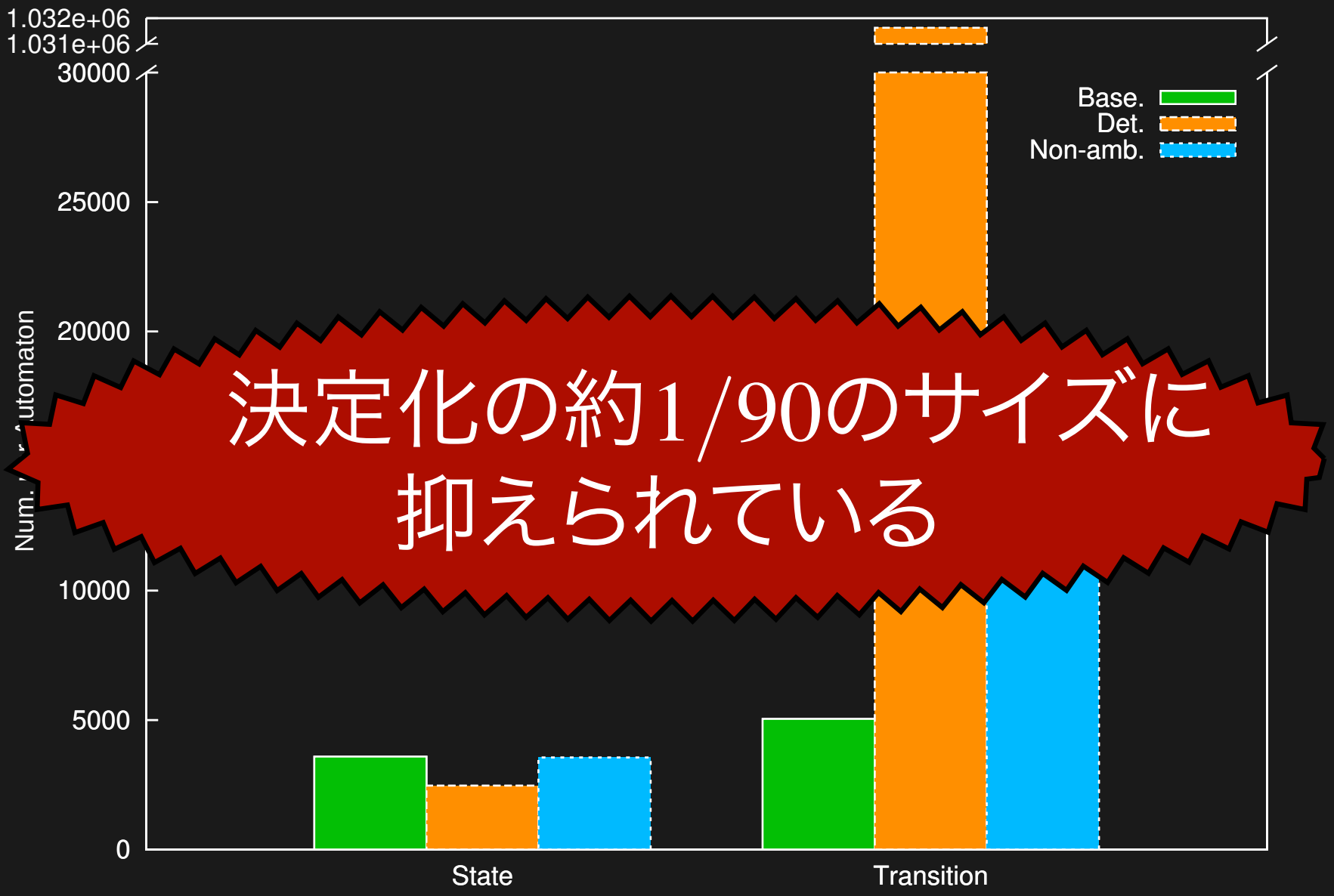
実験結果: 文圧縮タスク


200
150
100
50
0
Num. per Automaton
State
Transition
Base.
Det.
Non-amb.

実験結果: 機械翻訳タスク

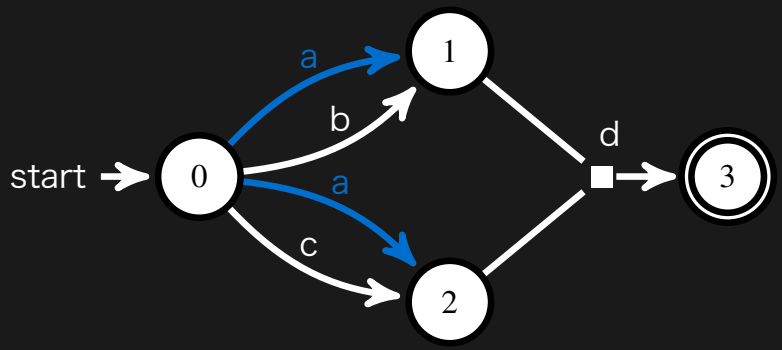


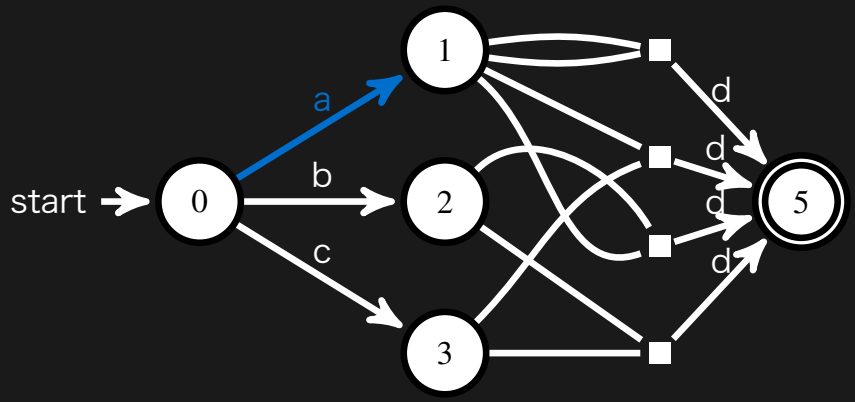
実験結果: 機械翻訳タスク



実験結果の分析

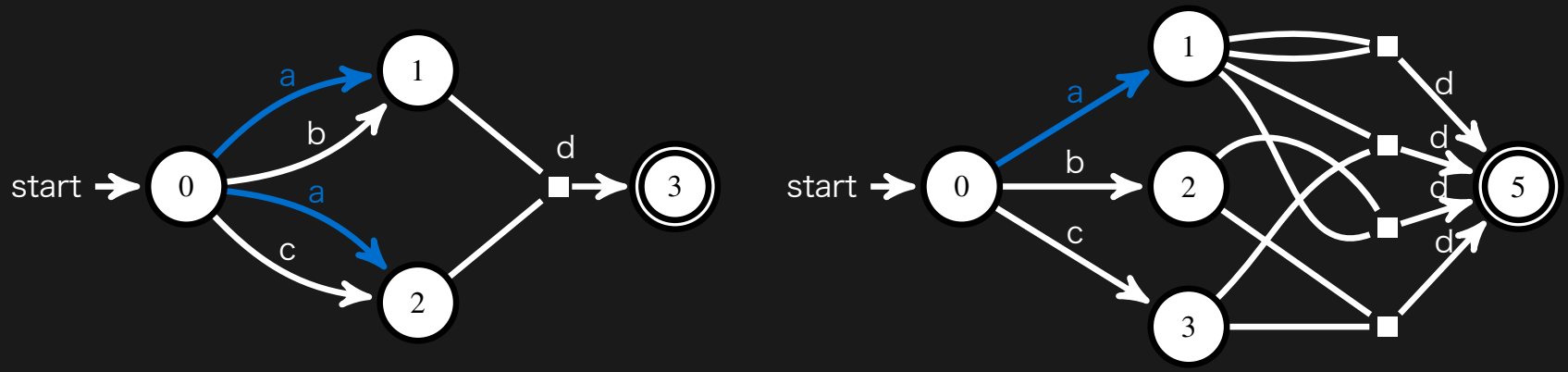
- なぜ, 機械翻訳タスクで決定化の結果は遷移数が増大したか?



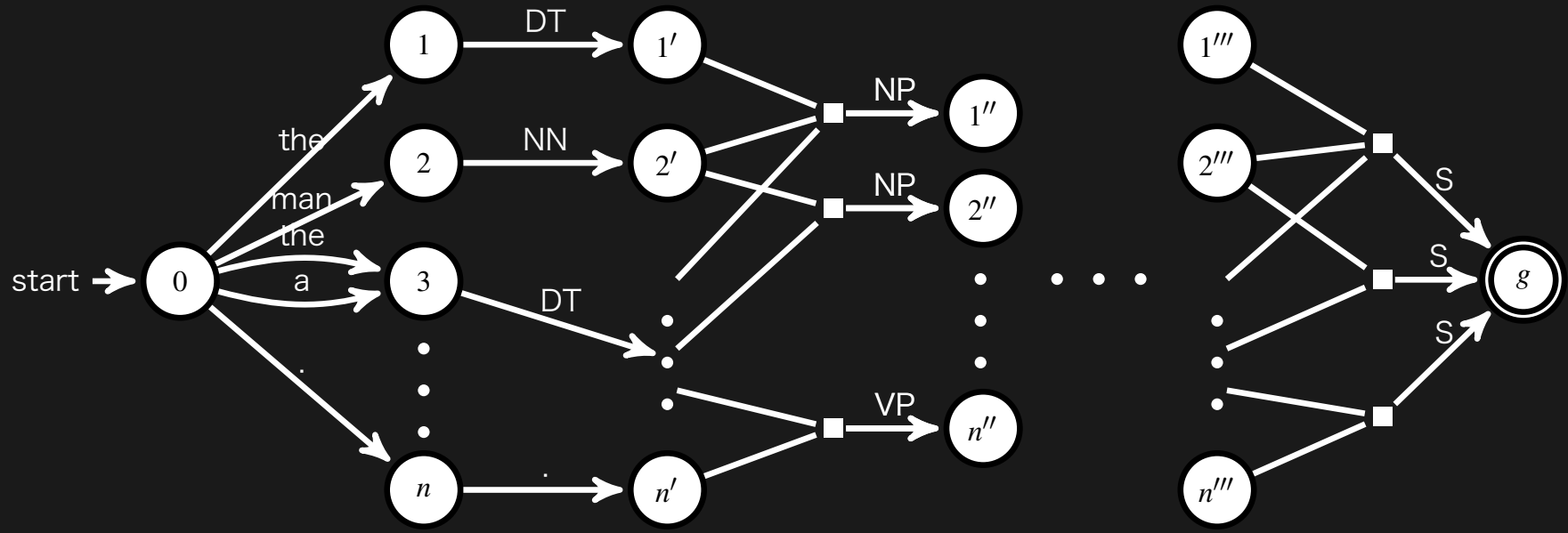


実験結果の分析

- なぜ, 機械翻訳タスクで決定化の結果は遷移数が増大したか?

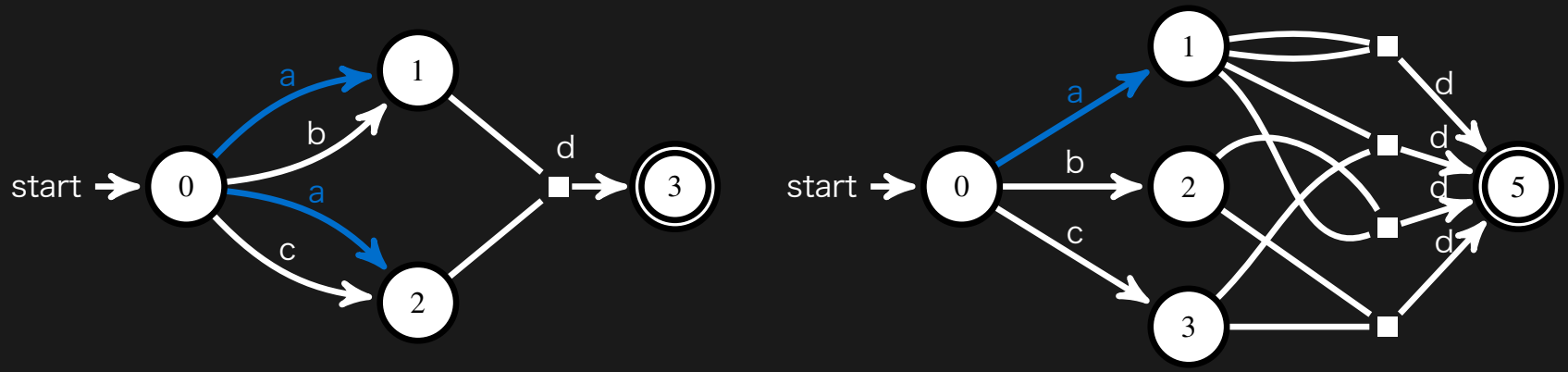


- 機械翻訳は語彙が多様かつ膨大

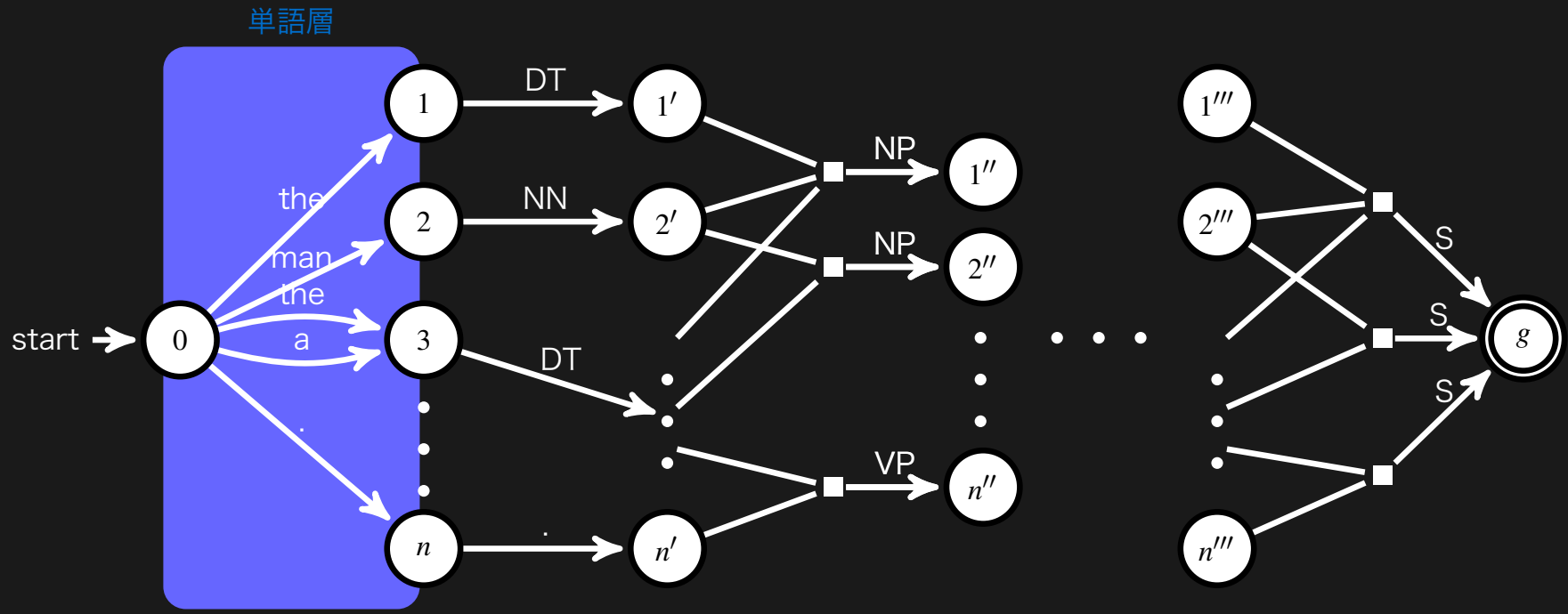


実験結果の分析

- なぜ, 機械翻訳タスクで決定化の結果は遷移数が増大したか?

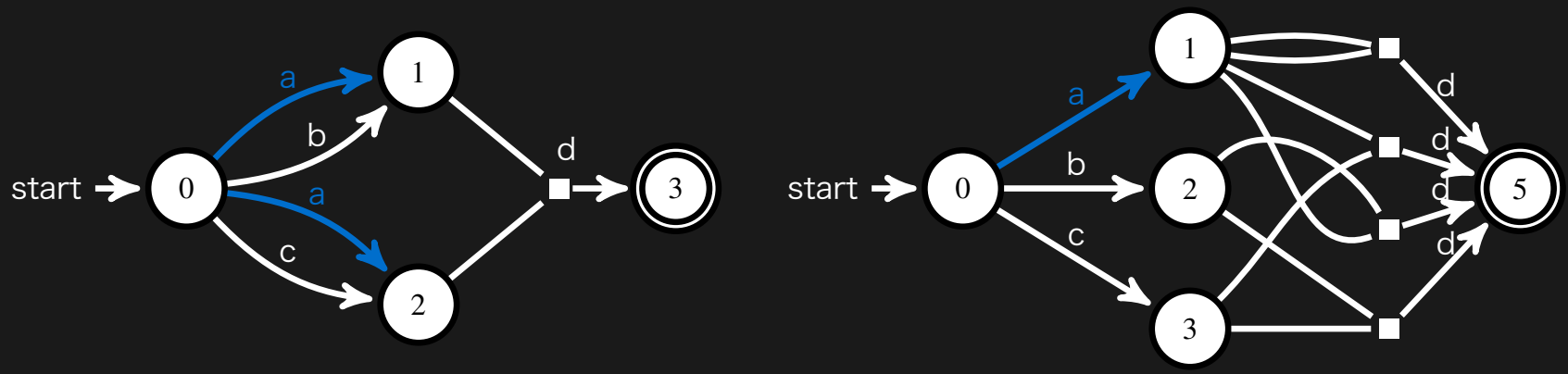


- 機械翻訳は語彙が多様かつ膨大

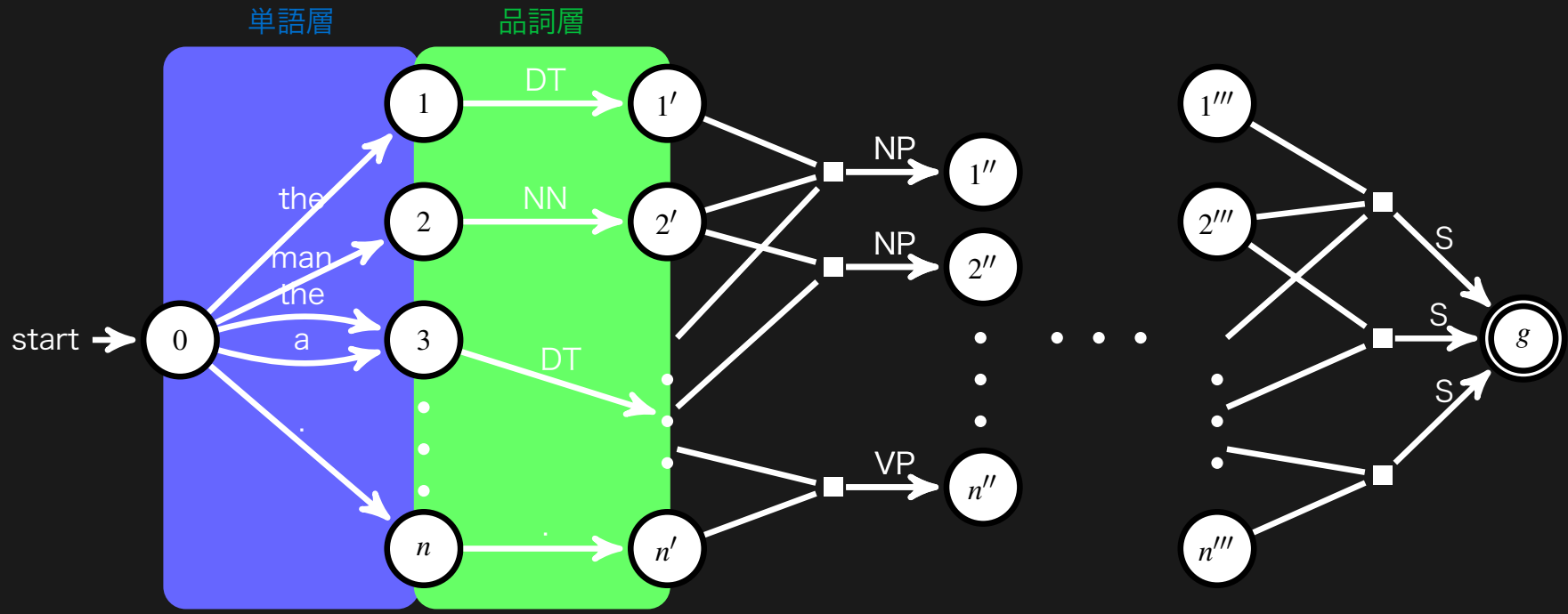


実験結果の分析

- なぜ, 機械翻訳タスクで決定化の結果は遷移数が増大したか?

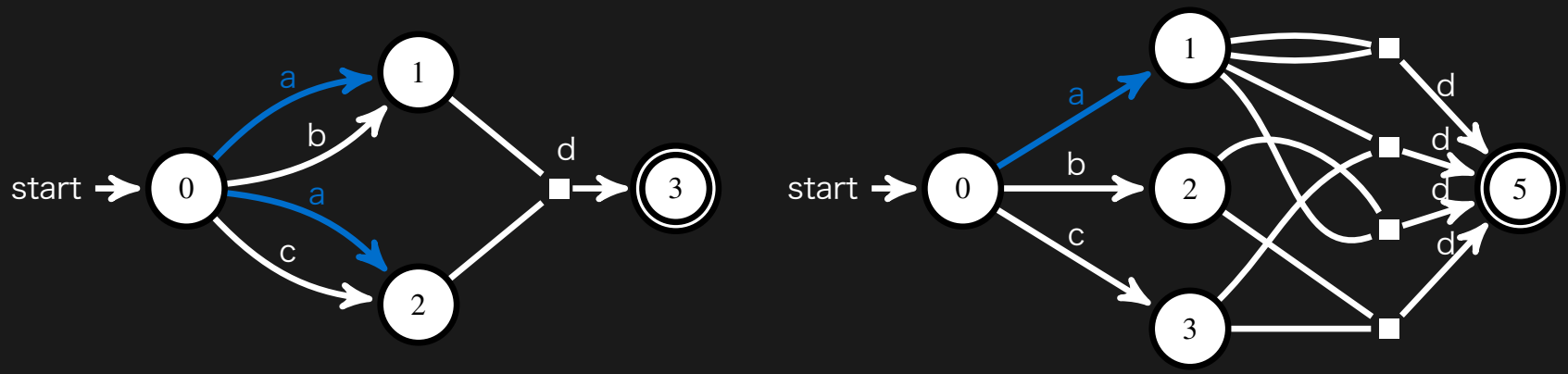


- 機械翻訳は語彙が多様かつ膨大

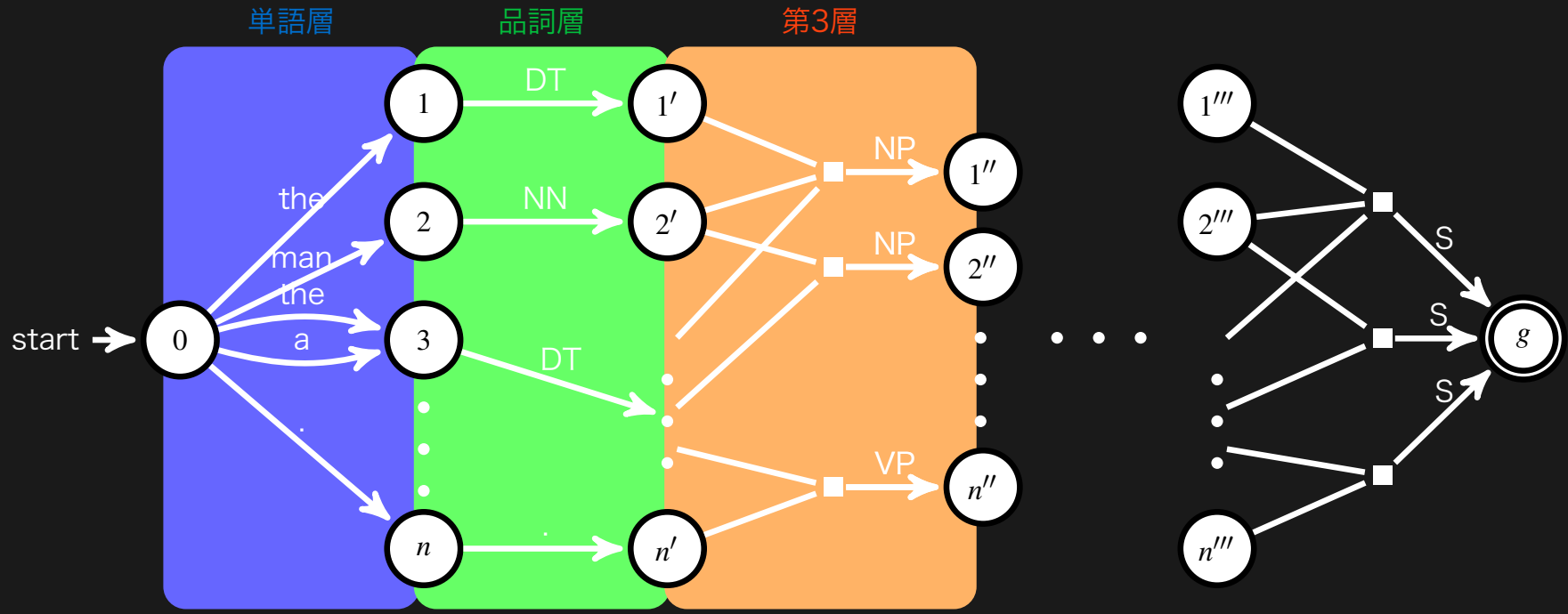


実験結果の分析

- なぜ, 機械翻訳タスクで決定化の結果は遷移数が増大したか?

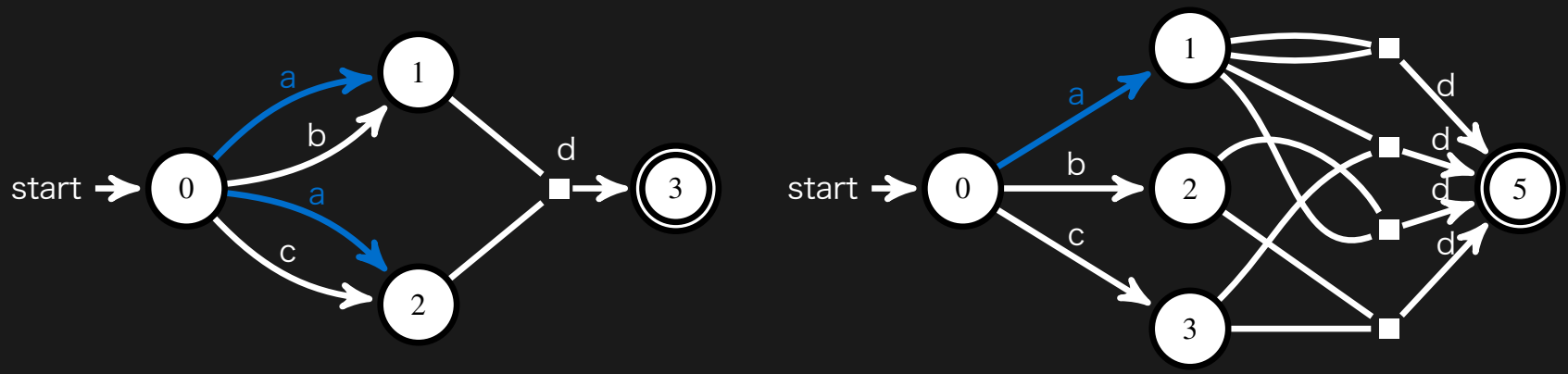


- 機械翻訳は語彙が多様かつ膨大

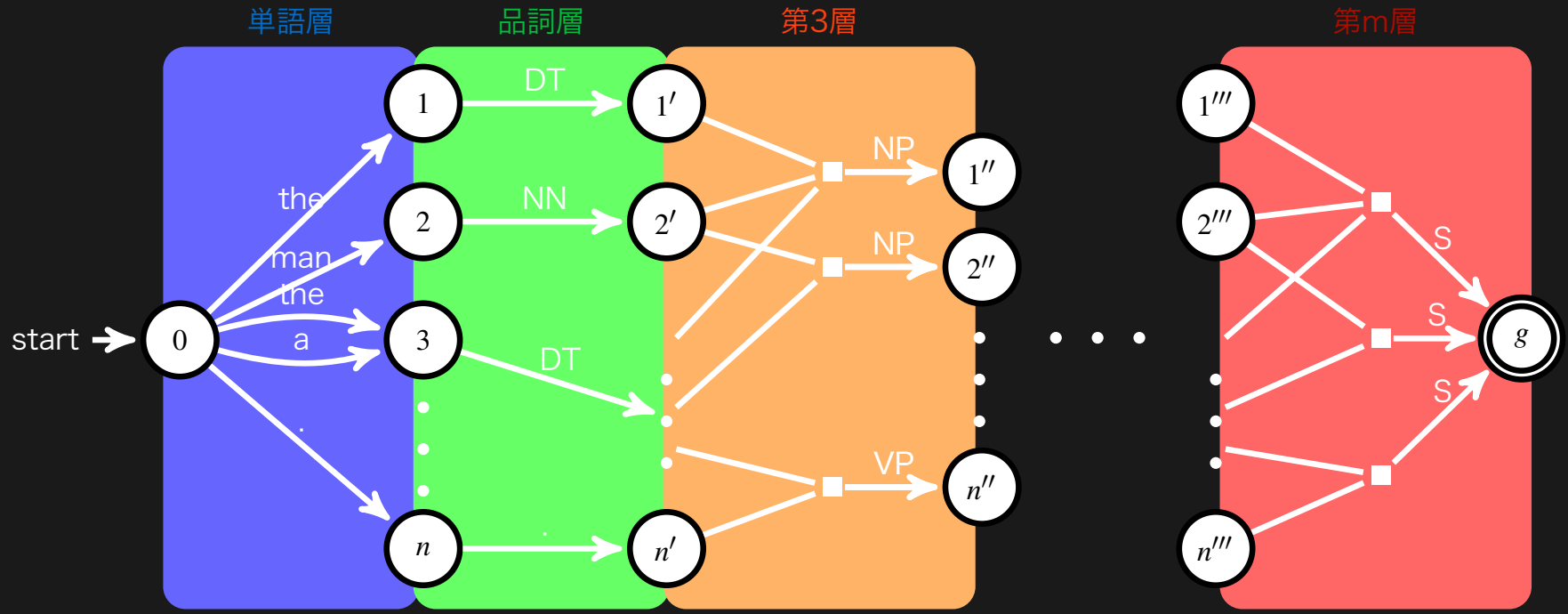


実験結果の分析

- なぜ, 機械翻訳タスクで決定化の結果は遷移数が増大したか?



- 機械翻訳は語彙が多様かつ膨大



- 単語層, 品詞層, 第3層の遷移数

	単語層	品詞層	第3層

- 単語層, 品詞層, 第3層の遷移数

	単語層	品詞層	第3層
ベース	1,547	1,621	1,531

- 単語層, 品詞層, 第3層の遷移数

	単語層	品詞層	第3層
ベース	1,547	1,621	1,531
決定化	462	497	593,222

- 単語層, 品詞層, 第3層の遷移数

	単語層	品詞層	第3層
ベース	1,547	1,621	1,531
決定化	462	497	593,222
無曖昧化	1,476	1,534	10,020

まとめ

今後の課題

まとめと今後の課題

- まとめ
  - 重み付き木オートマトンのための曖昧性解消演算を提案
  - アルゴリズムの正当性, 重み計算の正当性, 閉路を含む場合の適用十分条件などを証明
  - 決定化に比べ, 記憶容量の点で大きな利点がある
- 今後の課題
  - より詳細な実験結果の分析
  - より大規模なデータを使った実験
  - DAGオートマトン (Kamimura 81,82)に対する更なる一般化