

# Transformace Nedeterministických Büchiho Automatů na Slim Automaty

Pavel Šimovec

Fakulta Informatiky, Masarykova Univerzita

30. června 2021

# Motivace

# Motivace

- ▶ Nedeterministické Büchiho automaty obecně nemají vlastnost "dobrý pro Markovovy rozhodovací procesy" (dále GFM, good for Markov decision processes) → nejsou vhodné pro analýzu pravděpodobnostních systémů

# Motivace

- ▶ Nedeterministické Büchiho automaty obecně nemají vlastnost "dobrý pro Markovovy rozhodovací procesy" (dále GFM, good for Markov decision processes) → nejsou vhodné pro analýzu pravděpodobnostních systémů
- ▶ Deterministické  $\omega$ -automaty jsou obecně GFM, ale mohou mít větší velikost či složitější akceptační podmínku (např. Rabinovu, Streettovu)

# Motivace

- ▶ Nedeterministické Büchiho automaty obecně nemají vlastnost "dobrý pro Markovovy rozhodovací procesy" (dále GFM, good for Markov decision processes) → nejsou vhodné pro analýzu pravděpodobnostních systémů
- ▶ Deterministické  $\omega$ -automaty jsou obecně GFM, ale mohou mít větší velikost či složitější akceptační podmínku (např. Rabinovu, Streettovu)
- ▶ Motivace je vyhnout se determinizaci a zároveň mít GFM automaty

# Motivace

- ▶ Nedeterministické Büchiho automaty obecně nemají vlastnost "dobrý pro Markovovy rozhodovací procesy" (dále GFM, good for Markov decision processes) → nejsou vhodné pro analýzu pravděpodobnostních systémů
- ▶ Deterministické  $\omega$ -automaty jsou obecně GFM, ale mohou mít větší velikost či složitější akceptační podmínku (např. Rabinovu, Streettovu)
- ▶ Motivace je vyhnout se determinizaci a zároveň mít GFM automaty
- ▶ Využití v analýze pravděpodobnostních systému - nástroj Mungojerrie, který vyžaduje GFM automaty

# Motivace

- ▶ Nedeterministické Büchiho automaty obecně nemají vlastnost "dobrý pro Markovovy rozhodovací procesy" (dále GFM, good for Markov decision processes) → nejsou vhodné pro analýzu pravděpodobnostních systémů
- ▶ Deterministické  $\omega$ -automaty jsou obecně GFM, ale mohou mít větší velikost či složitější akceptační podmínku (např. Rabinovu, Streettovu)
- ▶ Motivace je vyhnout se determinizaci a zároveň mít GFM automaty
- ▶ Využití v analýze pravděpodobnostních systému - nástroj Mungojerrie, který vyžaduje GFM automaty
- ▶ Motivace zrychlit nástroj Mungojerrie

# Motivace

- ▶ Nedeterministické Büchiho automaty obecně nemají vlastnost "dobrý pro Markovovy rozhodovací procesy" (dále GFM, good for Markov decision processes) → nejsou vhodné pro analýzu pravděpodobnostních systémů
- ▶ Deterministické  $\omega$ -automaty jsou obecně GFM, ale mohou mít větší velikost či složitější akceptační podmínku (např. Rabinovu, Streettovu)
- ▶ Motivace je vyhnout se determinizaci a zároveň mít GFM automaty
- ▶ Využití v analýze pravděpodobnostních systému - nástroj Mungojerrie, který vyžaduje GFM automaty
- ▶ Motivace zrychlit nástroj Mungojerrie
- ▶ Slim automaty jsou nedeterministické a zároveň mají GFM vlastnost



# Přínos v teorii

## Přínos v teorii

- ▶ Slim automaty byly popsány ve článku "Good-for-MDPs Automata for Probabilistic Analysis and Reinforcement Learning", Ernst Moritz Hahn et al., TACAS 2020

## Přínos v teorii

- ▶ Slim automaty byly popsány ve článku "Good-for-MDPs Automata for Probabilistic Analysis and Reinforcement Learning", Ernst Moritz Hahn et al., TACAS 2020
- ▶ První formální popis **weak slim automatů** (také od autorů článku, ale nebyly formálně popsány, slim automaty z původního článku pojmenovány jako **strong**)

## Přínos v teorii

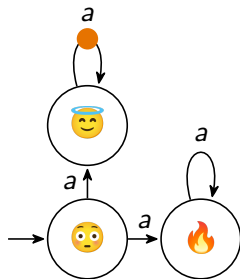
- ▶ Slim automaty byly popsány ve článku "Good-for-MDPs Automata for Probabilistic Analysis and Reinforcement Learning", Ernst Moritz Hahn et al., TACAS 2020
- ▶ První formální popis **weak slim automatů** (také od autorů článku, ale nebyly formálně popsány, slim automaty z původního článku pojmenovány jako **strong**)
- ▶ Algoritmus z článku byl definován pouze na Büchiho automaty (TBA), rozšíření algoritmu na **generalizované** Büchiho automaty (TGBA)

## Přínos v teorii

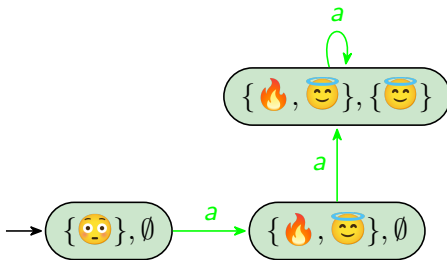
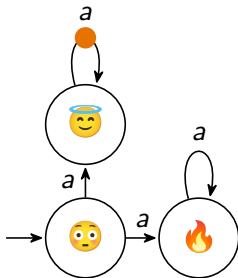
- ▶ Slim automaty byly popsány ve článku "Good-for-MDPs Automata for Probabilistic Analysis and Reinforcement Learning", Ernst Moritz Hahn et al., TACAS 2020
- ▶ První formální popis **weak slim automatů** (také od autorů článku, ale nebyly formálně popsány, slim automaty z původního článku pojmenovány jako **strong**)
- ▶ Algoritmus z článku byl definován pouze na Büchiho automaty (TBA), rozšíření algoritmu na **generalizované** Büchiho automaty (TGBA)
  - ▶ Bez tohoto rozšíření je třeba nejprve provést degeneralizaci

# Ukázka konstrukce slim automatů

# Ukázka konstrukce slim automatů

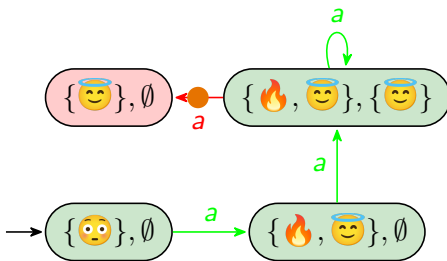
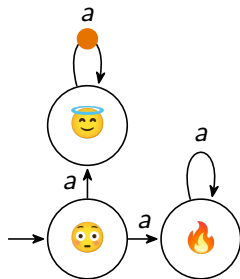


# Ukázka konstrukce slim automatů

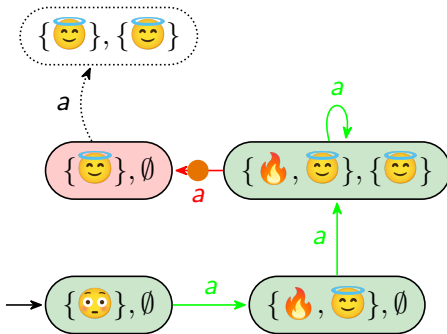
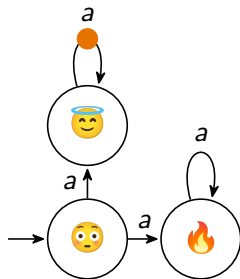




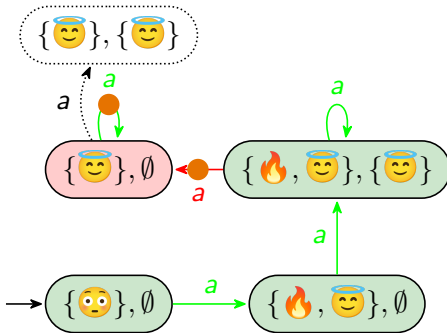
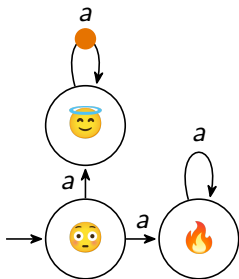
# Ukázka konstrukce slim automatů



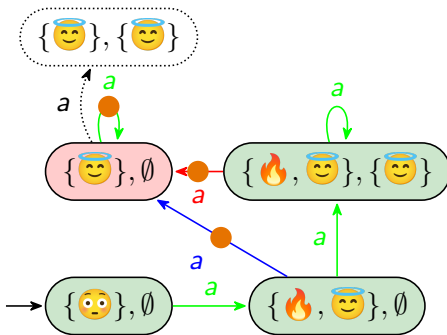
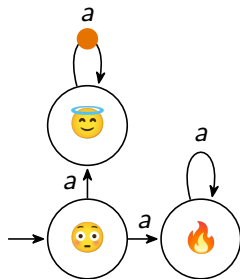
# Ukázka konstrukce slim automatů



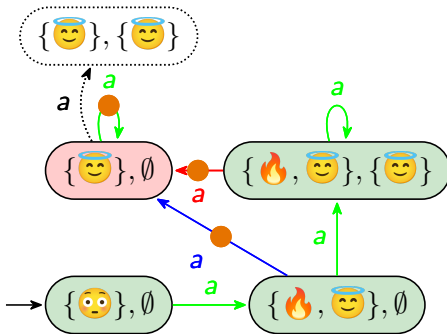
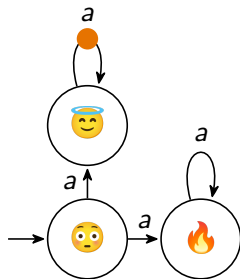
# Ukázka konstrukce slim automatů



# Ukázka konstrukce slim automatů



# Ukázka konstrukce slim automatů



# Implementace

# Implementace

- ▶ Implementace algoritmu z původního článku a rozšíření na TGBA v nástroji Seminor

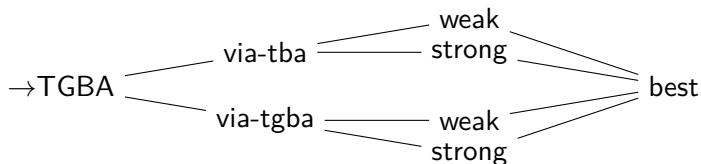
# Implementace

- ▶ Implementace algoritmu z původního článku a rozšíření na TGBA v nástroji Seminador
- ▶ Přidána možnost post-optimalizace pomocí knihovny SPOT



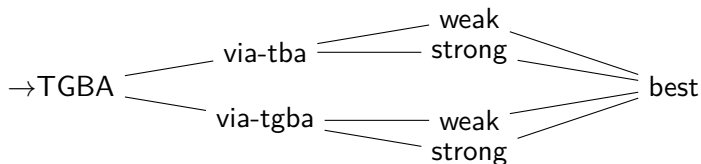
# Implementace

- Implementace algoritmu z původního článku a rozšíření na TGBA v nástroji Seminor
- Přidána možnost post-optimalizace pomocí knihovny SPOT



# Implementace

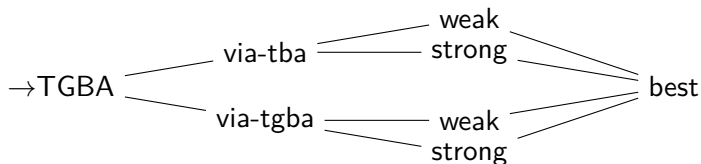
- ▶ Implementace algoritmu z původního článku a rozšíření na TGBA v nástroji Seminor
- ▶ Přidána možnost post-optimalizace pomocí knihovny SPOT



- ▶ Přidány testy na jazykovou ekvivalenci slim automatů se zdrojovým automatem

# Implementace

- ▶ Implementace algoritmu z původního článku a rozšíření na TGBA v nástroji Seminor
- ▶ Přidána možnost post-optimalizace pomocí knihovny SPOT



- ▶ Přidány testy na jazykovou ekvivalenci slim automatů se zdrojovým automatem
- ▶ V době implementace nebyly k dispozici jiné implementace (později nástroj ePMC)

# Evaluace

# Evaluace

Automaty porovnáváme 2 způsoby:

# Evaluace

Automaty porovnáváme 2 způsoby:

- ▶ porovnání počtu stavů automatů
  - ▶ mezi implementovanými možnostmi
  - ▶ s jinými nástroji

# Evaluace

Automaty porovnáváme 2 způsoby:

- ▶ porovnání počtu stavů automatů
  - ▶ mezi implementovanými možnostmi
  - ▶ s jinými nástroji
- ▶ vliv na analýzu pravděpodobnostních systémů v nástroji Mungojerrie

# Porovnání počtu stavů automatů

Porovnání kumulativního počtu stavů slim automatů tvořených Seminátorem z 500 náhodných LTL formulí:

| slim     | weak  | strong | nejlepší |
|----------|-------|--------|----------|
| via tba  | 9160  | 7725   | 7456     |
| via tgba | 10317 | 8793   | 8434     |
| nejlepší | 8958  | 7578   | 7275     |



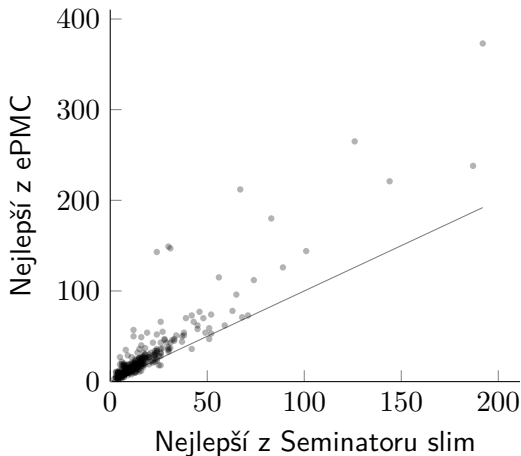
# Porovnání počtu stavů automatů

Porovnání kumulativního počtu stavů s jinými nástroji:  
(použito 500 stejných náhodných LTL formulí)

| nástroj                    | kumulativní počet stavů |
|----------------------------|-------------------------|
| ltl3tela (deterministické) | 3975                    |
| ltl2ldb                    | 4632                    |
| <b>Seminator slim</b>      | <b>7133</b>             |
| ePMC                       | 10132                   |

## Porovnání počtu stavů automatů

Scatter plot porovnávající velikosti slim automatů mezi nástroji ePMC a Seminátor:



# Porovnání vlivu na analýzu pravděpodobnostních systémů

# Porovnání vlivu na analýzu pravděpodobnostních systémů

- ▶ **Mungojerrie** je nástroj pro analýzu pravděpodobnostních systémů, vydaný letos (rok 2021)

# Porovnání vlivu na analýzu pravděpodobnostních systémů

- ▶ **Mungojerrie** je nástroj pro analýzu pravděpodobnostních systémů, vydáný letos (rok 2021)
- ▶ využívá zpětnovazebné učení

# Porovnání vlivu na analýzu pravděpodobnostních systémů

- ▶ **Mungojerrie** je nástroj pro analýzu pravděpodobnostních systémů, vydáný letos (rok 2021)
- ▶ využívá zpětnovazebné učení
- ▶ zpětnovazebné učení v Mungojerrie má 2 fáze:

# Porovnání vlivu na analýzu pravděpodobnostních systémů

- ▶ **Mungojerrie** je nástroj pro analýzu pravděpodobnostních systémů, vydaný letos (rok 2021)
- ▶ využívá zpětnovazebné učení
- ▶ zpětnovazebné učení v Mungojerrie má 2 fáze:
  - ▶ fázi učení s daným počtem epizod
  - ▶ fázi model checkingu

# Porovnání vlivu na analýzu pravděpodobnostních systémů

- ▶ **Mungojerrie** je nástroj pro analýzu pravděpodobnostních systémů, vydaný letos (rok 2021)
- ▶ využívá zpětnovazebné učení
- ▶ zpětnovazebné učení v Mungojerrie má 2 fáze:
  - ▶ fázi učení s daným počtem epizod
  - ▶ fázi model checkingu
- ▶ **cíl je definován GFM automatem**, který nástroji dodáme



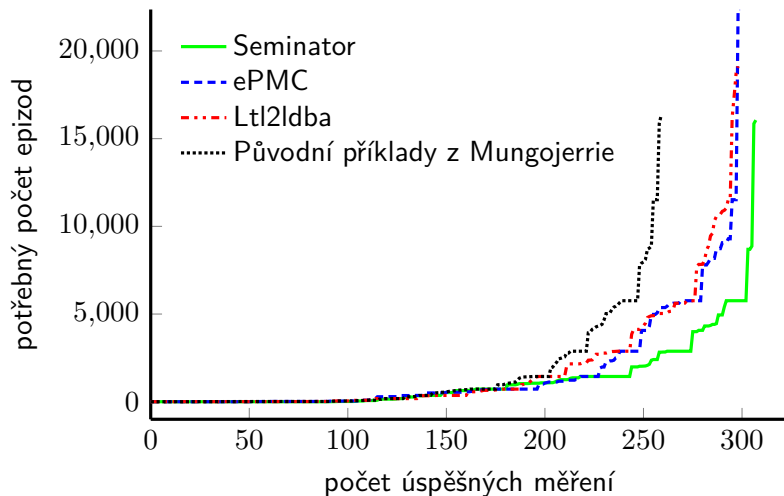
# Porovnání vlivu na analýzu pravděpodobnostních systémů

- ▶ **Mungojerrie** je nástroj pro analýzu pravděpodobnostních systémů, vydaný letos (rok 2021)
- ▶ využívá zpětnovazebné učení
- ▶ zpětnovazebné učení v Mungojerrie má 2 fáze:
  - ▶ fázi učení s daným počtem epizod
  - ▶ fázi model checkingu
- ▶ **cíl je definován GFM automatem**, který nástroji dodáme
- ▶ měříme po kolika epizodách máme pravděpodobnost dosažení cíle rovnou jedné

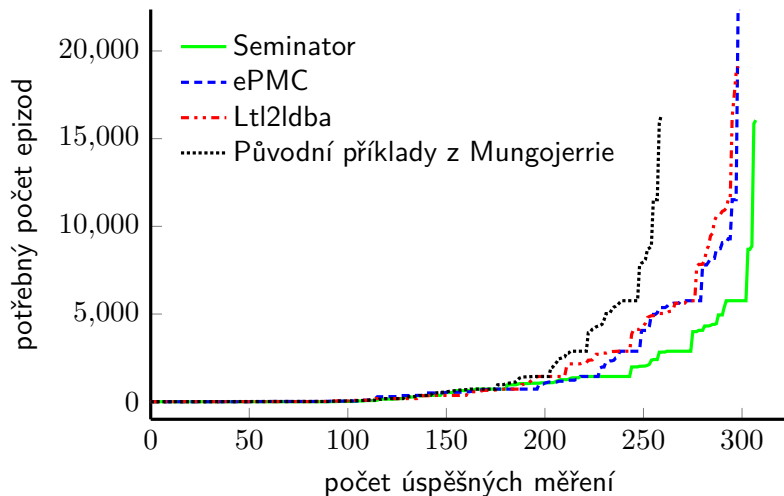
# Porovnání vlivu na analýzu pravděpodobnostních systémů

- ▶ **Mungojerrie** je nástroj pro analýzu pravděpodobnostních systémů, vydáný letos (rok 2021)
- ▶ využívá zpětnovazebné učení
- ▶ zpětnovazebné učení v Mungojerrie má 2 fáze:
  - ▶ fázi učení s daným počtem epizod
  - ▶ fázi model checkingu
- ▶ **cíl je definován GFM automatem**, který nástroji dodáme
- ▶ měříme po kolika epizodách máme pravděpodobnost dosažení cíle rovnou jedné
- ▶ měříme desetkrát pro seedy 0-9

# Porovnání vlivu na analýzu pravděpodobnostních systémů



# Porovnání vlivu na analýzu pravděpodobnostních systémů



*Děkuji za Vaši pozornost!*

# Reakce na posudek vedoucího

- ▶ ...Navíc se domnívám, že zmíněný příklad *Figure 4.1* je špatně, neboť barevně naznačuje existenci slabých slim hran, které však v tomto automatu nemohou být, protože ve všech stavech platí  $S' = \emptyset$ 
  - ▶ Figure 4.1 → Barvy přechodů také zvýrazňují úrovně, barvy odpovídají akceptačním podmínkám ze zdrojového automatu. Není zde spojitost s obarvením hran z přechozího příkladu. Automat není špatně, nemá žádné slim hrany, tudíž je zároveň weak i strong slim automaty.

# Reakce na posudek vedoucího

Otázky:

1. Tabulka 6.2 uvádí, že Seminátor při volbě via tba na formulích z literatury produkuje silné slim automaty s 431 stavy, zatímco při volbě best produkuje automaty s 436 stavy. Jak je to možné, když volba best má vždy produkovat automat s nejvýše tolika stavy jako volba via tba?
  - ▶ To je překlep, nastal při přepisu přepočítaných hodnot, má to být v obou případech 436.

# Reakce na posudek vedoucího

Otázky:

1. Tabulka 6.2 uvádí, že Seminátor při volbě via tba na formulích z literatury produkuje silné slim automaty s 431 stavy, zatímco při volbě best produkuje automaty s 436 stavy. Jak je to možné, když volba best má vždy produkovat automat s nejvýše tolika stavy jako volba via tba?
  - ▶ To je překlep, nastal při přepisu přepočítaných hodnot, má to být v obou případech 436.
2. Proč je v tabulce 6.7 u nástroje Seminátor slim best uveden nižší počet stavů než v tabulce 6.2 pro volby best/best? Čekal bych identické hodnoty.
  - ▶ Srovnáváme pouze automaty kde všechny nástroje dokončí výpočet.

# Reakce na posudek vedoucího

## Otázky:

1. Tabulka 6.2 uvádí, že Seminarator při volbě via tba na formulích z literatury produkuje silné slim automaty s 431 stavy, zatímco při volbě best produkuje automaty s 436 stavy. Jak je to možné, když volba best má vždy produkovat automat s nejvýše tolika stavy jako volba via tba?
  - ▶ To je překlep, nastal při přepisu přepočítaných hodnot, má to být v obou případech 436.
2. Proč je v tabulce 6.7 u nástroje Seminarator slim best uveden nižší počet stavů než v tabulce 6.2 pro volby best/best? Čekal bych identické hodnoty.
  - ▶ Srovnáváme pouze automaty kde všechny nástroje dokončí výpočet.
3. Proč se liší hodnoty na řádcích failures v tabulkách 7.1 a 7.2?
  - ▶ V sedmé kapitole mám zmíněno "We exclude uninteresting benchmarks, where all tools achieve the same result." Odebráním jednoho nástroje vznikly výsledky, kde mají všechny zbývající nástroje stejný výsledek.



# Reakce na posudek oponenta

## Změny v kódu

'V přiloženém ZIP archivu jsem mezi všemi soubory našel seminator-ba2slim/src/slim.hpp o 39 řádcích (včetně prázdných), který má v záhlaví uvedeno „Created by psimovec on 9/16/20“. Ostatní soubory vypadají, že jsou v nástroji Seminátor původní.'  
Změny v kódu:

| soubor                 | změny             |
|------------------------|-------------------|
| Makefile.am            | +1                |
| tests/slim.test        | +16 (Nový soubor) |
| src/breakpoint_twa.cpp | +18               |
| src/breakpoint_twa.hpp | +16               |
| src/main.cpp           | +95               |
| src/slim.hpp           | +39 (Nový soubor) |

Dále jsem vytvořil skripty na evaluaci. (Dohromady přes 200 řádků pouze na tvorbu a zpracování dat, nepočítaje interpretaci a zobrazení výsledků)

# Reakce na posudek oponenta

## Literature/random datasety

'V tabulce 6.1 netuším, co je „literature“ a co je „random“.'

Tyto datasety byly již v seminator-evaluation, jedná se o nesemideterministické automaty vybrané pro evaluaci Seminátoru.

random → 500 náhodných LTL formulí generovaných SPOTem tak, aby byly nesemideterministické.

literature → 20 formulí vybraných pro evaluaci Seminátoru z literatury:

- ▶ beem.ltl: R. Pelánek: BEEM: Benchmarks for explicit model checkers. Proceedings of Spin'07. LNCS 4595.
- ▶ liberrouter.ltl: Holeček, T. Kratochvila, V. Řehák, D. Šafránek, and P. Šimeček: Verification Results in Liberrouter Project. Tech. Report 03, CESNET, 2004.
- ▶ dac: M. B. Dwyer and G. S. Avrunin and J. C. Corbett: Property Specification Patterns for Finite-state Verification. Proceedings of FMSP'98.
- ▶ eh: K. Etessami and G. J. Holzmann: Optimizing Büchi Automata. Proceedings of Concur'00. LNCS 1877.