Szoftver verifikáció és validáció

A CSMA/CD protokoll modellezése és verifikációja

Móczár Zoltán
moczar@tmit.bme.hu

Budapesti Műszaki és Gazdaságtudományi Egyetem
Távközlési és Médiainformatikai Tanszék

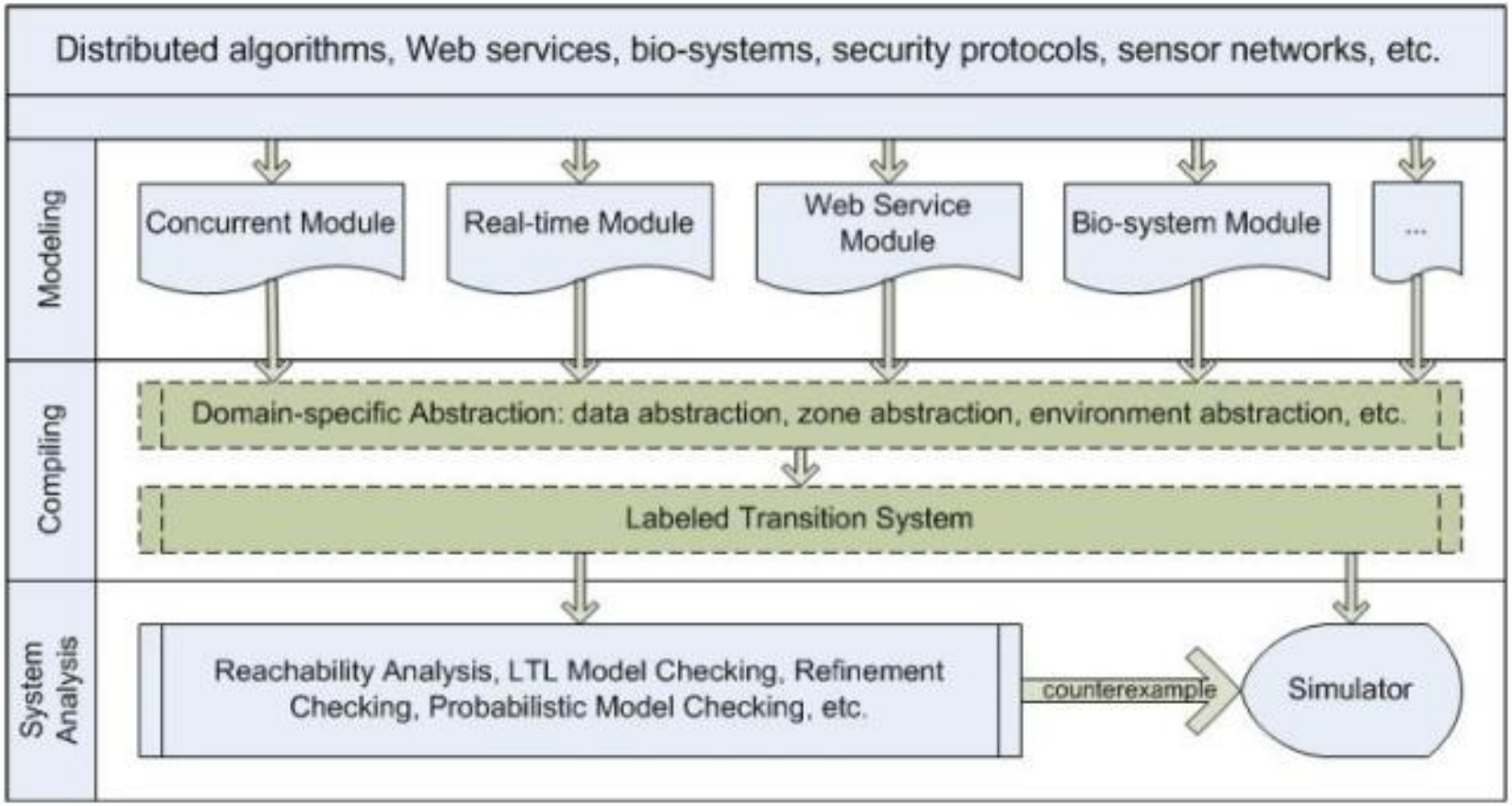
Áttekintés

- a verifikációs eszköz (PAT) bemutatása
  - főbb jellemzők
  - a rendszer felépítése és működési elve
  - előnyös tulajdonságok
- esettanulmány
  - a probléma definiálása
  - formális modellezés
  - kritikus tulajdonságok ellenőrzése
  - szimulációs eredmények

Process Analysis Toolkit (1)

- a PAT rendszer alapvető jellemzői
  - valósidejű rendszerek formális modellezése, verifikációja és szimulációja
  - különböző modellellenőrzési technikák támogatása
  - 1400+ regisztrált felhasználó, 300+ szervezet, 40+ ország
  - C# nyelven készül
  - *http://www.comp.nus.edu.sg/~pat/*

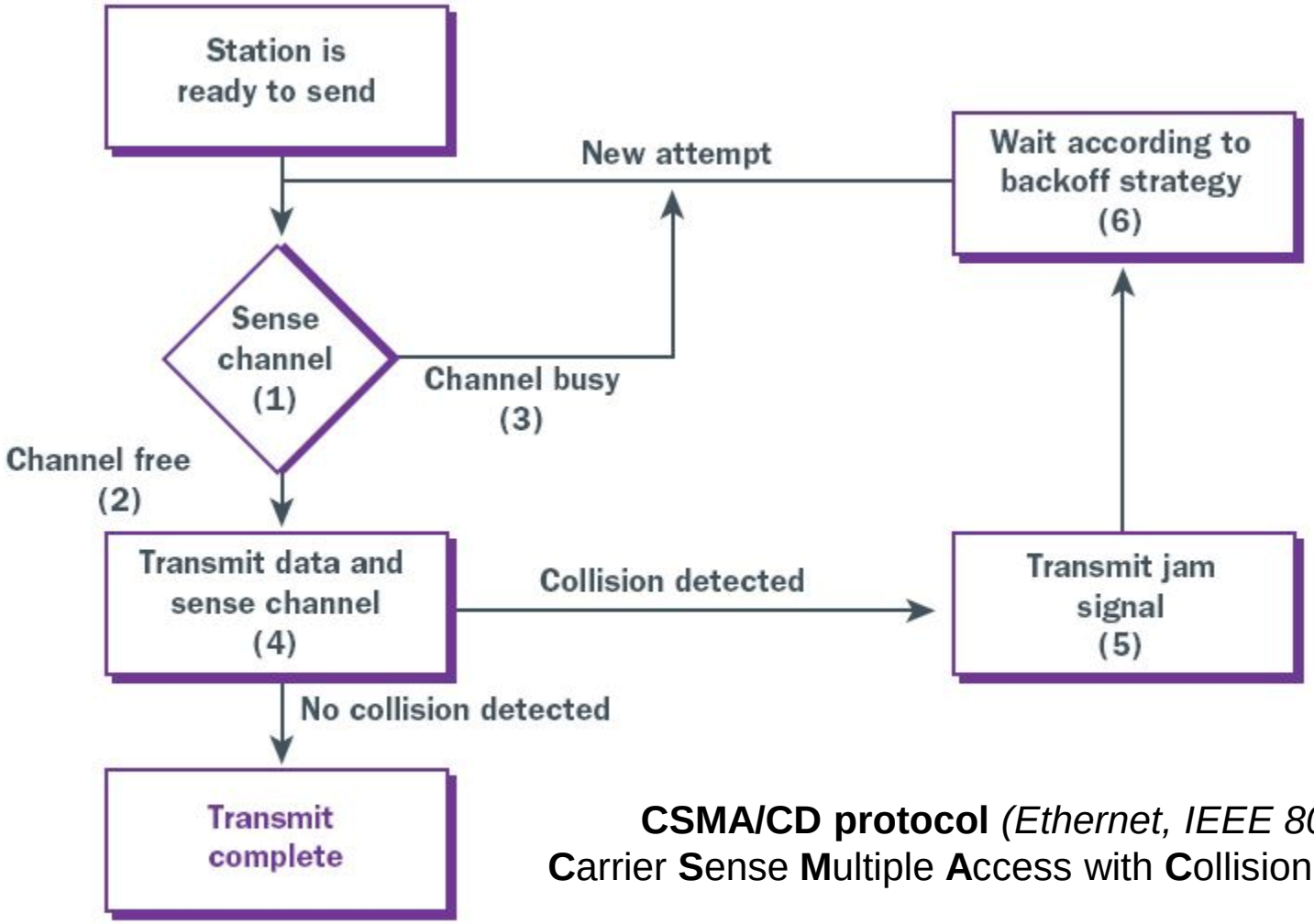
Process Analysis Toolkit (2)



Process Analysis Toolkit (3)

- előnyös tulajdonságok
  - felhasználóbarát modellszerkesztő és szimulátor
  - nagy állapottér esetén is jó teljesítmény
    - különböző optimalizációs technikák használata (pl. részleges rendezés redukció, párhuzamos modellellenőrzés)
  - könnyen bővíthető
  - számos területen és projektben alkalmazták már sikeresen
  - ~40 publikáció 2008 óta

Esettanulmány



CSMA/CD protocol *(Ethernet, IEEE 802.3)*
Carrier **S**ense **M**ultiple **A**ccess with **C**ollision **D**etection

Timed CSP

$$
\begin{array}{ll}
P = Stop \mid Skip & \text{– primitives} \\
\mid\ e \rightarrow P & \text{– event prefixing} \\
\mid\ [b]P & \text{– state guard} \\
\mid\ \mathit{if}\ b\ \mathit{then}\ P\ \mathit{else}\ Q & \text{– if-then-else} \\
\mid\ P \,\Box\, Q & \text{– general choice} \\
\mid\ P \parallel Q & \text{– parallel composition} \\
\mid\ P;\ Q & \text{– sequential composition} \\
\mid\ P \setminus X & \text{– hiding} \\
\mid\ P \mathrel{\widehat{=}} Q & \text{– process referencing} \\
\mid\ \mathit{Wait}[d] & \text{– delay} \\
\mid\ P\ \mathit{timeout}[d]\ Q & \text{– timeout} \\
\mid\ P\ \mathit{interrupt}[d]\ Q & \text{– timed interrupt} \\
\mid\ P\ \mathit{within}[d] & \text{– react within some time} \\
\mid\ P\ \mathit{waituntil}[d] & \text{– wait until} \\
\mid\ P\ \mathit{deadline}[d] & \text{– deadline}
\end{array}
$$

A protokoll modellezése (1)

- feltételezések
  - 10 Mb/s Ethernet
  - maximális jelterjedési idő: $\sigma = 26\ \mu s$
  - üzenetek mérete: 1024 byte
  - üzenet átviteli ideje: $\lambda = 808\ \mu s$
  - újraküldés: 2σ (52 μs) időn belül (~~backoff strategy~~)
  - az átvitel során nem veszik el üzenet
  - a végpontokon nincs puffer az üzenetek tárolására
- két rendszerkomponens
  - küldő (sender)
  - csatorna (bus)

A protokoll modellezése (2)

Category	Name	Description
Global Definition	N	Constant: number of senders
	channel newMess 0	Sender gets messages to send
	channel begin 0	Sender starts sending message
	channel busy 0	Sender senses a busy bus
	channel cd 0	Sender detects a collision
	channel end 0	Sender completes its transmission
Sender Behavior	WaitFor(i)	Sender i is waiting for a message from the upper level
	Trans(i)	Sender i is sending a message
	Retry(i)	Sender i is waiting to retry after detecting a collision or a busy bus
Bus Behavior	Idle	Bus is free, no sender is transmiting
	Active	One sender starts transmitting and is detecting collision
	Active1	One sender is transmitting messages, bus is busy
	Collision	Collision occurs and bus broadcasts the collision information to all senders

A protokoll modellezése (3)

A küldő modellje (példa):

$$
\begin{aligned}
WaitFor(i) = {} & (cd?i \rightarrow WaitFor(i)) \\
& \Box\ (newMess!i \rightarrow ((begin!i \rightarrow Trans(i)) \\
& \qquad \Box\ (busy?i \rightarrow Retry(i)) \\
& \qquad \Box\ (cd?i \rightarrow Retry(i))));
\end{aligned}
$$

$$
\begin{aligned}
Trans(i) = {} & (cd?i \rightarrow Retry(i) within [0, 52]) \\
& \Box\ (atomic\{end!i \rightarrow Skip\} within [808, 808]; \\
& \quad WaitFor(i));
\end{aligned}
$$

$$
\begin{aligned}
Retry(i) = {} & (newMess!i \rightarrow ((begin!i \rightarrow Trans(i) within [0, 52]) \\
& \qquad \Box\ (busy?i \rightarrow Retry(i) within [0, 52]) \\
& \qquad \Box\ (cd?i \rightarrow Retry(i) within [0, 52])));
\end{aligned}
$$

A protokoll modellezése (4)

- további modellek
  - a csatorna modellje
  - a broadcast folyamat modellje

- a CSMA/CD protokoll modellje

$$CSMACD = (|||\ x : \{0..N-1\} @ WaitFor(x))\ |||\ Idle;$$

Verifikáció

Kritikus követelmények:

- holtpont mentesség (P0)
  (deadlock freeness)
- divergens működés elkerülése (P1)
  (timed divergence-free)
- ütközés detektálása korlátos időn belül (P2)
  (collision detection in a given bounded delay)

Timed refinement checking:

$$\#assert\ CSMACD\ refines < T >\ Spec;$$

a protokoll modellje

ellenőrizni kívánt tulajdonság

Eredmények

Property	No. of Senders	Result	#States	#Transitions	Time(sec)
P0	4	Yes	787	1075	0.20
P0	5	Yes	2789	3847	0.60
P0	6	Yes	8851	12227	2.28
P0	7	Yes	26109	35991	8.43
P0	8	Yes	73123	100419	31.03
P0	9	Yes	196997	269319	108.69
P0	10	Yes	514915	700611	361.58
P1	4	Yes	787	1075	0.17
P1	5	Yes	2789	3847	0.66
P1	6	Yes	8851	12227	2.53
P1	7	Yes	26109	35991	9.79
P1	8	Yes	73123	100419	35.69
P1	9	Yes	196997	269319	123.24
P1	10	Yes	514915	700611	407.12
P2	4	Yes	787	1075	0.20
P2	5	Yes	2789	3847	0.90
P2	6	Yes	8851	12227	3.69
P2	7	Yes	26109	35991	14.74
P2	8	Yes	73123	100419	55.38
P2	9	Yes	196997	269319	196.35
P2	10	Yes	514915	700611	655.38

Tesztkörnyezet: Intel® Core™ 2 Duo (2,33 GHz) CPU, 3,25 GB memória, MS Windows XP3

Köszönöm a figyelmet!