













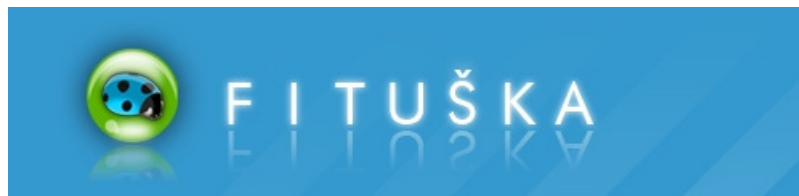


**Fórum:**

-  [Hledat](#)
-  [Členové](#)
-  [FAQ](#)
-  [Uživatelský panel](#)
-  [0 nových zpráv](#)
-  [Odhlásit \[ skeWer \]](#)

**Odkazy:**

-  [WIS FIT](#)
-  [Rozvrh pro 3BIT](#)
-  [Osobní rozvrh](#)
-  [Fórum SU FIT](#)
-  [Pastebin pro fitušku](#)
-  [FIT Wiki](#)
-  [vidoaq's FIT server](#)
-  [Fituška na Facebooku](#)
-  [VUT Burza](#)



Poslední návštěva: 02 úno 2011 12:32 am

Právě je 02 úno 2011 08:59 am

Svátek má Nela.

Nyní je lichý týden.

**Termíny:** ([přidat](#), [editovat](#), [log](#))[IMP](#) - Druhý opravný termín (3.2.)[IMS](#) - Druhý opravný termín (4.2.)[Dotazník SU FIT a SKAS \(12.2.\)](#) - [detaily](#)[IBP](#) - Odevzdání BP (18.5.) - [detaily](#)

Zbývá 1 dnů, 3 hodin a 0 minut

Zbývá 1 dnů, 23 hodin a 0 minut

Zbývá 10 dnů, 15 hodin a 0 minut

Zbývá 105 dnů, 9 hodin a 0 minut

[Vyhledat témata bez odpovědí](#) | [Zobrazit aktivní témata](#)[Zobrazit nepřečtené příspěvky](#) | [Zobrazit nové příspěvky](#) | [Zobrazit vaše příspěvky](#)[Obsah fóra](#) » [Předměty \(2010/2011\)](#) » [3BIT](#) » [Zimní semestr](#) » [IMS](#)

## [Souhrn témat]



nové téma



odpovědět

Stránka 2 z 2 [ Příspěvků: 18 ]

[Přejít na stránku Předchozí 1, 2](#)[Sledovat toto téma](#) | [Přidat do záložek](#) | [Verze pro tisk](#) | [Napsat e-mail](#) | [Ignorovat toto téma](#)[Předchozí téma](#) | [Následující téma](#)**Autor****Zpráva**[skeWer](#)**Předmět příspěvku:** Re: [Souhrn témat] **Napsal(a):** 01 úno 2011 06:41 pm

4BIT

## Řízení kombinované simulace

**Jak to funguje ?**

- Kombinovaná simulace kombinuje diskrétní a spojitou:
  - Sleduje spojitě veličiny (s nějakým krokem) a používá numerickou integraci.
  - Plánuje události a vkládá je do kalendáře.
- Kromě [dokročení na konec simulace](#) tedy musíme dávat pozor, abychom nepřekročili čas nějaké události - [dokročení na událost](#).
- Protože chceme sledovat spojitě veličiny po celou simulaci, [končíme stejně jako u spojitě simulace - až vyprší čas](#) (nikoliv když je kalendář prázdný).

**Pseudokód**

Ve spojitě simulaci musíme vždy před krokem num. metody **updateovat integrátory**. Následující kód předpokládá, že `krok_numericke_metody()` si je updatuje uvnitř sebe (buď to PPMu napište, nebo před každé volání funkce volejte `Update_integrators()`... to stejné jde asi říct o `print_results()`, ačkoliv tisk výsledků možná ani PP nutně nechce).

**Kód:**[Číslování řádků on/off](#) | [Rozbalit/Sbalit](#) | [Vybrat vše](#)

```

1. inicializace poc. stavu a promennych a casu;
2.
3. while (cas < koncovy_cas) {
4.     vyber_zaznam_z_kalendare();
5.
6.     while ( (cas + krok < koncovy_cas) && (cas + krok < aktivacni_cas_pristi_
       udalosti) ) { /"nepřekročení" události / konce
7.         krok_numericke_metody();
8.         cas += krok;
9.     }
10.
11.    //v tomhle místě jsme buď kousek před událostí (kousek <= normalni_krok), nebo
       kousek před koncem simulace
12.    //čili před provedením události / koncem simulace musíme udělat ještě jeden krok,
       ale jeho velikost určíme
13.    //podle toho, jestli bueme provádět událost nebo končit
14.
15.    if (aktivacni_cas_pristi_udalosti > koncovy_cas) { //první udalost leží mimo
       simulaci -> kašlem na ni
16.        krok = koncovy_cas - cas;
17.        krok_numericke_metody();
18.        cas += krok; //stejně tak bychom mohli napsat: cas = koncovy_cas
19.        konec_simulace();
20.    }
21.    else {
22.        krok = aktivacni_cas_pristi_udalosti - cas;
23.        krok_numericke_metody();

```

```

24.     cas += krok; //stejně tak bychom mohli napsat: cas = aktivacni_cas_pristi_
      udalosti
25.     vyber_zaznam_kalendare();
26.     proved_udalost();
27.     krok = normalni_delka_kroku;
28. }
29. }

```

Součástí každého odborníka v IT, který se zabývá sítěmi, je, že se **nebojí** číst RFC dokumenty a umí si je najít. -Matoušek Petr, Ing., Ph.D.  
**They are my space marines and they shall know no fear. -The Emperor of Mankind**



Nahoru

profil sz email

upravit citace

skeWer

Předmět příspěvku: Re: [Souhrn témat]

Napsal(a): 01 úno 2011 09:39 pm

4BIT



## Řízení simulace - "zvláštní případy"

Pod "zvláštní případy" řadím dvě věci, na které se ptá na zkoušce:

- Ukončování simulace zvláštní událostí
- Vyvolávání stavové události

### • Ukončování simulace zvláštní událostí

Zde jde o to, že simulace běží, dokud první událost v kalendáři není nějaká zvláštní událost. Tzn. může se to potenciálně objevit v diskrétní nebo kombinované simulaci.

Pak vždy těsně před provedením události musíme testovat, jestli se nejedná o událost speciální (ukončující). Pokud ano, končíme simulaci.

Kód:	Číslování řádků on/off   Rozbalit/Sbalit   Vybrat vše
1. while(1) {	
2.     ...	
3.     if(udalost == ukoncujiuci_udalost) konec_simulace();	
4.     else proved_udalost();	
5.     ...	
6. }	

#### Příklad

Konkrétní implementace např. zde.

### • Vyvolávání stavové události

V kombinované simulaci můžeme mít kromě klasicky plánovaných událostí také události stavové. To jsou události, které jsou vyvolávané na základě **změny** hodnoty určité spojitě proměnné. Respektive: stavová událost vzniká ve chvíli, kdy se platnost stavové podmínky změní z false na true, případně naopak (následně může stavová podmínka platit jak dlouho chce - stavová událost se vyvolá jen při tom překlopení).

#### Problémy

- U plánovaných událostí je nutné dokročit. U stavových událostí ale dopředu nevíme, kdy k ní dojde.
- Může se stát, že přijde nějaká stavová událost a náraz změny hodnotu některé spojitě proměnné. U numerických metod se toto musí ošetřit: víceřadkové metody si normálně pamatují několik předchozích stavů, které by teď nebylo správné použít při výpočtu - musí se začít znovu od počátku. I některé jednokrokové metody v rámci optimalizací si pamatují některé minulé stavy.
- Při nevhodně zvolené podmínce se nemusí povést detekce kvůli nepřesnostem neumerickýchmetod. Při špatně zvoleném kroku můžeme někdy "minout" změnu. Obojí viz slide 271.

Na místo obyčejného dokročení se nejdříve spočte krok numerické integrace a **pokud se zjistí, že se změnila stavová podmínka**, provede se jakýsi **rollback** v čase zpět a **zkrátí se krok**. Pak se (v další iteraci) z toho původního času šoupneme o menší krok.

Takhle se posunujeme po stále menších a menších krocích, až se dostaneme na nějakou minimální délku (tu si stanovíme podle potřeb, odvíjí se od ní přesnost detekce).

**V tento okamžik vyvoláme stavovou událost**, obnovíme původní velikost kroku a pokračujeme.

#### Pseudokód

Zde je třeba oproti základní kombinované simulaci nahradit každé volání `krok_numericke_metody();` **\***. Nejlépe si následující kód obalit nějakou funkcí a pak ji volat místo `krok_numericke_metody();`.

Kód:	Číslování řádků on/off   Rozbalit/Sbalit   Vybrat vše
1. <code>ulozeni_stavu_a_casu();</code>	

```

2.   krok_numericke_metody(); //včetně posunutí času
3.   vyhodnoceni_stavovych_podminek();
4.   if (podminka_zmenena) {
5.       if (krok <= minimalni_krok) { //technicky vzato by krok neměl nikdy být
menší než minimální... ve slidech (str. 273) to ale má takhle
6.           potvrzeni_zmen_podminek();
7.           proved_stavovou_udalost();
8.           krok = normalni_krok;
9.       } else {
10.          obnova_stavu_a_casu();
11.          zmensi_krok(); //v SIMLIBu: krok = krok / 2; myslím že uzná obojí
12.          if (krok < minimalni_krok) krok = minimalni_krok;
13.      }
14.  }
15.

```

\*: na přednášce to možná přímo nezmiňoval a ve slidech taky ne, ale imo to tak je. Ke stavové události by mohlo dojít i při dokračování na událost nebo na konec simulace - i v těch případech musíme být schopni provádět rollback. Pokud ale dojde ke stavové události při dokračování na událost / konec simulace, znamenalo by to, že jsme dokoročili správně. Museli bychom to tedy nějak ošetřovat, třeba nějak takhle:

Kód:	Číslování řádků on/off   Rozbalit/Sbalit   Vybrat vše
1.	if (aktivacni_cas_pristi_udalosti > koncovy_cas) { //první udalost leží mimo simulaci -> kašlem na ni
2.	krok = koncovy_cas - cas;
3.	krok_numericke_metody(); //tohle nahradíme tím kódem výše
4.	if(doslo_ke_zmene_stavove_podminky) continue; //vyskočíme zpět do hlavní smyčky, v příští iteraci s do téhle if-větvě dostaneme znovu a skončíme jen tehdy, když nedojde ke změně stavové podmínky
5.	cas += krok; //stejně tak bychom mohli napsat: cas = koncovy_cas
6.	konec_simulace();
7.	}

Stejně tak v kódu u dokročení na událost.

Součástí každého odborníka v IT, který se zabývá sítěmi, je, že se **nebojí** číst RFC dokumenty a umí si je najít. -Matoušek Petr, Ing., Ph.D.  
They are my space marines and they shall know no fear. -The Emperor of Mankind



Nahoru



Zobrazit příspěvky za předchozí: Všechny příspěvky Seřadit podle Čas odeslání Vzestupně Přejít



Stránka 2 z 2 [ Příspěvků: 18 ]

[Přejít na stránku Předchozí 1, 2](#)

[Obsah fóra](#) » [Předměty \(2010/2011\)](#) » [3BIT](#) » [Zimní semestr](#) » [IMS](#)

**Kdo je online**

Registrovaní uživatelé: Alibaba, beny, birkoff, BlueGene, Cospel, Dark, david\_b, Ehm, El Kamil, hacky, Hasimir, HeX, Hlavaja, imli, jilji, kedysek, Krammenzer, Liwin, Itfatal, muffik, Nazghul, PG, Polos, rockmachine, skeWer, Srnka, tapak, treslo, v3s, Vaan, zolex

**Můžete** zakládat nová témata v tomto fóru

**Můžete** odpovídat v tomto fóru

**Můžete** upravovat své příspěvky v tomto fóru

**Můžete** mazat své příspěvky v tomto fóru

**Můžete** přikládat soubory v tomto fóru

Hledat:  Přejít

Přejít na: IMS Přejít

© 2007-2010 fituska, powered by phpBB © 2000-2008 phpBB Group  
Český překlad – phpBB.cz

Brought into the light by

