



Matlab (10)

Matlab – SimEvents

SimEvents je rozšírenie Simulinku o nástroje na modelovanie a simuláciu diskretných udalostných systémov použitím zásobníkov a serverov. So SimEvents môžeme vytvoriť simuláciu diskretných udalostí modelu v Simulinku na simulovanie prechodu entít cez sieť zásobníkov, serverov, brán a prepínačov založených na udalostiach. SimEvents a Simulink poskytujú integrované prostredie na modelovanie hybridných dynamických systémov obsahujúcich spojité, diskkrétne a diskrétno-udalostné komponenty.

Diskrétno-udalostná simulácia, alebo udalostná simulácia, dovoľuje prechod stavu systému v závislosti od asynchrónnych diskretných javov, ktoré sa nazývajú udalostami. Na porovnanie, simulácia formálne založená na diferenciálnych rovniciach, v ktorých je čas závislou premennou, je časovo založená simulácia, pretože prechod stavu závisí od času. Simulink je navrhnutý pre časovo založené simulácie, zatiaľ čo SimEvents je navrhnutý pre diskretno-udalostné simulácie. Naša voľba rôznych spôsobov simulácie môže závisieť od jednotlivých javov, ktoré skúmame, alebo od spôsobu výberu, ako ich skúmať. Uvedieme niekoľko príkladov na ilustráciu týchto rozdielov:

Predpokladajme, že nás zaujíma trajektória vzlietajúceho lietadla. V tomto prípade použijeme časovo založenú simuláciu, pretože hľadanie trajektórie vyžaduje riešenie diferenciálnych rovníc.

V inej situácii predpokladajme, že nás zaujíma, ako dlho v priemere čaká lietadlo, kým príde na rad a môže použiť vzletovú dráhu. Avšak nezaujíma nás do detailov pohybu lietadla, ktoré má voľnú dráhu a môže vzlietnuť. Mali by sme použiť diskretno-udalostnú simuláciu, v ktorej podstatné udalosti zahŕňajú príchod nového lietadla na vzletovú dráhu a uvoľnenie vzletovej dráhy na vzlet lietadla, ktoré čaká v rade.

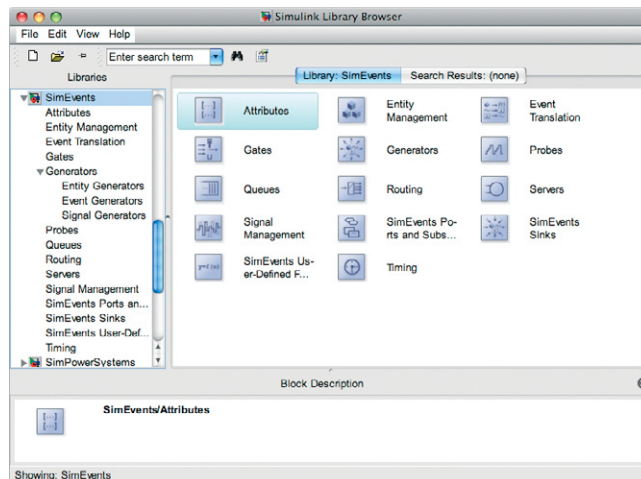
A na koniec predpokladajme, že nás zaujíma, ako dlho lietadlo čaká v rade, ale chceme do detailov namodelovať vzlet namiesto použitia štatistického rozdelenia na namodelovanie času, ktorý každé lietadlo potrebuje na použitie vzletovej dráhy. V tomto prípade bude najlepšie použiť kombináciu časovo založených a diskretno-udalostných simulácií, kde časovo založený aspekt riadi detaily vzletu lietadiel a diskretno-udalostný aspekt správanie lietadiel čakajúcich v rade.

Diskrétno-udalostná simulácia typicky zahrnuje diskretné položky. Podľa definície sa tieto položky v SimEvents nazývajú entity. Entity môžu prechádzať cez sieť zásobníkov, serverov, brán a prepínačov počas simulácie. Entity môžu obsahovať dáta, ktoré sa v SimEvents nazývajú atribútmi.

V diskretno-udalostných simuláciách je udalosť okamžitý diskretný incident, ktorý mení stav premenných, výstupy a výskyt ďalších udalostí. Podporované udalosti v SimEvents sú:

- postup entity z jedného bloku do druhého,
- ukončenie služby na entite v serveri,
- nulové kríženie signálu pripojeného do bloku, ktorý je nakonfigurovaný reagovať na nulové kríženie; tieto udalosti sa tiež nazývajú spúšťacími (nábehovými) hranami,
- volanie funkcie, ktorá je diskretným vyvolaním požiadavky prechádzajúcej z bloku do bloku špeciálnym signálom nazývaným Function-call signal (signál volajúci funkciu); volanie funkcie je odporúčanou možnosťou, ako spraviť Stateflow bloky a bloky v knižnici Simulinku odpovedajúce na asynchrónne zmeny stavov.

Udalosti v simulácii môžu závisieť jedna od druhej. Jedna udalosť môže byť základnou príčinou ďalšej udalosti. Príchod prvej entity v zásobníku zapríčiní, že dĺžka zásobníka sa zmení z 0 na 1. Jedna udalosť môže spustiť vznik ďalšej udalosti, ale iba za istých podmienok. Dokončenie služby entity spustí príchod entity zo servera, ale iba v prípade, že nasledujúci blok je schopný prijať príchod tejto entity. V tomto prípade



Obr.1 Bloky knižnice SimEvents

jedna udalosť umožní vznik druhej udalosti, ale formálne sa to nestáva. Udalosti nemajú grafickú reprezentáciu. Ich výskyt môžeme posúdiť pozorovaním ich dôsledkov použitím Instantaneous Event Counting Scope bloku (blok okamžitého počítania udalosti) alebo nástroja vytvárajúceho záznamy o udalostiach.

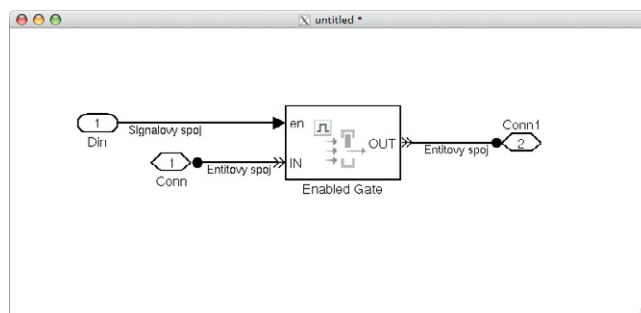
SimEvents je súčasťou distribúcie Matlab od verzie R2006a. V súčasnosti (R2008b) je vo verzii 2.3 a ide o multiplatformový softvér. Po inštalácii sa automaticky doplní do Simulink Library Browser-a (obr. 1).

Vkladanie jednotlivých blokov funguje metódou „drag and drop“. Predtým, ako sa začne vytvárať model, treba zmeniť nastavenia Simulinku na modelovanie diskretno-udalostných systémov príkazom v Matlabe

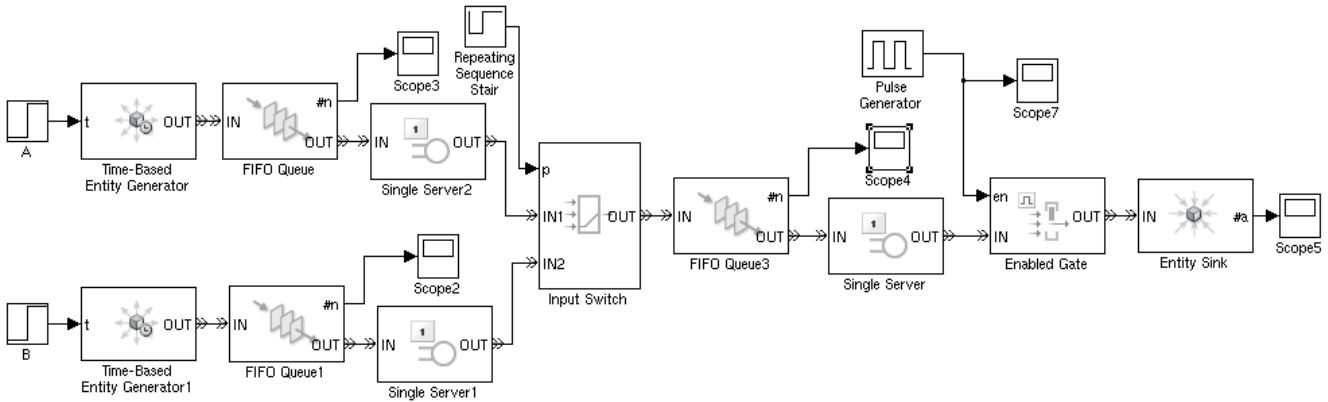
`simeventsstartup('des')` – tento príkaz nastaví Simulink pre diskretno-udalostné simulácie. Ak treba kombinovať diskretno-udalostný systém so spojitým, musíme použiť hybridné nastavenie.

`simeventsstartup('hybrid')`

SimEvents zavádza do Simulinku odlišný typ prepojenia medzi blokmi. Ide o entitové spojenia. Od klasických signálnych spojení sa odlišujú iným konektorom na vstupe aj na výstupe. Zatiaľ čo klasické signálne spojenia musia smerovať od výstupu jedného bloku na vstup iného, entitové bloky majú konektory zameniteľné (obr. 2).



Obr.2 Signálne a entitové spojenie



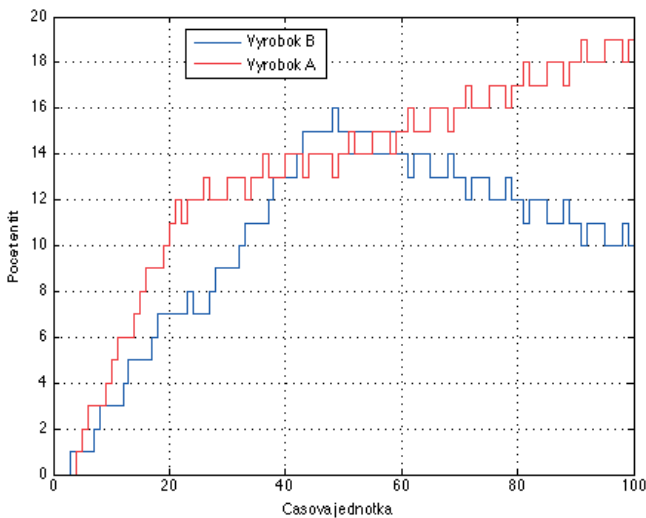
Obr.3 Výrobný proces v SimEvents

Využitie blokov SimEvents si ukážeme na nasledujúcom príklade. Naším cieľom bude sledovanie plnenia zásobníkov v procese balenia.

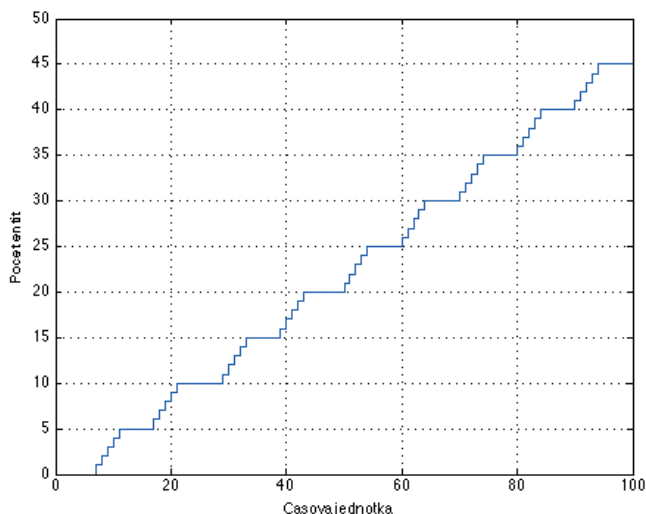
Výrobok (entita) A aj B sa generujú pomocou generátora entít na základe vstupného portu. Signál Step určuje, ako rýchlo sa budú entity generovať. Výrobok A sa generuje každú časovú jednotku. V našom prípade 1 sekunda (môže ísť aj o pomerné jednotky) až po čas 20 sekúnd a následne sa výrobný proces spomalí na jeden výrobok za dve jednotky. Podobným spôsobom sa prepne výroba výrobku B v čase 40 sekúnd. FIFO zásobníky slúžia na skladovanie výrobkov pred spracovaním preto, lebo môže nastať situácia, keď sa výrobky generujú rýchlejšie, ako sa stihnú spracovať. Server pred Switchom (prepínačom) zaručuje, že cez Switch vždy prejde jeden výrobok za časovú jednotku. Poradie, akým prechádzajú entity Switchom, je dané signálom vstupujúcim do Switchu. V našom prípade sme nastavili signál tak, aby

Switch púšťal najskôr dva výrobky A a tri výrobky B. Následne sa výrobky ukladajú opäť do zásobníka. Ďalší server symbolizuje spracovanie balenia. Tento proces využíva aj Gate (bránu) na spracovanie práve piatich prvkov v poradí, akým sme ich púšťali Switchom. Čas, kedy je brána zatvorená, môžeme považovať za plnenie baliaceho stroja. Zabalenie kolekcie výrobkov sme nastavili na jednu časovú jednotku na výrobok. Na grafe vidíme stav zásobníkov (obr. 4).

Zásobník A sa naplňa rýchlejšie a po spomalení výroby stúpa pomalším tempom. Zásobník B sa po spomalení vyprázdňuje. Je to dané tým, že kolekcia potrebuje viac výrobkov B ako A. Zásobník pred strojom sa neustále naplňa a je možné, že pri dlhšej simulácii by sa jeho kapacita vyčerpala. Z tohto dôvodu môžeme tvrdiť, že do procesu môžeme pridať napríklad ďalší stroj na balenie. Výstup entít zo stroja môžeme sledovať pomocou bloku Sink (obr. 5).



Obr.4 Zásobník A a B



Obr.5 Výstup balenia

Tento akademický príklad ukazuje využitie len niektorých prvkov SimEvents a spolu s mnohými ďalšími je vynikajúci nástroj na modelovanie procesov vo výrobe, ako napríklad v našom prípade sledovanie výrobkov, zásobníkov a mnoho iných entít. SimEvents je pomerne nový produkt, ale napriek tomu sa teší veľkej obľube. V kombinácii so Stateflow tvoria silný nástroj na modelovanie udalostných systémov. Doplnením blokov zo Simulinku možno jednoducho modelovať komplikované hybridné systémy. Predstavitelia spoločnosti The MathWorks tvrdia, že neexistuje produkt, ktorý by dokázal takto efektívne kombinovať udalostné a spojité systémy v jednej simulačnej schéme. Jeho uplatnenie sa nájde v modelovaní počítačových sietí alebo výrobných liniek, kde sa stretávame s rozhodovacou logikou.

Prípadné otázky spojené s témou SimEvents alebo aj ďalšie otázky ohľadom systému Matlab môžete klást' na diskusnom fóre www.matlab.sk.

Článok vznikol s podporou APVV-99-045805.

Literatúra

- [1] The MathWorks, Inc., SimEvents – User Guide
- [2] The MathWorks, Inc., Sim Events – Getting started
- [3] http://www.mathworks.com/products/simevents/?s_cid=HP_FP_SL_SimEvents

**Ing. Martin Foltin, PhD.
Bc. Kamil Valigura
Ing. Michal Blaho**

**Slovenská technická univerzita v Bratislave
Fakulta elektrotechniky a informatiky
Ústav riadenia a priemyselnej informatiky
Ilkovičova 3, 812 19 Bratislava
e-mail: martin.foltin@syprin.sk
michal.blaho@syprin.sk**

