

T125 AUTOMATIZAVIMAS, ROBOTOTECHNIKA

Hibridinių valdymo sistemų modeliavimo Petri tinklais ypatumai

R. Kragynys

Valdymo technologijų katedra, Kauno technologijos universitetas,

Studentų g. 48-230, LT-51367 Kaunas; tel. +370-37-300253; el.p. ricardas.kragynys@stud.ktu.lt

S. Bartkevičius

Teorinės elektrotechnikos katedra, Kauno technologijos universitetas,

Studentų g. 48-230, LT-51367 Kaunas; tel. +370-37-300253; el.p. stanislovas.bartkevicius@ktu.lt

Įvadas

Gamybos sistemų kompiuterinis sujungimas, robotika, lanksčios gamybos sistemos (FMS – angl. *Flexible Manufacturing Systems*) – dažnas mokslinių tyrimų ir studijų objektas leidžiantis pasiekti geresnių našumo, patikimumo, lankstumo rezultatų. Tačiau nors FMS vis plačiau diegiamos, daugelis gamintojų dėl jų technologinio kompleksiško, pritaikymo, įgyvendinimo ir ypač integravimo kliūčių vis dar vengia šių pažangių technologijų [1].

Moderni gamybos sistema yra sudėtingos konfigūracijos sistema, sudaryta iš įvairių išteklių: mechanizmų (mašinos, staklės), transporto sistemų (robotai, transporteriai), žaliavų, įrankių ir žmonių. Kadangi išteklių riboti, todėl išskyla įvairių jų paskirstymo uždavinių. Pagrindiniai uždaviniai yra tokie:

- taip paskirstyti išteklius, kad visą darbų kompleksą būtų galima atlikti per trumpiausią laiką;
- rasti kiek mažiausia išteklių reikia visiems darbams atlikti per tam tikrą laiką.

Be optimaliai naudojamų išteklių, gamybos linijos t. p. turi būti pakankamai lanksčios bei mobilios persikirstant darbus (pvz., sustojus kuriai nors gamybos linijai (mašinai), darbus persikirstant kitoms linijoms (mašinoms)).

Šioje srityje ne iki galo išspręsti tokie klausimai bei su tuo susijusios problemos:

- aukšta FMS įgyvendinimo kaina, derinant sistemą sugaištama daug laiko;
- valdymo programos nepakankamai mobilios;
- ribotas gamybos sistemų eksploatacijos lankstumas.

Šios ir kitos problemos išplaukia iš to, kad dėl vyraujančio kompleksiško valdikliai panaudojami neefektyviai. Labiausiai pastebimos kompleksiško priežastys yra:

- didelė tokių gamybos sistemų įvykių bei su jais susijusios laiko skalės įvairovė;
- sistemų procesai iš prigimties yra stochastiniai;
- valdymo sistemos (VS) turi dirbti su ne iki galo determinuotais duomenimis ir pasižymėti adaptyvumu;
- lanksčios gamybos sistemos yra nedeterminuotos;

- diskretinėms sistemoms iš principo netinka taikyti daugumą gerai žinomų klasikinė analitinių metodų.

Didėjant skaičiavimo technikos pajėgumams, automatizacijos lygiui, sistemos kompleksiskumui ir pasiskirstymui gamyboje, vis plačiau nagrinėjami diskrečių įvykių (DE – angl. *Discrete Event*) valdikliai, tačiau nėra universalus valdiklių pritaikymo pramonės sistemose, pavyzdžiui; tokiose kaip FMS, gamybos proceso valdymas, eismo valdymas ir kt. Be to, dabar nagrinėjant šią problemą neatsižvelgiama į tai, kad diskretinė VS dalis turi būti fiziškai paskirstyta daugelyje kompiuterių, kad pasiektų tinkamą skaičiavimų našumą, reikalingą didelės apimties kompleksinei sistemai valdyti. Kadangi gamybos sistemos fiziškai gali būti pasiskirsčiusios, todėl valdymo struktūra taip pat turi būti paskirstyta [2].

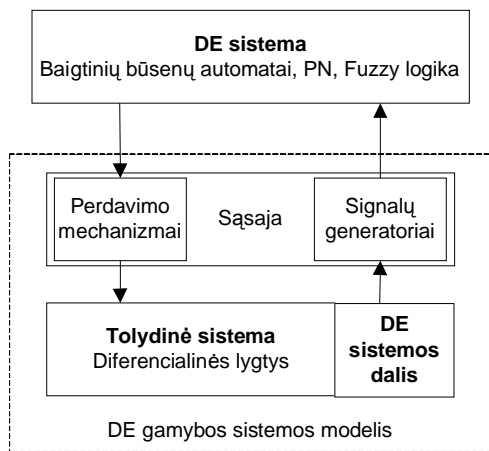
Siekiant iširti kuriamų sistemų dinamiką, našumą, aklaviečių nebuvimą, galimus konfliktus bei parinkti optimalų variantą, neapsieinama be modeliavimo. Sistemos modelis turi atitikti reikalavimus realiai kuriamai sistemai ir lengvai būti įgyvendinamas praktiškai.

Nagrinėjimo objektas būtų gamybos procesų ir išteklių modeliavimas, koordinavimas bei komunikacijų su fizine aplinka įgyvendinimas, siekiant fiziškai valdyti visą sistemą. Straipsnyje aptarsime modelio, kuris įgyvendinamas kaip VS, sudarymo Petri tinklais (PN – angl. *Petri Nets*) ypatumus.

Hibridinės valdymo sistemos modelio ypatybės

Viena iš galimybių tirti sudėtingas VS (o kūrimo stadijoje bene – vienintelė galimybė) yra modeliavimas. Tolydinius procesus, tokius kaip gamyba, cheminiai procesai, transporto sistemos ir kt., valdant skaičiavimo technika gaunama hibridinė VS – DE sistema, sujungiama su tolydine sistema [3], [4], [5]. Kiekvienas hibridinės sistemos sluoksnis modeliuojamas atskirai. Apatinis sluoksnis apima fizinę gamybos sistemą bei visus hibridinės VS tolydinius komponentus (žemo lygio valdiklius, laiko skaitiklius, modeliuojama sistemos plėtra laikui bėgant), kurie dažniausiai gali būti aprašomi diferencialinėmis lygtimis. Aukštesnysis sluoksnis yra DE sistema arba, kitaip tariant, DE valdiklis yra „įvykių generatorius“. Čia dažnai kaip aprašomoji kalbos sintaksė

pasirenkami baigtinių būsenų automatai, neraiški (angl. *Fuzzi*) logika. Hibridinė VS taip pat turi sąsają, kuri konvertuoja tolydinį signalą į įvykių sekas ir atvirkščiai, taip sujungdama tolydinę sistemą ir DE valdiklį (žr. 1 pav.).



1 pav. Hibridinės valdymo sistemos architektūra

Nagrinėjant sistemas šiuo metodu, sprendžiamos problemos: tolydinės sistemos sujungimas su diskretine, pavyzdžiui, baigtiniu automatu ar PN; saugumo problema – valdiklis turi užtikrinti, kad sistema nepatektų į pavojingą darbo režimą, pavyzdžiui, turi būti užtikrinta, kad du sąveikaujantys robotai nesusidurs; pasiekiamumo problema, kai VS turi užtikrinti gamybą nuo pradinės būsenos iki galutinės. Tačiau pažymėtina, kad, sudarant sistemų hibridinius modelius, dažnai nagrinėjama įvykių seka nesvarstant laiko problemas.

VS modeliavimo eiga bei naudojamos taisyklės priklauso nuo to, kokią modelį norime gauti. Galima skirti dvi modelių rūšis:

- imitacinius modelius;
- analitinius modelius.

Imitacinio modelio kūrimas, imitacija yra puiki priemonė sistemos funkcionavimui suprasti. Ji leidžia sukurti detalų sistemos modelį. Analitinis modelis yra abstraktesnis, formalus analizės metodas, kurio sprendinys yra matematinė išraiška, labiau tinkanti spręsti atskirų sistemos mazgų charakteristikoms nei realiai VS, kuri dažnu atveju gali būti per didelė, kad būtų galima gauti matematinę funkcionavimo išraišką. Kuriant imitacinį modelį išsprendžiami sistemos konfliktai, nustatomos ir išsprendžiamos aklavietės bei, naudojant analitinius modeliavimo įrankius, sudėliojami procesų prioritetai.

Imitaciniai ir analitiniai modeliai gali būti tiek deterministiniai, tiek tikimybiniai. Daugelio sistemų funkcionavimas yra deterministinis (pvz., paleidus du kartus tą pačią programą su tais pačiais įėjimo duomenimis gaunami tie patys išėjimo duomenys). Tačiau dažnai labai kompleksinių sistemų detalės nėra žinomos arba, jei žinomos, tai jų įvertinimas gali gerokai padidinti modelio kompleksumą, todėl yra daromos tikimybės prielaidos. Tikimybinis modelis dažnai yra palankesnis, nes galima gauti pakankamai tikslūs bendresnius rezultatus bei vertinti sistemos jautrumą kintant parametrams.

Svarbi modelio charakteristika yra sistemos elgsenos atvaizdavimas laiko skalėje. Realiai visi laiko matavimo prietaisai turi diskretinę laiko skalę, kuri sąlygoja baigtinį tikslumą. Dauguma modernių sistemų iš tikrųjų yra diskretinės ir naudoja diskretinę laiko skalę. Tuo tarpu modeliai dažnai naudoja tolydinę laiko skalę, todėl yra paprastesni. Jei laiko skalė būtų diskrecinė, tai modelyje reikėtų atsižvelgti į tai, kad tarp dviejų tolydinio laiko žymių galimi daugialypiai įvykiai. Esant tolydinei laiko skalei, užuot darius kokią nors tikimybinę prielaidą, galima fiksuoti visus įvykius ir tuo laiko momentu atsižvelgti tik į vieną įvykį. Taigi valdymo sistemų modeliuose natūralu nagrinėti stochastinius procesus tolydinėje laiko skalėje.

Stochastinis (atsitiktinis) procesas yra iš atsitiktinių dydžių šeimos $\{X(t) : t \in T\}$, aprašomas per nustatytą tikimybių erdvę, generuojantis vertes iš būsenų erdvės S ir indeksuojamas parametru t , kur t yra iš indeksų sekos T ; normaliai $T = [0, \infty]$. Taigi stochastinis procesas yra matematinis modelis, gebantis aprašyti tikimybinis įvykius kaip parametro, kuris nagrinėjamu atveju yra laikas, funkciją.

Funkcijos $X(t)$ priklausomybės nuo t grafikas yra vadinamas stochastinio proceso realizavimu. Atsitiktinio kintamojo $X(t)$ vertės vadinamos būsenomis S . Būsenų erdvė S gali būti diskretinė, t.y. baigtinė arba begalinė seka $S = \{s_1, s_2, \dots\}$, nors paprastai ji gali būti ir tolydinė. Valdymo uždaviniuose pagrindinis stochastinių procesų charakteristikos modelis – indeksų seka T vaizduoja sistemos raidą laikui bėgant. Indeksų seka t , p. gali būti diskretinė ir tolydinė (diskretinio ir tolydinio laiko stochastiniai procesai). Stochastinis procesas, kuris vyksta kaip tinklo būsenų procesas (stochastinių PN pagrindas), turi diskretinę būsenų erdvę ir nekintamą trajektorijos modelį.

Petri tinklai – valdymo uždavinių modeliavimo įrankis

Daugialypių išteklių ir konkuruojančių daugialypių darbo užduočių srautų kompleksinis darbas automatinės gamybos sistemas gali atvesti prie aklaviečių, kai bet kuris procesas bus nuslopintas. Tokie įvykiai gali sugadinti ar paralyžuoti visą sistemą, todėl daug dėmesio yra skiriama planavimo ir kontrolės metodams, kurie leidžia išvengti gamybos sistemos atvedimo prie aklaviečių. PN užima svarbią vietą kaip veiksmingas modeliavimo ir kiekybinės analizės tyrinėjimų įrankis, ypač tinkamas sistemose, kur vyrauja konkurencija, konfliktai ir sinchronizacija.

PN dažnai yra apibūdinamas tokiu rinkiniu (P, T, F, W, M_0) [6]. Čia $P = \{p_1, p_2, \dots, p_n\}$, $n > 0$ baigtinė pozicijų aibė; $T = \{t_1, t_2, \dots, t_s\}$, $s > 0$, baigtinė pereigų aibė; $F \subseteq (P \times T) \cup (T \times P)$ baigtinė netuščia įėjimų į pereigas ir išėjimo iš jų aibė. Įėjimų į pereigas $t \in T$ iš pozicijų aibė žymima $I(t)$, analogiškai išėjimų aibė $O(t)$. W apibrėžia jungčių $(p, t) \in F$ verčių (funkcijų) $w(p, t)$ aibę. M_0 – pradinis tinklo žymėjimas (tinklo būseną) iš tinklo žymėjimų multiaibės $M: P \rightarrow \mathbb{N}$ lemia kiekvienoje pozicijoje esantį žetonų skaičių (žetonų pasiskirstymą tinkle). $M_0(p)$ – žetonų skaičius pozicijoje p žymėjime M_0 . Dinaminė PN elgsena nusakoma dviem taisyklėmis:

- (*leidimo taisyklė*) pereiga $t \in T$ yra leidžiama žymėjime m , jei kiekviena įėjimų pozicija turi pakankamą skaičių žetonų, t. y. jei žymėjime $m \forall p \in I(t), M(p) \geq w(p, t)$. $E(m)$ yra aibė pereinamų, leidžiamų žymėjime m . Ne kiekviena $t \in E(m)$ gali įvykti;

- (*įvykdymo taisyklė*) pereigai įvykus žymėjime m , išimamas tam tikras skaičius žetonų iš įėjimo pozicijų ir įdedamas tam tikras skaičius žetonų į išėjimo pozicijas, taip sukuriama naują tinklo žymėjimą m' , t. y. $\forall p \in I(t) \cup O(t), m'(p) = m(p) - w(p, t) + w(t, p)$.

Dėl grafinės struktūros bei taikytinų matematinų analizės metodų gausos PN taikymai dažnai yra susiję su sistemos statinės struktūros modeliavimu, dinaminės elgsenos bei sistemos logikos funkcionavimo patikrinimu (imitavimu). Tai reiškia, kad nagrinėjame sistemos funkcionalumą bei dinamines ypatybes, kaip antai:

- ribų ypatybę – kiek daug (kiek mažai) tam tikros spalvos žetonų mes galime turėti toje pozicijoje;
- grįžtamumo ypatybę – žymė, į kurią visuomet yra galimybė sugrįžti;
- gyvybingumo ypatybę – iš kiekvienos pasiekiamos žymės egzistuoja įvykių seka.

Šias ypatybes galima patikrinti naudojant formalius PN analizės metodus, tokius kaip įvykių grafai, pozicijų bei pereinamų invariantai, redukcijos taisyklės ir k. t.

Kad būtų galima nagrinėti sistemos charakteristikas (pvz., maksimalų laiką, reikalingą atitinkamai operacijai atlikti, ar pareikalavimo laukimo trukmę), kurios yra aktualios valdymo uždaviniuose, reikia praplėsti modelį laiko ir/arba tikimybės sąvoka [6].

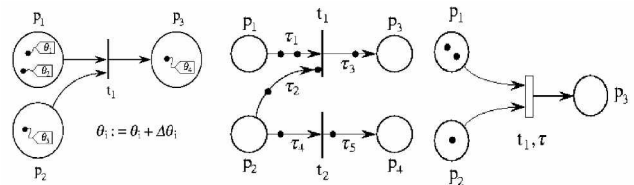
Laiko dislokacija Petri tinkluose

Dauguma laiko išplėtimų naudoja globalų laiką (kuris gali būti realaus arba diskretinio tipo), ir tokiu atveju jau galima kalbėti apie laiko intervalą, kuriame egzistuoja atitinkama tinklo būseną. PN su laiko išplėtimu procesas prasideda laiko momentu τ ir baigiasi laiko momentu $\tau + \theta > 0$ kai visos pereigos įvyko bent po vieną kartą. $\forall p_i \in P, \forall \tau: 0 \leq \tau \leq \theta, M[p_i](\tau)$ lemia žetonų skaičių pozicijoje p_i laiko momentu τ .

Laikas PN gali būti įvertintas įvairiai (žr. 2 pav.). Viena iš galimybių – įgyvendinti laiką kaip žetono atributą. Žetonas turi laiko žymę, kuri parodo, po kurio laiko žetonas yra leidžiamas pereigoje, kai jis ją pasiekia. Ši laiko žymė gali būti nustatyta kiekvienai pereigai atskirai. Toks laiko vertinimas dažnai pasirenkamas sprendžiant transporto uždavinius. Taip pat yra galimybė turėti vėlinimą tarp žetonų pašalinimo iš įėjimų bei žetonų įdėjimo į išėjimus (*šakos atributas*). Kelio laikas asocijuojamas su kiekviena šaka; žetonus leidžiama naudoti tik tada, kai jie pasiekia pereigą. Dažniausiai pasitaikanti laiko realizacija tinkle yra vėlinimas, sąlygojamas pereigos (*pereigos atributas*) [7].

Visais paminėtais trimis atvejais vėlinimas gali priklausyti nuo spalvos (duomenų, kuriuos teikia žetonas) tipo ir gali būti aprašytas kaip:

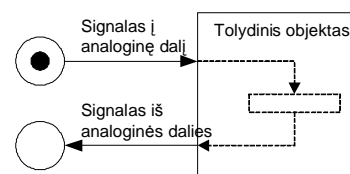
- nekintamas dydis (konstanta);
- dydis, parenkamas atsitiktinai iš nustatyto intervalo;
- dydis, nustatomas tikimybinio (neigiamu eksponentiniu) pasiskirstymu.



2 pav. Laiko įvertinimas klasikiniame Petri tinkle kaip žetono, šakos ar pereigos atributas

PN tinklai, turintys laiko žymę, nustatomą tikta tikimybinio būdu, dar vadinami stochastiniais PN (SPN). Jei į SPN įtrauksime ir momentines pereigas, tai tokie tinklai bus vadinami apibendrintais SPN (GSPN – angl. *Generalised SPN*).

Modeliuojant hibridines VS, didelis dėmesys skiriamas sąsajai, kuri tolydinio laiko signalą verčia į įvykių sekas ir atvirkščiai. Taip modeliuojamas dinaminis sistemos funkcionavimas. Yra darbų, kur naudojantis PN pagalba modeliuojami tiek tolydinis, tiek diskretinis objektai. Tuomet sujungus sąsają ir tolydinį gamybos modelį, gaunamas bendras DE gamybos sistemos modelis (žr. 1 pav.). Modeliuojant tokias sistemas Petri tinklais, reikalingas signalas iš tinklo, kuris inicijuoja kokį nors procesą. Procesas savo ruožtu taip pat turi generuoti signalą, kuris pasiunčiamas atgal į tinklą (žr. 3 pav.). Taigi tolydinio objekto procesą galima vertinti kaip PN pereigą, kuriai būdingas vėlinimas. Pavyzdžiui, [8] tokiai pereigai modeliuoti naudojama speciali pereiga – „procesas“. „Procesu“ galima modeliuoti įvairaus kito signalus. Bendruoju atveju modeliuojama struktūriniais blokais ir taip gaunamas diferencialinių lygčių atvaizdas. Tokiu atveju „procesas“ gali būti laikomas pereiga, kurios vėlinimas nėra apibrėžtas iš anksto. Šiuo principu naudojant PN atskirai modeliuojamas valdiklis (diskretiniai signalai) ir objektas. Gautas modelis atspindėtų realios sistemos struktūrą, t. p. būtų nesunkiai įgyvendinamas praktiškai.



3 pav. Petri tinklo ryšys su tolydiniu objektu

Taigi valdymo uždaviniuose natūraliausia būtų laiką PN modelyje susieti su pereiga. Pereigos atvaizduoja procesą (veiklumą), o procesas savo ruožtu suteikia laiką. Daugumą PN taikymų naudoja kaip tik tokį laiko vertinimą ir tik keletas modelių laiko vėlinimą yra susiję su pozicija ir/arba šakomis. Aukšto lygio PN su spalvotais žetonais (žymintais atitinkamą duomenų tipą) sprendžiant transporto uždavinius dažnai laiko žymė susiejama su žetonu [6], [9]. Laiko žymė žetone rodo laiką, kada jį bus leidžiama naudoti. Šis laikas, pateiktas žetono, yra aktyvizuojamas pereigoje, t. y. laikas yra sąlygojamas pereigos. Modeliavimo pakete CENTAURUS-C, kuriame yra realizuotas spalvotų PN (CPN – angl. *Coloured PN*) modeliavimas, laikas yra globalus dydis, apibūdinamas

kintamuoju \$Time, kurio vertę vartotojas gali tik skaityti. Laiko inkrementą sukelia aktyvūs pereinimų vėlinimai, procesai arba kintamąjį \$Time turintys pereinimų ar šakų reiškiniai, jeigu jie yra [8].

Konkurencinis pereinimų aktyvinimas

Automatinės gamybos sistemos susideda iš įvairių rūšių išteklių kaip antai mašinų, robotų, transporterių ir t. t. Puslaidininkinių įtaisų gamyboje įrangą gali sudaryti daugiau kaip šimtas skirtingų ar tų pačių mašinų. Tokios didelės apimties pramoninės sistemos gali prisiderinti prie įvairių prioritetus turinčių darbų bei gali perstruktūrizuoti darbą jei mašinos sugenda ar yra sustabdomos planiniam remontui. FMS užduoties darbas gali turėti alternatyvų kelią. Kelio pasirinkimo lankstumo nauda yra tai, jog didėja sistemos pralaidumas ir mašinų apkrovimas. Kita vertus didėja automatinės gamybos sistemos planavimo ir kontrolės kompleksiskumas, vyrauja nuolatinė konkurencija, kur procesai konkuruoja dėl palankesnio gamybos maršruto, mašinų, mašinos konkuruoja dėl įrankių, transporto sistemų, robotų ir t. t.

Pereiga yra leidžiama klasikinio PN žymėjime, jei kiekviena pereinimo įėjimo pozicija turi pakankamai žetonų (paprastai 1). Tik leidžiama pereiga gali įvykti. Leidžiamose pereinimuose yra konfliktas, jeigu jos dalijasi įėjimo pozicijas. Konfliktas klasikiniuose PN dažniausiai sprendžiamas keičiant tinklo struktūrą. Įvedus laiką į PN, būtina apibrėžti pereinimo „leidimo“ ir „įvykdymo“ taisykles (kuris momentas sukelia laiko inkrementą), kurios sąlygoja konflikto sprendimo būdą.

Literatūroje esama dviejų tipų laikinių PN semantikos: „varžybų“ ir „perrinkimo“ semantikos [9], [10]. Modeliuojant abiem atvejais galima pasiekti tą patį rezultatą, tačiau jei sukurtas modelis turi dalyvauti realioje sistemoje, pavyzdžiui, VS, reikia įvertinti kiekvienos semantikos niuansus. Dauguma autorių, kurdami SPN modelius, naudoja „varžybų“ semantiką. Tokie SPN yra patogūs kokybiniam modeliui rodikliams nagrinėti, nes turi tiesioginį ryšį su Markovo grandinėmis. „Varžybų“ semantikos atveju „įvykdymas“ yra momentinis, t. y. laikas susiejamas su pereinimo „leidimu“. Esant konfliktinei situacijai, paleidžiamos visos konflikte dalyvaujančios (leidžiamos) pereinimo (procesai) ir „įvyksta“ ta pereiga, kurios delsa yra mažiausia. Likusių pereinimų laiko (kartu ir proceso prioriteto) statusas, atsižvelgiant į modeliuojamą sistemą, interpretuojamas įvairiai. Tokia laiko „leidimo“ ir „įvykdymo“ semantika tinka modeliuojant ir realizuojant, pavyzdžiui, duomenų perdavimo protokolus, kuriuose, esant kolizijoms ir konfliktams, duomenys gali būti „atmetami“. Realiose VS paleistas procesas negali būti atmestas, todėl sudarant sistemų modelius, pagal kuriuos įgyvendinamos realios sistemos, konfliktas turi būti išspręstas dar prieš atsidarant pereinimams, t. y. nedalyvaujant laikui. Tai pasiekama naudojant modelyje „perrinkimo“ semantiką. Čia konfliktas sprendžiamas „leidimo“ laiko momentu (atsitiktiniu perrinkimu ar prioritetais) ir tik tai atrinktos pereinimo paleidžia procesus. Jei atrinkta pereiga įvyksta, ji išima žetonus iš $I(t)$ (įvykdymo pradžios įvykis), „paslepia“ žetonus įvykdymo periodu, paskui žetonai įdedami į $O(t)$ (įvykdymo pabaigos įvykis). Taigi vėlinimą sukelia užvaldos trukmė (angl. *holding time*), t. y. žetonas

užtrunka pereinimo laukdamas, kol ji įvyks. Naudojant perrinkimo semantiką modeliavimo metu panaikinami neapibrėžtumai ir išsprendžiami visi konfliktai.

Stochastinių ir apibendrintų stochastinių Petri tinklų analizavimo metodai

SPN ir GSPN vykstantys procesai yra stochastiniai ir tam tikrais atvejais gali būti panaudoti jau žinomų stochastinių procesų analizavimo algoritmai. Pateiksime vieną iš tokių stochastinių procesų ir jų analizavimo algoritmą.

Atsitiktinio kintamojo X tikimybinis aprašymas gaunamas per tikimybinę pasiskirstymo funkciją (PDF – angl. *Probability Density Function*):

$$f_X(x) = (d/dx)P\{X \leq x\}, \quad (1)$$

$$-\infty < x < \infty$$

Norint gauti atsitiktinio proceso tikimybinį aprašymą, reikia sujungti procese dalyvaujančių atsitiktinių kintamųjų PDF:

$$P\{X(t_1) \leq x_1, X(t_2) \leq x_2, \dots, X(t_n) \leq x_n\} \quad (2)$$

Bendruoju atveju atsitiktinio proceso pilnutinis tikimybinis apibūdinimas yra sunkiai įveikiama užduotis. Kuriant analitinius modelius svarbios yra atskiros stochastinių procesų grupės, kurių tikimybinis apibūdinimas yra paprastas. Vieną iš tokių klasių sudaro Markovo procesai.

Tarkim, $\{X(t) : t \in T\}$ yra stochastinis procesas su baigtine, begaline arba tolydine būsenų erdve S . $X(t)$ vadinamas Markovo procesu, jei bet kokiai laiko taškų $t_0 < t_1 < t_2 < \dots < t_n < t$ sekai, $X(t)$ pasiskirstymo priklausomybė nuo verčių $X(t_0), X(t_1), X(t_2), \dots, X(t_n)$ priklauso tik nuo $X(t_n)$. T. y. Markovo procesas yra toks stochastinis procesas, kur bet kuriame fiksuotame laiko taške t_n būsimą vertę priklauso tik nuo vertės tame laiko taške ir nepriklauso nuo prieš tai buvusių. Šios savybės yra vadinamos Markovo savybėmis.

Markovo procesai gali būti įvairių tipų priklausomai nuo to, kaip juose yra interpretuojamos būsenos, laikas, procesai (įvykiai).

Tarkim, kad vektorinis atsitiktinis dydis diskretiniais laiko sekų momentais aprašomas dydžiais $x[k]$, $k=0, 1, 2, \dots$. Kiekvienas iš vektorių turi neperraikiamą tikimybių seką ir neperraikiamas yra bendras tikimybių pasiskirstymas $p[x[k], x[k-1], \dots, x[0]]$.

Seka $x[0], x[1], \dots, x[k], \dots$ yra „balta“, jeigu tikimybių $x[k]$ pasiskirstymas nepriklauso nuo ankstesnių sekos narių, t. y.

$$p[x[k] | x[k-1], \dots, x[0]] = P[x[k]]. \quad (3)$$

Seka $x[0], x[1], \dots, x[k], \dots$ vadinama Markovo (arba paprasta Markovo) seka, jeigu kiekvieno sekos nario tikimybinis pasiskirstymas priklauso tik nuo prieš tai buvusio nario vertės, t. y.

$$p[x[k] | x[k-1], \dots, x[0]] = P[x[k] | x[k-1]]. \quad (4)$$

Šią tikimybę galima nagrinėti kaip perėjimo iš pozicijos $x[k-1]$ į poziciją $x[k]$ tikimybę, kai ji priklauso tikrai nuo šių narių.

SPN veikia tolydinio laiko srityje, kur kiekvienas vėlinimas aprašomas PDF. Jei modelyje vyrauja vėlinimai tik su neigiama eksponentine PDF bei naudojama varžybų semantika, tai tokį modelį galima nesunkiai keisti į Markovo grandines.

Markovo grandinėmis (MG) vadinami tokie Markovo procesai, kurių būsenų erdvė $x[k]$, $k=0, 1, 2 \dots$ yra apskaičiuojama. Šio kiekio elementus galima pažymėti $x^{<v>}[k]$, kur v sveikasis skaičius. MG tikimybių pasiskirstymas yra diskretinis. Perėjimo iš būsenos $x^{<\mu>}[k-1]$ į būseną $x^{<v>}[k]$ tikimybė apibūdinama perėjimo tikimybių matrica:

$$p[x^{<v>}[k] | x^{<\mu>}[k-1]]. \quad (5)$$

Jei parametras t yra tolydinis, tai procesas yra tolydinio laiko Markovo grandinė (CTMC – angl. *Continuous-Time Markov Chain*).

Tolydaus laiko Markovo grandinės su baigtine arba begaline būsenų erdve yra iš atsitiktinių dydžių šeimos $\{X(t) : t \geq 0\}$, turi diskretinę būsenų erdvę S , nekintantį trajektorijos modelį ir atitinka Markovo ypatybę:

$$P\{X(t+u) = j | X(v) : 0 \leq v \leq t, X(t) = i\} \quad (6)$$

$$= P\{X(t+u) = j | X(t) = i\}$$

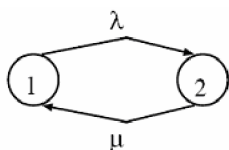
kur visi $i, j \in S$ ir $t, u \geq 0$.

CTMC atveju laiko vėlinimas yra atsitiktinis dydis su neigiama eksponentine tikimybės pasiskirstymo funkcija

$$f_x(x) = \mu e^{-\mu x} u(x), \quad (7)$$

kur $u(x)$ yra žingsnio funkcija ($u(x)=0$, kai $t < 0$, ir $u(x)=1$, kai $t \geq 0$), ir μ yra PDF parametras.

Praktikoje CTMC yra aprašomas būsenų perėjimų parametrų diagrama ir/arba būsenų perėjimų tikimybių matrica Q , kur elementai pagrindinės matricos įstrižainės (diagonalės) išorėje yra perėjimų iš būsenos į būseną eksponentinio pasiskirstymo parametrai, o įstrižainėje yra kiekvienos eilutės elementų suma. 4 pav. pavaizduota dviejų būsenų CTMC perėjimų parametrų diagrama. Pirmos būsenos vidutinis vėlinimas yra λ^{-1} , o antros būsenos - μ^{-1} . Plačiau apie MG panaudojimą rašoma [11], [12], [13].



4 pav. Markovo grandinės modelio pavyzdys

Šiuo atveju būsenų perėjimų tikimybių matricą gauname tokią:

$$Q = \begin{bmatrix} -\lambda & \lambda \\ \mu & -\mu \end{bmatrix}. \quad (8)$$

Norint gauti analitinę modelio išraišką, svarbu nustatyti, ar procesai, vykstantys SPN modelyje, yra Markovo procesai. Klasikiniuose PN, kur ryšiai tarp pozicijų apibūdinami fiksuota matrica, vykstantys procesai yra Markovo procesai, jeigu pereinimo procese dalyvauja atsitiktiniai dydžiai. Jeigu yra konfliktų, koks bus vyraujantis procesas, priklauso ir nuo konflikto sprendimo būdo (varžybų ar perrinkimo semantikos). Modeliuojant VS išskyla reikalavimai, kurie ne visada yra palankūs norint gauti analitinę modelio išraišką. CPN, be sujungimų matricos, egzistuoja dar ir šakų, pereinimo reiškinų, kurie gali būti sąlyginiai. Norint nustatyti ar čia vykstantys procesai yra Markovo ir ar gali būti nagrinėjami naudojant MG, reikalingos papildomos studijos. Tiek SPN, tiek GSPN yra aukšto lygio modeliavimo įrankis. Čia įėjimo duomenys gali būti įtraukti tiesiai iš objekto arba iš kitomis priemonėmis sumodeliuoto mazgo, pvz. eilių tinklo (QN – angl. *Queueing Networks*), kur MG yra plačiai naudojamas matematinis įrankis. Bendruoju atveju modelio imitavimas gali būti universali priemonė kurti VS modeliams – valdymo algoritmams, todėl svarbu turėti tam skirtą įrankį bei tam tikslui sukurtą taisyklių bazę.

Išvados

Apžvelgus hibridinių VS modeliavimo Petri tinklais ypatumus, galima daryti tokias išvadas:

1. Praplėtus PN laiko bei tikimybės sąvoką, paprastėja hibridinių sistemų ryšio tarp diskretinės VS ir tolydinio objekto modeliavimas. Gautas sistemos modelis atspindi ir įvertinta tiek diskretinį valdymo algoritmą, tiek tolydinį objektą.

2. Tinkamai parinkus modeliavimo taisykles (laiko tipą ir dislokaciją tinkle, pereinimo leidimo įvykdymo semantikas), įgalinančios funkcionuoti sąsają su objektu, galima modelį sujungti su realia sistema. Toks modeliavimo principas gali padėti sudaryti valdymo algoritmus.

Literatūra

1. **Kaighobadi M., Venkatesh K.** Flexible Manufacturing Systems: An Overview// International Journal of Operations & Production Management: - 1994. - Vol.14, No.4. - P. 26-49.
2. **Gonzalez F.** Real-Time Control of Distributed Large Scale Flexible Manufacturing Systems// NSF Design, Service and Manufacturing Grantees and Research Conference/SMU, Award Number: DMI-0200375. - Dallas, Texas, 2004. - 17 p.
3. **Koutsoukos X. D., Antsaklis P. J., Lemmon M. D., Stiver J. A.** Supervisory Control of Hybrid Systems// Proc. of the IEEE. – 2000. - 51 p.
4. **Jalilvand A., Khanmohammadi S.** Modeling and Control of a Manufacturing System as a Hybrid System using Petri Net// IJSIT Lecture Notes of 1st

- International Conference on Informatics. – September 2004. - Vol.1, No.2. - P. 29-34.
5. **Tilbury D. M., Khargonekar P. P.** Discrete Event Control of Manufacturing Systems// The Mechanical systems design handbook: modeling, measurement, and control. ISBN 0-8493-8596-2. - Edited by Osita D.I. Nwokah. - Yildirim Hurmuzlu. - 2002. - P. 55-75.
 6. **Jensen K.** Coloured Petri Nets: basic concepts, analysis methods and practical use. - Germany, 1997. - Vol.1. - P.123-153.
 7. **Gianfranco B.** An Introduction to Generalised Stochastic Petri Nets// 21st International Conference on Application and Theory of Petri Nets. - Aarhus, Denmark, June 26-30, 2000. - P. 217-266.
 8. **Bartkevičius S., Mačerauskas V., Šarkauskas K.** Spalvotųjų Petri tinklų taikymas valdymo sistemoms modeliuoti// Elektronika ir elektrotechnika – Kaunas: Technologija, 2003. - Nr.4(46). - P. 7-11.
 9. **Van der Aalst W.M.P., Van Hee K.M. and Reijers H.A.** Analysis of Discrete-time Stochastic Petri Nets// Statistica Neerlandica. – 2000. - Vol.54, No.2. - P. 237-255.
 10. **Aimone Marsan M., Bablo G., Bobbio A., etc.** The Effect of Execution Policies on the Semantics and Analysis of Stochastic Petri Nets// IEEE Transactions on Software Engineering. – July 1989. - Vol.15, No.7. - P. 832-845.
 11. **Lindemann Ch.** Performance Modelling with Deterministic and Stochastic Petri Nets// GMD Research Institute FIRST. - Berlin, Germany, 1998. - 405 p.
 12. **Ajmone Marsan M., Bobbio A., Donatelli S.** Petri nets in performance analysis: an introduction, Lecture Notes in Computer Science// Lectures on Petri Nets I: Basic Models. - Springer-Verlag, 1998. - Vol.1491. - P. 211-256.
 13. **Belch G., Greiner S., Meer H., Trivedi K.S.** Queueing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications. ISBN 0-471-20058-1. - John Wiley & Sons, Inc., 605 Third Avenue, New York, 1998. - 745 p.

Pateikta spaudai 2005 03 15

R. Kragnys, S. Bartkevičius. Hibridinių valdymo sistemų modeliavimo Petri tinklais ypatumai // Elektronika ir elektrotechnika. –Kaunas: Technologija, 2005. Nr. 6(62). – P.82-87.

Siekiant įveikti šiuolaikinių valdymo sistemų kompleksiskumą bei kurti „saugius“ valdymo algoritmus, reikalingas tinkamas modeliavimo įrankis bei tam tikslui sukurta taisyklių bazė. Valdymo algoritmas ir valdomas procesas sudaro hibridinę valdymo sistemą. Šiame straipsnyje pateikti hibridinių valdymo sistemų modeliavimo ir valdymo algoritmo į kurį taip pat įtrauktas ir tolydinis objektas, sudarymo laikiniais Petri tinklais ypatumai. Vertinant valdymo sistemų modeliavimo specifiką, apžvelgtos laiko dislokacijos ir realizavimo laikiniuose Petri tinkluose būdai. Pateikti modeliavimo principai įgalinantys funkcionuoti sąsają su objektu. Toks modeliavimo principas gali padėti sudaryti valdymo algoritmus. Il. 4, bibl. 13 (lietuvių kalba; santraukos lietuvių, anglų ir rusų k.).

R.Kragnys, S. Bartkevicius. Peculiarity of Hybrid control systems modeling using Petri Nets // Electronics and Electrical Engineering. – Kaunas: Technologija, 2005. – No. 6(62). – P.82-87.

With intentions to remove today's difficulties in control systems and to create "safe" algorithms of control it is necessary to have the suitable tool of modeling and the created base of rules. The algorithm of control and operated process compose a hybrid control system. In this paper the particularities of modeling of hybrid control systems and creation of algorithms of control with timed Petri nets are presented. The continuous object is included also in presented nets. The dislocation and realization of time in timed Petri nets are reviewed considering the specific of control systems modeling. Principles of modeling which influences in functioning of interface with object are presented. These principles of modeling could be used for composition algorithm of control. Ill. 4, bibl. 13 (in Lithuanian; summaries in Lithuanian, English, Russian)

Р. Крагнис, С. Барткавичюс. Особенности моделирования гибридных систем управления с Петри сетями // Электроника и электротехника. – Каунас: Технология, 2005. – № 6(62). – С.82-87.

Чтобы удалить сегодняшние трудности в системах управления и создать «безопасные» алгоритмы управления необходимо иметь тому пригодный инструмент моделирования и созданную базу правил. Алгоритм управления и управляемый процесс составляют гибридную систему управления. В этой работе представлены особенности моделирования гибридных систем управления и создание алгоритмов управления с временными Петри сетями, в которые в том числе включен и непрерывный объект. В связи с особенностями моделирования систем управления рассмотрены варианты расположения и реализация времени во временных Петри сетях. Представлены принципы моделирования, которые влияют на функционирование интерфейса с объектом. Такой принцип моделирования может быть использован для создания алгоритмов управления. Ил. 4, библи. 13 (на литовском языке; рефераты на литовском, английском и русском яз.).