

Lanksčios linijos valdymo optimizavimas

S. Bartkevičius

Teorinės elektrotechnikos katedra, Kauno technologijos universitetas,
Studentų g. 48, LT-51367 Kaunas, Lietuva; tel. +370 37 300253, el. p. Stanislovas.Bartkevicius@ktu.lt

J. Daunoras, K. Šarkauskas

Valdymo technologijų katedra, Kauno technologijos universitetas,
Studentų g. 48, LT-51367 Kaunas, Lietuva; tel. +370 37 300298, el. p.: Jonas.Daunoras@ktu.lt,
Kastytis.Sarkauskas@ktu.lt

Įvadas

Lanksčias gamybos sistemas (LGS) sudaro keletas gamybinių įrenginių (staklių) ir jas siejanti transportavimo sistema. Modernias transportavimo sistemas dažnai sudaro robotai, kurie ne tik perneša gaminius ar ruošinius iš vieno įrenginio į kitą, bet ir atlieka kitokias staklių aptarnavimo funkcijas, pavyzdžiui, įdeda ruošinį į staklių griebtuvą, išima jau apdirbtą detalę. LGS „lankstumą“, t. y. gebėjimą gaminti įvairius gaminius, daugiausia lemia transportavimo sistemos ypatumai. Kitas svarbus LGS darbo aspektas yra jos panaudojimo efektyvumas, nes šios sistemos yra labai brangios ir, siekiant jų rentabilumo, būtina maksimaliai apkrauti visus įrenginius.

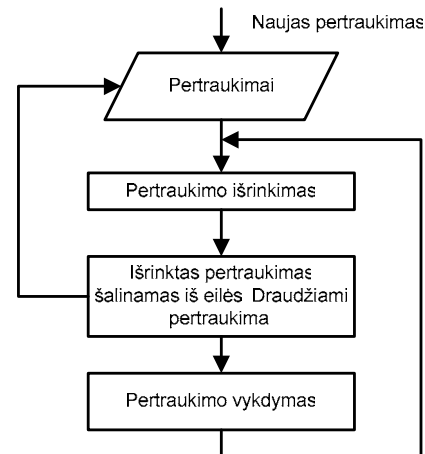
Didelė LGS valdymo sistemų problema yra vadinamos „aklavietės“, kai sistema sustoja, nes nebėra racionalaus tolesnių veiksmų pasirinkimo [1]. Todėl, siekiant išvengti tokių situacijų, atliekama išankstinė LGS valdymo sistemų analizė. Sudėtingų LGS analizinės beveik neįmanoma gauti, todėl plačiai taikomas modeliavimas [2]. Ypačingą vietą čia užima modeliavimas Petri tinklais [3,7]. Šitai analizuojant LGS valdymo sistemą, laikoma, kad valdymo procesas susideda iš skirtingus prioritetus turinčių įvykių (veiksmų). Prioritetų sistema iš dalies leidžia išvengti „aklavietės“, tačiau anaiptol negarantuoja optimalaus sistemos veikimo. LGS valdymo sistemoms projektuoti ir analizuoti, siekiant optimalaus valdymo, taikomas ir tikimybinis metodas, grįstas Markovo grandinėmis [4,8,9], kuris yra formalizuotas PEPA, TIPP ir kt. algebromis [5,6,10,11]. Tikimybinis požiūris grindžiamas tuo, kad šiuolaikinėse LGS naudojami įrenginiai dažnai turi adaptyvias valdymo sistemas, todėl jų atliekamų operacijų trukmė nėra griežtai fiksuota ir kinta priklausomai nuo gaminio parametru, t. y. sistemos įvykiai yra asinchroniniai. Taikant ir PEPA ar TIPP algebrą, taip pat išlieka fiksuotų prioritetų sistema. Iš dalies transportavimo sistemos valdymą galima optimizuoti taikant „atkaitinimo“ algoritmą [6], tačiau tam būtina žinoti būsimą veiksmų seką, kas vargu ar įmanoma sistemoje su asinchroniniais įvykiais.

Šio darbo tikslas – parodyti, kad taikant adaptyvias prioritetų sistemą galima iš esmės pagerinti LGS valdymą, gerokai padidinant įrenginių apkrovimą, ir aptarti roboti-

zuotos transportavimo sistemos adaptyvius valdymo algoritmus.

Funkcionavimo valdymas su pertraukimais

Racionalus valdymas grindžiamas pertraukimų režimu. Procesorinės sistemos turi pertraukimų mechanizmą, kuris funkcionuoja lygiagrečiai su procesoriumi, ir pagal atitinkamą reikalavimą pertraukdamas programos vykdymo seką, keičia valdymo programos darbą. Šiam valdymo būdai yra būdingi įtaisų, reikalaujančių pertraukimo, prioritetai. Aukštesnį prioritetą turintis įtaisas, pertraukia žemesnio prioritetą turintį valdymo programą, įrašydamas būtiną žemesnio prioritetą turinčios programos informaciją į dėklą, ir, baigęs pertraukimo aptarnavimą, grąžina informaciją iš dėklo ir toliau vykdo pertrauktą programą.



1 pav. Tradicinis valdymo algoritmas, kai išrenkamas aukščiausias prioritetą turintis pertraukimas, o naujai atsirandantys pertraukimai, kad ir kokio prioritetą jie būtų, įtraukiami į eilę

Jeigu pagal veikimo logiką vykdoma operacija negali būti pertraukiama, pertraukimai yra draudžiami. Roboto darbo specifiška kaip tik ir remiasi tokia valdymo logika, nes, leidus aukštesniojo prioritetą turinčiam pertraukimui pertraukti

jau vykdomą žemesnio prioriteto programą, galimi beprasmiškai roboto važinėjimui į vieną ir kitą pusę. Tokios sistemos funkcionavimo algoritmas parodytas 1 paveiksle.

Lanksčią gamybos liniją sudaro programinės staklės, kurias aptarnauja robotai ir manipulatoriai, gabendami ruošinius ir pagamintas detales tokiomis maršrutais: staklės – staklės, sandėlis – staklės, staklės– transporteris ir galbūt sandėlis – transporteris. Robotai ir manipulatoriai yra išdėstyti išilgai transporterio, ir kiekvienas robotas aptarnauja jam priskirtą darbo vietų (staklių) skaičių.

Adaptyvūs valdymo algoritmai

Modernių LGS atskirų įtaisų valdymo sistemos yra adaptyvios, siekiančios optimaliai apdoroti detalę pagal vieną ar keletą jos parametrų. Todėl detalės apdorojimo gamybiniame įtaise trukmė yra kintama ir griežtai nustatytas laiko atžvilgiu įtaisų aptarnavimo grafikas yra nežinomas, nes gamybos procesas darosi asinchroninis. Todėl norint, kad asinchroniškai veikiančios įtaisai būtų aptarnauti laiku, atsiranda optimalaus manipuliatorių valdymo problema.

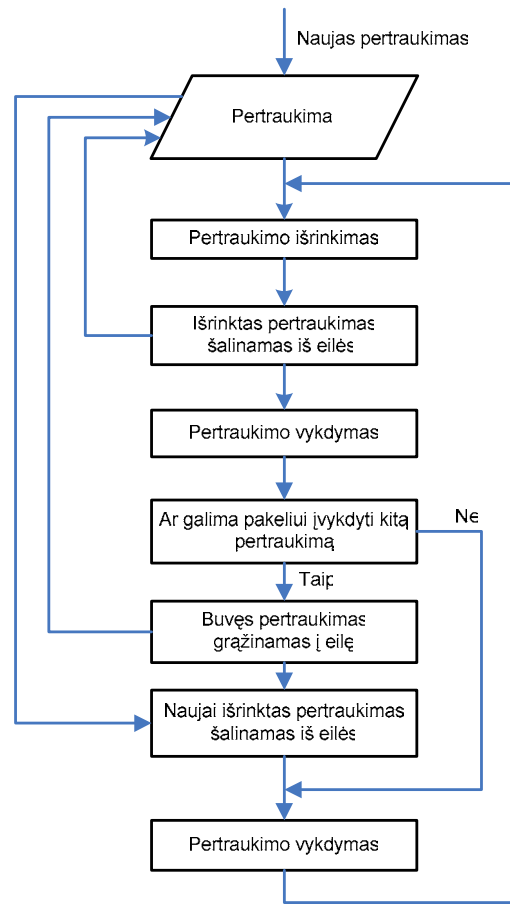
Komponuojant automatizuotą gamybinę liniją, įrenginiai renkami tuo principu, kad jie visi funkcionuotų nepertaukiamai. Tai savaime aišku, nes įrenginiai, kurių funkcionavimą stengiamasi visiškai automatizuoti, yra brangūs, todėl jie turi būti maksimaliai panaudojami. Mūsų paminėtu atveju galima parinkti greičiau funkcionuojantį robotą arba skirti jam aptarnauti mažiau darbo vietų, tačiau abiem atvejais sistema brangsta. Tačiau ir tai negelbsti situacijos, jei darbo vietose procesų trukmių diapazonas yra platus (elektronikos, mikroelektronikos įtaisų gamyba, metalo apdirbimo procesai ir pan.), nes tuomet brangūs greitaveikiai robotai dažnai „dykinėja“, lėtesni nespėja aptarnauti gamybos vietų. Todėl tenka taikstytis su nedidelėmis atskirų gamybos įtaisų prastovomis siekiant optimalaus visos sistemos veikimo. Šiuo atveju racionalus (optimalus) robotų valdymas tampa ypač svarbus.

Roboto funkcionavimas yra ypatingas tuo, kad robotas, vykdydamas operacijas, turi pakeisti savo vietą. Vieta paprastai pakeičiama tuščiomis ir santykiškai šiai operacijai sugaištama daugiau laiko nei operacijoms, kai dirba roboto manipulatoriai. Kai aptarnaujami procesai tampa asinchroniniai, robotas yra priverstas dažniau keisti savo pozicionavimo vietą, tuo pat metu neatlikdamas jokio darbinio veiksmo.

2 pav. pateiktame valdymo algoritme siūloma keisti pertraukimų aptarnavimo mechanizmą. Visi pertraukimai su savo prioritetais įtraukiami į eilę, o vykdymo tvarka analizuojama programiškai ir, jei robotas atlieka tuščią operaciją, žiūrima, ar naujai atsiradusį pertraukimą galima atlikti pakeliui. Jei tokia situacija yra galima, roboto vykdoma tuščia operacija nutraukiama ir toliau vykdoma naujai išrinkta operacija, o atšauktasis pertraukimas gali būti vykdomas dvejopai:

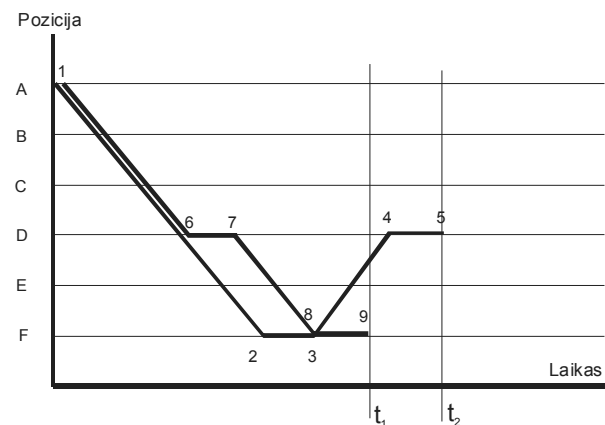
1) atšauktas pertraukimo reikalavimas grąžinamas į bendrą eilę ir toliau bus vykdomas pagal bendrus išrinkimo principus;

2) pertraukimas besąlygiškai bus vykdomas tuoj pat po to, kai bus įvykdyta jį pertraukusioji operacija.



2 pav. Pertraukimo vykdymo metu analizuojama, ar egzistuoja kitas procesas, kurį būtų galima pakeliui įvykdyti. Jei tokia situacija galima, jau vykdomas pertraukimas nutraukiamas, jis grąžinamas į bendrą eilę, o vykdomas naujai išrinktas

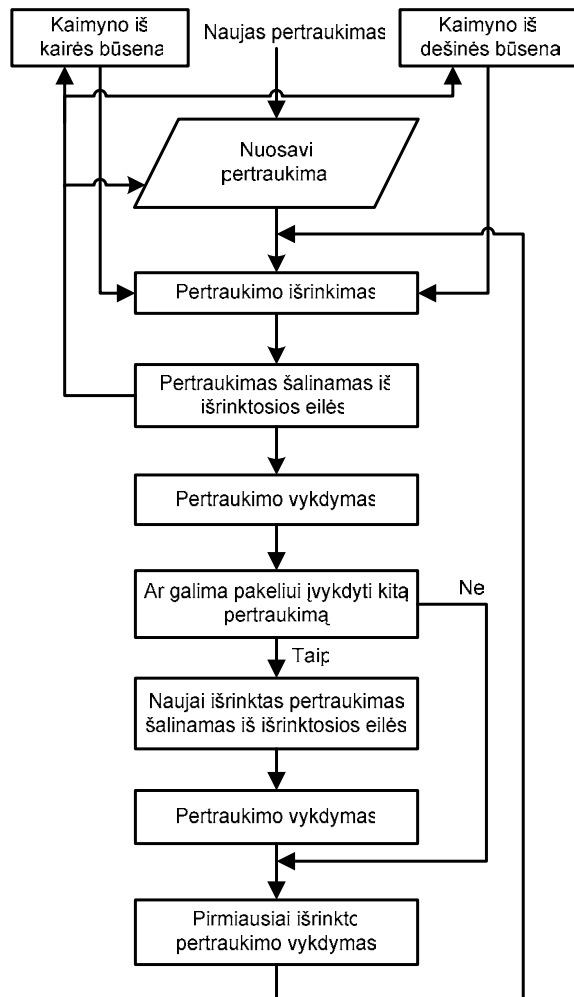
Algoritmo efektyvumo esmė yra ta, kad sudėjus drauge laiką, sugaištą šioms operacijoms atlikti, yra minimizuojamas roboto važinėjimas tuščiomis.



3 pav. Roboto aptarnaujančio pozicijas A-F, darbo ciklograma, rodanti laiko ekonomijos gavimo principą

Situacijos, kurios atsiranda aptarnaujant asinchroninius procesus, turi momentų, kada robotas, važiuodamas tuščias, galėtų aptarnauti tarpinį procesą, tam papildomai

negaišdamas daug laiko, tačiau klasikinis valdymo algoritmas to neleidžia. Nagrinėjame pavyzdyje robotas aptarnauja šešias pozicijas (3 pav.). Jei tuo pat metu reikalingi keli pertraukimai, išrinkimo logika, kaip įprasta, turi atitikti pasirinktus prioritetus. Tarkime, kad robotas yra A pozicijoje ir gauna užduotį aptarnauti F poziciją. Iki F pozicijos robotas važiuoja tuščias. Robotui jau važiuojant, atsiranda poreikis aptarnauti D poziciją. Klasikinio valdymo atveju naujai atsiradęs pertraukimas įtraukiamas į eilę, ir apie jo vykdymą bus sprendžiama kai bus užbaigtas dabar atliekamas pertraukimas. Taigi, ciklogramą atspindi trajektorija 1-2-3-4-5, kuriai realizuoti sugaištas laikas t_2 .

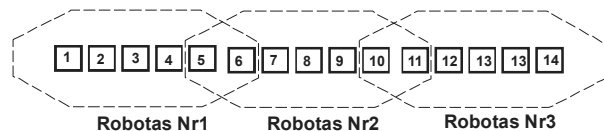


4 pav. Valdymo algoritmas, kai robotas turi savo aptarnavimo zoną, taip pat esant reikalui gali aptarnauti ir kraštines gretutinių robotų pozicijas, analizuodamas savo ir jų užimtumą

Jei bus valdoma pagal siūlomą algoritimą, naujai atsiradęs pertraukimas yra analizuojamas atsižvelgiant į esamą situaciją, o kadangi robotas važiuoja tuščiai, pakeliui yra aptarnaujama D pozicija ir toliau robotas tęsia savo judesį iki F pozicijos. Šiuo atveju ciklogramos trajektorija yra 1-6-7-8-9 ir bendrai sugaištas laikas yra t_1 . Matome, kad dviejų operacijų suminės vykdymo trukmės vienu ir kitu

atveju yra skirtingos, o naudojant siūlomą algoritimą, per dvi vykdomas operacijas sutaupoma laiko.

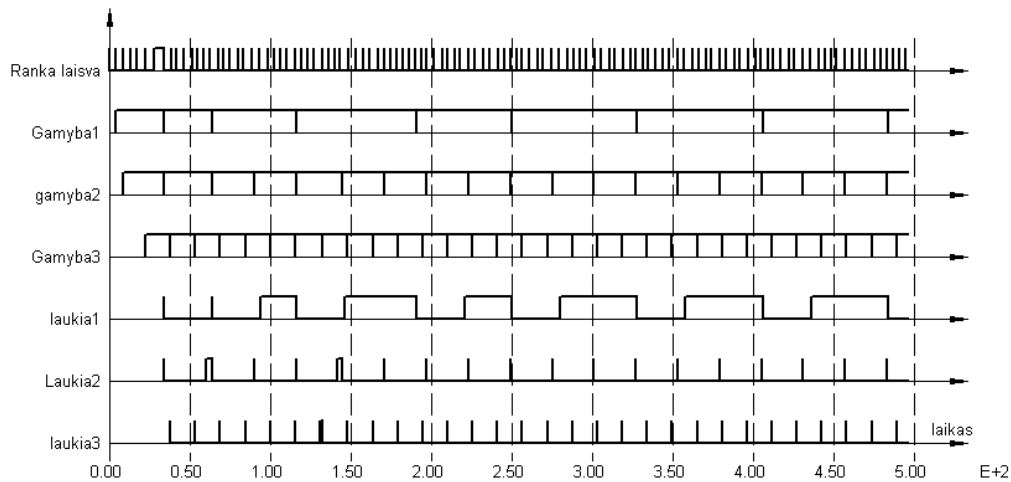
4 paveiksle pavaizduotas dar daugiau gamybinių išteklių taupantis valdymo algoritmas. Kai lanksčią liniją aptarnauja keli robotai, kiekvienam robotui yra priskirta individuali darbo zona su fiksuotu skaičiumi aptarnaujamų įrenginių. Kai įrenginių procesai yra adaptyvūs ir jų kreipimasis į valdymo įtaisą tampa asinchroninis, galima situacija, kai kuris nors robotas nebespėja aptarnauti visų į jį besikreipiančių įrenginių. Dėl to įrenginiai prastovi. Tačiau tuo pat metu jo kaimynai tokios problemos neturi, todėl kyla mintis, ar jie nors kiek negalėtų padėti. Galėtų, jeigu kitaip būtų suformuotos pradinės sąlygos. Jei 5 paveiksle robotui Nr.2 būtų priskiriama tik jo paties aptarnaujama zona (6-7-8-9-10 įrenginiai), po vieną kraštinių įrenginių būtų leidžiama aptarnauti kaimynams (robotui Nr.1 6-tąjį įrenginį, o robotui Nr.3 – 10-tąjį įrenginį), taip pat galima aptarnauti ir po vieną gretutinių kaimynų įrenginį (5-tąjį įrenginį, priklausantį robotui Nr.1, ir 11-tąjį įrenginį, priklausantį robotui Nr.3).



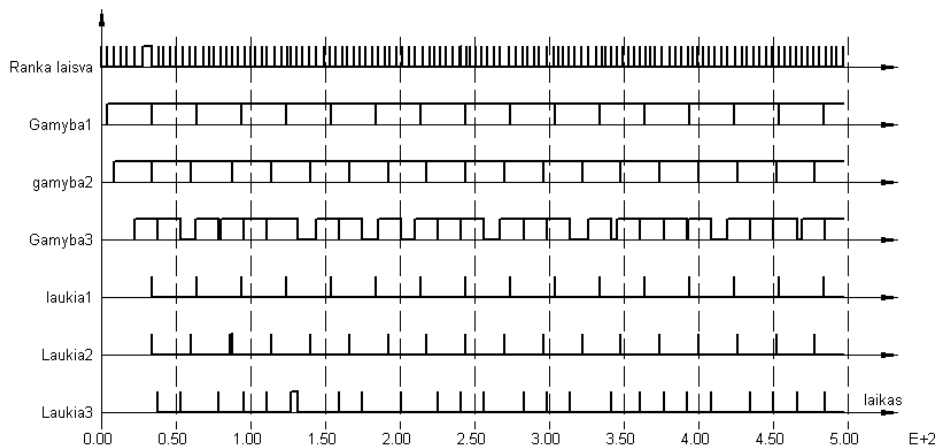
5 pav. Situacija, aiškinanti valdymo algoritimą, pateiktą 4 paveiksle, kai robotai gali aptarnauti ne tik jiems priskirtas pozicijas, bet ir kraštines gretutines kaimynų pozicijas. Robotui Nr.1 priskirti 1-5 įrenginiai, robotui Nr.2 – 6-10 įrenginiai, o robotui Nr.3 – 11-14 įrenginiai

Aprašyti funkcionavimo algoritmai buvo sumodeliuoti spalvotaisiais Petri tinklais, naudojant programų paketą CENTAURUS [11], sudaryti trys modeliai. Modeliavimo rezultatai pateikti atitinkamuose paveiksluose: 6 pav. – aptarnaujant klasikiniu pertraukimus veikiančiu algoritmu, 7 pav. – kai atšauktas pertraukimas grąžinamas į bendrą eilę, ir 8 pav. - kai atšauktas pertraukimas vykdomas tuoj pat po jį atšaukusio pertraukimo. Šiuose paveiksluose matome, kad įrenginių prastovos gali būti dviejų tipų. Pirmoji prastova atsiranda, dėl to, kad nėra kur dėti pagamintos detalės, nes neatlaisvintas dėklas su anksčiau pagaminta detale. Tai paveiksluose rodo diagramos su užrašu „laukia“. Čia yra laukiama, kada transportavimo robotas atlaisvins dėklą, kurį užima anksčiau pagaminta detalė. 6 paveiksle, kur pavaizduotas funkcionavimas pagal klasikinį algoritimą, t. y. paskutinis aptarnaujamas įrenginys yra tas, kuris turi žemiausią prioritetą, matome, kad prastovos susidaro procesui, turinčiam žemiausią prioritetą. Valdant pagal siūlomus algoritmus (7 ir 8 pav.) prastovų dėl šios priežasties nebelyka.

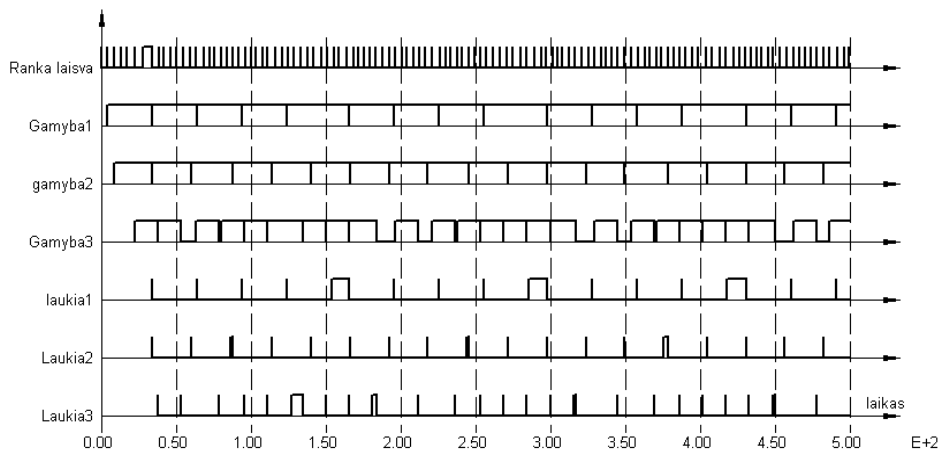
Kita prastovos situacija susidaro tuomet, kai detalė pagaminama, o naujo ruošinio ruošinių dėkle robotas dar nepristatęs. Tai paveiksluose rodo diagramos su užrašu „gamyba“. Valdant pagal klasikinį algoritimą (6 pav.) prastovos susidaro dėl to, kad laiku nepaimamos vieno įrenginio pagamintos detalės. Valdant pagal kitus algoritmus, prastovos pasiskirsto tarp įvairių procesų. Valdant pagal algoritimą, kai atšauktas pertraukimas grąžinamas į bendrą eilę, pradeda vyrauti prastovos dėl ne laiku pristatomų ruošinių.



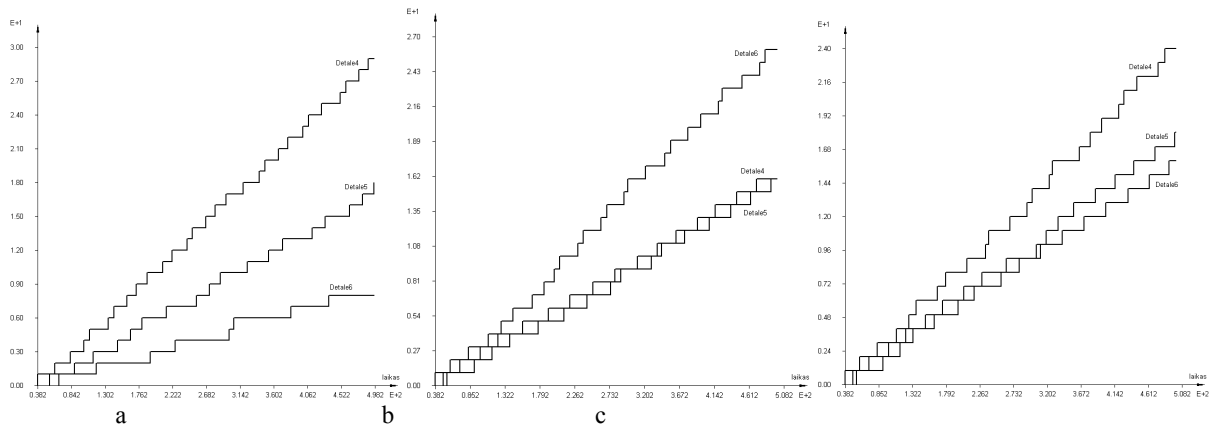
6 pav. Trijų įtaisų su adaptyviais valdymo procesais darbą aptarnaujančio roboto valdymas pagal klasikinį pertraukimų aptarnavimo algoritmą: Laukia – įrenginyje detalė pagaminta ir laukiama, kada bus atlaisvintas dėklas, Gamyba – įrenginys laisvas, bet ruošinys gamybai nepristatytas



7 pav. Trijų įtaisų su adaptyviais valdymo procesais darbą aptarnaujančio roboto valdymas kai atšauktas pertraukimas grąžinamas į bendrą eilę: Laukia – įrenginyje detalė pagaminta ir laukiama, kada bus atlaisvintas dėklas, Gamyba – įrenginys laisvas, bet ruošinys gamybai nepristatytas



8 pav. Trijų įtaisų su adaptyviais valdymo procesais darbą aptarnaujančio roboto valdymas, kai atšauktas pertraukimas vykdomas tuoj pat po jį atšaukusio pertraukimo: Laukia – įrenginyje detalė pagaminta ir laukiama, kada bus atlaisvintas lizdas, Gamyba – įrenginys laisvas, bet ruošinys gamybai nepristatytas



9 pav. Trijų įtaisų detalių pristatymo į sandėlį grafikai. a – valdant pagal klasikinį algoritmą, b – valdant pagal algoritmą, kai pertraukta operacija grąžinama į bendrą eilę, c – valdant pagal algoritmą, kai pertraukta operacija vykdoma tuoj pat po pertraukusios

Lygindami modelių su skirtingais valdymo algoritmais funkcionavimą, matome, kad, dirbant pagal klasikinį valdymo algoritmą, didžiausia yra pirmosios detalės prastova, nes šio įrenginio pagaminta detalė laiku nenunešama į sandėlį. Dirbant pagal algoritmą, kai atšauktas pertraukimas grąžinamas į bendrą eilę, prastovos gaunamos trečiajam įrenginiui, nes jam laiku nepristatomas ruošinys. Tuo tarpu dirbant pagal algoritmą, kai atšauktas pertraukimas vykdomas tuoj pat po jį atšaukusio pertraukimo, įrenginių prastovos pasiskirsto tolygiau visuose procesuose. Jeigu paanalizuosime, kaipgi yra su pagaminamų detalių skaičiumi, matysime, kad blogiausia situacija susidaro, kai dirbama pagal klasikinį algoritmą. Tai aiškiai matyti 9 paveiksle, a.

Išvados

Straipsnyje nagrinėtos adaptyvaus roboto funkcionavimo problemos, kai aptarnaujami procesai taip pat yra adaptyvūs. Modeliavimas parodė, kad būtina analizuoti ir vertinti naujai atsiradusio pertraukimo sukeltą situaciją ir priimti reikiamus sprendimus.

Naudojant siūlomi valdymo algoritmus nereikia jokių papildomų materialinių išlaidų, geriau panaudojami gamybiniai pajėgumai su tais pačiais transportavimo įtaiso resursais.

Kad algoritmų sėkmingai veiktų, reikia kitaip organizuoti pertraukimo mechanizmo funkcionavimą. Modeliavimas parodė, kad įrenginiai kur kas efektyviau panaudojami, kai valdoma analizuojant situacijas, kada, robotui vykdant užduotį, pakeliui galima atlikti kitas operacijas. Tai sudaro galimybės efektyviau panaudoti visus gamybos procese dalyvaujančius įtaisus.

Literatūra

1. **Lawley M., Revelotis S., Ferreira P.** Design guidelines for Deadlock Handling Strategies in Flexible Manufacturing Systems // International journal of Flexible Manufacturing Systems. – 1997. – No. 9(1). – P. 5-29.
2. **Bonder D.A., Revel Otis S.A.** Virtual Factories: An Object-Oriented Simulation Based Framework for Real-Time FMS

Control // IEEE intern. Conf. on Emerging Technologies and Factory Automation. – Los Angeles, 1997.

3. **Hong Sen Yan, Zeng Wang, Xiao Cheng Jiao.** Modeling, Scheduling and Simulation of Product Development Process by Extended Stochastic High-Level Petri Nets // Robotics and Computer Integrated Manufacturing. – 2003. – No. 19. – P. 329–342.
4. **Harrison P. G., Knottenbelt W.J.** Passage-time Distributions in Large Markov chains // Proceedings of ACM SIGMETRICS, June 2002, Marina Del Rey, USA. – P. 77–85.
5. **Bradley J. T., Daves N.J.** A Matrix based method for analyzing stochastic process algebras // ICALP Workshop 2000, Process algebra and Performance Modeling Workshop / University of Waterloo. – Ontario, Canada, 2000. – P. 579–590.
6. **Gilmore S, Hillston J., Holton D.R.W., and Rettelbach M.** Specifications in Stochastic Process Algebra for a Robot Control Problem // International Journal of Production Research. – 1996. – 34(4). – P. 1065–1080.
7. **Bartkevičius S., Šarkauskas K.** Programinis paketas CEN-TAURUS. Modeliavimas, identifikavimas, optimizavimas. Kaunas: Technologija, 2003. – 95 p.
8. **Gaved A., Elia G., Menga G.** Object Oriented Modelling of a Robotized Manufacturing Cell // The International Handbook on Robot Simulation Systems. – John Wiley & Sons, Ltd. London, 1993.
9. **Hermanns H., Herzog U., Katoen J.P.** Process Algebra for Performance Evaluation // Theoretical Computer Science. – 2002. – Vol. 247 (1-2). – P. 43–87.
10. **Bartkevičius S., Mačerauskas V., Šarkauskas K.** Sistemos su asinchroniniais procesais optimizavimas taikant spalvotus Petri tinklus // Elektronika ir elektrotechnika: Mokslo darbai. – Kaunas: Technologija, 2004. – Nr. 5(54). – P.60–65.
11. **Bartkevičius S.K., Mačerauskas V., Šarkauskas K.** Spalvotųjų Petri tinklų taikymas valdymo sistemoms modeliuoti // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2003. – Nr. 4(16). – P. 7–11.

Pateikta spaudai 2004 05 25

S. Bartkevičius, J. Daunoras, K. Šarkauskas. Lanksčios linijos valdymo optimizavimas // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2005. – Nr. 1(57). – P. 56–61.

Optimalus ar adaptyvus valdymas lanksčiose gamybos linijose, kurias aptarnauja robotai ir manipulatoriai, sukelia asinchroninius procesus. Atlikta analizė parodė, kad, esant asinchroniniams procesams, valdymas pagal prioritetus praranda savo efektyvumą dėl procesų sinchronizacijos sutrikimo.

Norint optimaliai panaudoti gamybinius pajėgumus, tenka keisti valdymo algoritmus, ką akivaizdžiai parodė modeliavimo rezultatai. Norint efektyviai aptarnauti sistemą su asinchroniniais procesais, reikia naudoti algoritmus, nuolat analizuojančius naujai susidariusias situacijas ir sprendžiančius: ar toliau tęsti vykdomą operaciją, ar ją pertraukti ir vykdyti naują, buvusią operaciją atšaukiant ir įtraukiant į bendrą eilę, ar besąlygiškai vykdyti po naujos operacijos įvykdymo. Il. 9, bibl. 11 (lietuvių kalba; santraukos lietuvių, anglų ir rusų k.).

S. Bartkevičius, J. Daunoras, K. Šarkauskas. Optimization of Control of a Flexible Manufacturing Line // Electronics and Electrical Engineering. – Kaunas: Technologija, 2005. – No. 1(57). – P. 56–61.

Optimal and adaptive control of a flexible manufacturing line served by robots and manipulators causes the appearance of asynchronous processes. The carried out analysis showed that control according to priorities loses its effectiveness because of the failure of synchronization.

Desiring to use production capacities optimally, it is necessary to change the algorithms of interruption service. This became obvious after the carried out simulation. To attend system with asynchronous processes effectively, it is necessary to use algorithms of interruption service analyzing situations and the assuming solution what operation must be carried out: to continue the performed operation or to recall it and to carry out new, returning recalled into general queue or unconditionally to carry out after the fulfillment of operation, which replaced it. Ill. 9, bibl. 11 (in Lithuanian; summaries in Lithuanian, English and Russian).

С. Барткевичюс, И. Даунорас, К. Шаркаускас. Оптимальное управление гибкой линией // Электроника и электротехника. – Каунас: Технология, 2005. – № 1(57). – P. 56–61.

Оптимальное и адаптивное управление гибкой линией вызывает появление асинхронных процессов в автоматических линиях, обслуживаемых роботами и манипуляторами. Проведенный анализ показал, что управление по приоритетам теряет свою эффективность из-за сбоя синхронизации.

Для оптимального использования производственных мощностей приходится менять алгоритмы управления, что стало очевидным после проведенного моделирования. Желая эффективно обслуживать систему с асинхронными процессами, надо использовать алгоритмы управления, анализирующие ситуации и принимающие решения, которую операцию надо проводить: продолжать выполняемую операцию или отозвать ее и выполнить новую, возвращая отозванную в общую очередь или безусловно выполнять после выполнения заменившей ее операции. Ил. 9, библи. 11 (на литовском языке; рефераты на литовском, английском и русском яз.).