

СИСТЕМНЫЙ АНАЛИЗ, УПРАВЛЕНИЕ И ОБРАБОТКА ИНФОРМАЦИИ

УДК 005.591.6

DOI: 10.24412/2071-6168-2021-4-322-330

СТРУКТУРНЫЙ АНАЛИЗ МОДЕЛЕЙ ПРОЦЕССОВ И ВЫЯВЛЕНИЕ ЦИКЛИЧЕСКИХ СТРУКТУР

О.С. Крюков

Рассматривается проблема оптимизации производственных процессов с использованием методологии расширенных сетей Петри. Предложен подход определения циклических структур в процессе с учетом ветвлений.

Ключевые слова: сети Петри, моделирование, производственный процесс, структура процесса, анализ, циклы.

Введение. Автоматизация процесса анализа и оптимизации существующих или разрабатываемых процессов на предприятиях представляет важную задачу, так как позволяет облегчить выполнение объемных работ по учету множества зависимостей и условий. Однако, реализация этих действий непосредственно на производстве может повлечь серьезные расходы, поэтому изначально необходимо сформировать модель процесса, к которой в дальнейшем будут применяться различные оптимизации.

Одним из вариантов оптимизации производственного процесса, не предполагающим изменение зависимостей между его этапами, является внедрение параллельности. Для автоматизированного преобразования производственного процесса к его параллельному варианту следует выбрать подходящую форму представления процесса, а также провести ряд операций по анализу этой модели, в том числе определить наличие нелинейных структур, усложняющих процесс преобразования.

К подобным структурам в первую очередь следует отнести циклические участки, которые представляют собой многократно повторяющуюся последовательность действий. Следовательно, внедрение параллельного исполнения на этих участках способно в значительной мере сократить временные затраты производства.

В связи с этим необходимо определить оптимальный вариант модели процесса и разработать методы выделения циклических структур в ней.

Обзор литературы. Наиболее подробная систематизация методов моделирования процессов представлена в работе [1], где кратко представлены достоинства и недостатки различных методов, однако нет рекомендаций для их применения на различных уровнях рассмотрения процессов и для различных этапов жизненного цикла.

Много как российских, так и зарубежных исследований затрагивают вопросы эффективности и возможности применения отдельных методов моделирования процессов, таких как: многоагентное моделирование [3], использование структурно-функциональных методов моделирования [4], нейросетевые модели [5, 6], применение

сетей Петри и их модификаций: иерархические раскрашенные временные сети [7], объектно-ориентированные сети [8], нечеткие стохастические сети [9], вложенные сети [10-12], в том числе и раскрашенные временные [13] и прочие.

Однако, как говорится в работе [14], наиболее простым и эффективным представляется использование сетей Петри, расширенных семантическими связями [15-17], так как данный подход предполагает различия в последовательности операций процесса и в информационных и материальных потоках, необходимых для ее осуществления, а также предоставляет математический аппарат, позволяющий рассмотреть в динамике различные стороны процесса, исследовать временные и структурные характеристики процесса.

Вопросам выявления циклов в сетях Петри посвящены работы [18-22]. Данные работы описывают выявление циклов в сети как в рамках ее декомпозиции, так и для верификации структуры. Но в описанных методиках не производится учет такой структуры как ветвление, что при анализе сети может привести к некорректному результату. Следовательно, необходимо сформировать метод выявления циклов в сетях Петри с учетом ветвлений.

Расширенные сети Петри с семантическими связями. Простейшая расширенная сеть Петри с семантическими связями (РСПСС) может задаваться следующим множеством:

$$\Pi = \{A, \{Z^C, \tilde{R}^C, \hat{R}^C\}, \{Z^S, \tilde{R}^S, \hat{R}^S\}\},$$

где $A = \{a_{1(a)}, \dots, a_{j(a)}, \dots, a_{J(a)}\}$ – конечное множество позиций;

$Z^C = \{z_{1(z^C)}^C, \dots, z_{j(z^C)}^C, \dots, z_{J(z^C)}^C\}$ – конечное множество переходов по управлению;

\tilde{R}^C – матрица смежности размером $J(a) \times J(z^C)$, отображающая множество позиций в множество переходов по управлению; \hat{R}^C – матрица смежности размером $J(z^C) \times J(a)$, отображающая множество переходов по управлению в множество позиций; $Z^S = \{z_{1(z^S)}^S, \dots, z_{j(z^S)}^S, \dots, z_{J(z^S)}^S\}$ – конечное множество переходов по семантическим связям; \tilde{R}^S – матрица смежности размером $J(a) \times J(z^S)$, отображающая множество позиций в множество переходов по семантическим связям; \hat{R}^S – матрица смежности размером $J(z^S) \times J(a)$, отображающая множество переходов по семантическим связям в множество позиций.

Кроме того, задаются следующие функции переходов:

входная функция переходов по управлению:

$$I_A(Z^C) = \{I_A(z_{1(z^C)}^C), \dots, I_A(z_{j(z^C)}^C), \dots, I_A(z_{J(z^C)}^C)\};$$

выходная функция переходов по управлению:

$$O_A(Z^C) = \{O_A(z_{1(z^C)}^C), \dots, O_A(z_{j(z^C)}^C), \dots, O_A(z_{J(z^C)}^C)\};$$

входная функция переходов по семантическим связям:

$$I_A(Z^S) = \{I_A(z_{1(z^S)}^S), \dots, I_A(z_{j(z^S)}^S), \dots, I_A(z_{J(z^S)}^S)\};$$

выходная функция переходов по семантическим связям:

$$O_A(Z^S) = \{O_A(z_{1(z^S)}^S), \dots, O_A(z_{j(z^S)}^S), \dots, O_A(z_{J(z^S)}^S)\}.$$

Между позициями сети выполняется отношение предшествования по управлению. Позиция a_i считается предшествующей по управлению для a_j (задается как $a_i <_C a_j$), если существует такой путь по управлению из стартовой позиции процесса a_S в позицию a_j , что он включает в себя позицию a_i .

Для стартовой позиции процесса справедливо:

$$\exists z_{i(z^C)}^C (I_A(z_{i(z^C)}^C)) = a_S \wedge \neg \exists z_{j(z^C)}^C (O_A(z_{j(z^C)}^C) = a_S)$$

Структуры РСПСС. В модели процесса, построенной в соответствии с определением РСПСС, можно выделить три вида структур, формируемых в результате различных комбинации позиций и переходов по управлению:

линейный участок:

$$Ln = \{a_{i(a)}\}, |Ln| \geq 1;$$

цикл:

$$C = \{a_{cB}, (a_{cond}, a_{out}), Cp\},$$

где a_{cB} - начальная позиция, a_{cond} - условие (branch-позиция), a_{out} - внешняя позиция, в которую передается управление из цикла, Cp - тело цикла;

ветвление:

$$Br = \{a_{xB}, \{B_{x1}, \dots, B_{xn}\}, a_{xe}\},$$

где a_{xB} - начальная позиция ветвления, B_{xi} - ветви ветвления, a_{xe} - конечная позиция ветвления.

Циклы подразумевают возвращение по управлению на начальную позицию, наличие условия, выводящего управление за пределы цикла, и тела цикла, содержащего последовательность линейных участков, ветвлений и циклов.

Для цикла выполняются следующие условия:

$$\left\{ \begin{array}{l} \exists z_{i(z^C)}^C (O_A(z_{i(z^C)}^C) = a_{cB} \wedge I_A(z_{i(z^C)}^C) = a_{i(a)}) | a_{cB} <_C a_{i(a)} \\ |Z_O^C| > 1 \forall z_{i(z^C)}^C \in Z_O^C (I_A(z_{i(z^C)}^C) = a_{cond}) \wedge \neg \exists Br_i (a_{xB} = a_{cond}) \\ \exists z_{i(z^C)}^C (I_A(z_{i(z^C)}^C) = a_{cond} \wedge O_A(z_{i(z^C)}^C) = a_{out}) | a_{out} \notin Cp \wedge a_{out} \neq a_{cB} \end{array} \right. ;$$

В рамках работы выделим следующие формы циклов:

цикл с предусловием (рис. 1а):

$$a_{cond} = a_{cB};$$

цикл с постусловием (рис. 1б):

$$\exists z_{i(z^C)}^C (I_A(z_{i(z^C)}^C) = a_{cond} \wedge O_A(z_{i(z^C)}^C) = a_{cB}).$$

Кроме того, определим следующие формы вложенности циклов:

простая вложенность (рис. 2, а):

$$\exists C_{i(C)} (\exists C_{j(C)} (C_{j(C)} \in Cp_{i(C)} \wedge C_{j(C)} \notin Cp_{i(C)}));$$

вложенность с пересечением (рис. 2, б):

$$\exists C_{i(C)} (\exists C_{j(C)} (a_{cB_{i(C)}} = a_{cB_{j(C)}} \wedge C_{j(C)} \in Cp_{i(C)})).$$

Структурный анализ РСПСС. Для анализа структуры РСПСС необходимо построить модель выбранного процесса. Для этого каждый этап процесса преобразуется в позицию сети, а последовательность операций задается при помощи формирования управляющих переходов между позициями. Формирование семантических переходов, на данном этапе не требуется.

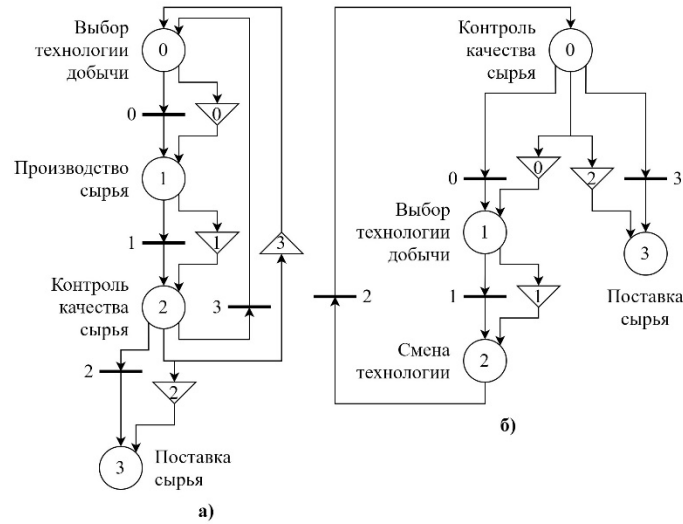


Рис. 1. Примеры циклов: а – с предусловием, б – с постусловием

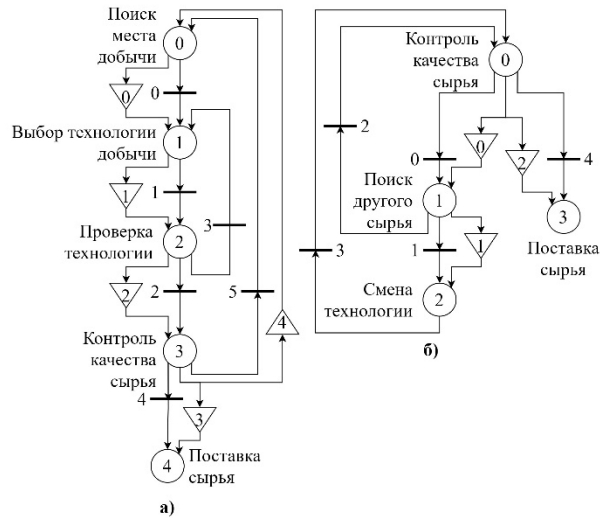


Рис. 2. Примеры вложенности циклов: а – простая вложенность; б – вложенность с пересечением

Для определения циклических структур, построенная сеть подвергается анализу в следующем порядке:

1. Поиск стартовой позиции сети a_S .
2. Поиск начальных позиций циклов.
3. Поиск начальных позиций ветвлений:

$$|Z_{O_i}^C| > 1 \forall z_{i(z^C)}^C \in Z_{O_i}^C (I_A(z_{i(z^C)}^C) = a_{xB}) \wedge \neg \exists C_i (a_{cB} = a_{xB}) \wedge \neg \exists z_{j(z^C)}^C \in Z_{O_i}^C (O_A(z_{j(z^C)}^C) = a_{i(a)}) | a_{i(a)} <_C a_{xB}.$$

4. Определение условий циклов и вложенности с пересечением.
5. Заполнение содержимого ветвей каждого из ветвлений.
6. Заполнения тела каждого из циклов.

Определение условий циклов и вложенности с пересечением выполняется по алгоритму, описанному ниже, в соответствии с обозначениями:

$C_{i(C)}$ - рассматриваемый цикл;

$Conds$ – множество позиций, являющихся возможными условиями для $C_{i(C)}$;

Z_O - множество переходов, в которые ведет $a_{cB_{i(C)}}$;

eC – множество уже рассмотренных циклов, начало которых совпадает с $C_{i(C)}$;

$parC$ - множество циклов, в которые может быть вложен $C_{i(C)}$.

Алгоритм выполняется для каждого найденного цикла.

Алгоритм *ФормированиеЦикла*($C_{i(C)}$)

Начало

$Conds = \{a_{i(a)}\} \mid a_{cB_{i(C)}} <_C a_{i(a)} \wedge \exists z_{i(Z^C)}^C (I_A(z_{i(Z^C)}^C) = a_{i(a)} \wedge O_A(z_{i(Z^C)}^C) = a_{cB_{i(C)}})$;

$Z_O = \{z_{i(Z^C)}^C\} \mid I_A(z_{i(Z^C)}^C) = a_{cB_{i(C)}} ;$

if $|Conds| = 1$ then

if $|Z_O| > 1$ then

$a_{cond_{i(C)}} = a_{cB_{i(C)}} ;$

$a_{out_{i(C)}} = a_{i(a)} \mid \exists z_{i(Z^C)}^C (I_A(z_{i(Z^C)}^C) = a_{cB_{i(C)}} \wedge O_A(z_{i(Z^C)}^C) = a_{i(a)}) \wedge a_{i(a)} \notin Cp_{i(C)} ;$

else

$a_{cond_{i(C)}} = Conds_0 ;$

$a_{out_{i(C)}} = a_{i(a)} \mid \exists z_{i(Z^C)}^C (I_A(z_{i(Z^C)}^C) = a_{cond_{i(C)}} \wedge O_A(z_{i(Z^C)}^C) = a_{i(a)}) \wedge a_{i(a)} \neq a_{cB_{i(C)}} ;$

end

else

$eC = \{C_{j(C)}\} \mid a_{cond_{j(C)}} = a_{cB_{i(C)}} \vee a_{cond_{j(C)}} \in Conds ;$

if $|eC| = 0$ and $|Z_O| > 1$ then

$a_{cond_{i(C)}} = a_{cB_{i(C)}} ;$

$a_{out_{i(C)}} = a_{i(a)} \mid \exists z_{i(Z^C)}^C (I_A(z_{i(Z^C)}^C) = a_{cB_{i(C)}} \wedge O_A(z_{i(Z^C)}^C) = a_{i(a)}) \wedge a_{i(a)} \notin Cp_{i(C)} ;$

else

$Conds = Conds \setminus a_{i(a)} \mid |Z_{Or}| = 1 \forall z_{i(Z^C)}^C \in Z_{Or} (I_A(z_{i(Z^C)}^C) = a_{i(a)}) ;$

$Conds = Conds \setminus a_{i(a)} \mid \exists C_{j(C)} (a_{cond_{j(C)}} = a_{i(a)}) ;$

$a_{cond_{i(C)}} = a_{i(a)} \mid a_{i(a)} \in Conds \wedge \forall a_{j(a)} \in Conds (a_{j(a)} <_C a_{i(a)}) ;$

$a_{out_{i(C)}} = a_{i(a)} \mid \exists z_{i(Z^C)}^C I_A(z_{i(Z^C)}^C) = a_{cond_{i(C)}} \wedge O_A(z_{i(Z^C)}^C) = a_{i(a)} \wedge a_{i(a)} \neq a_{cB_{i(C)}} ;$

if $|eC| = 1$ then

$Cp_{eC_0} = Cp_{eC_0} \cup C_{i(C)} ;$

else

$parC = \{C_{j(C)}\} \mid C_{j(C)} \in eC \wedge a_{cond_{j(C)}} \neq a_{cB_{j(C)}} ;$

$Cp_{j(C)} = Cp_{j(C)} \cup C_{i(C)} \mid C_{j(C)} \in parC \wedge \forall C_{k(C)} \in parC (a_{cond_{j(C)}} <_C a_{cond_{k(C)}}) ;$

end

end

end Конеч.

Заполнение тела циклов выполняется по алгоритмам ниже со следующими обозначения:

$C_{i(C)}$ - рассматриваемый цикл;

inC – вложенный в $C_{i(C)}$ цикл, начинающийся в $a_{cB_{i(C)}}$;

p – рассматриваемая позиция;

Ln – формируемый линейный участок;

np – следующая рассматриваемая позиция.

Алгоритмы поочередно выполняются для каждого из еще нерассмотренных циклов.

Алгоритм *АнализЦикла*($C_{i(C)}$)

Начало

$inC = C_{j(C)} \mid a_{cB_{j(C)}} = a_{cB_{i(C)}} \wedge C_{j(C)} \in Cp_{i(C)}$;

if $inC \neq \emptyset$ then

АнализТелаЦикла($C_{i(C)}, a_{out_{inC}}, \emptyset$);

else

$p = a_{i(a)} \mid \exists z_{i(Z^C)}^C (I_A(z_{i(Z^C)}^C) = a_{cB_{i(C)}} \wedge O_A(z_{i(Z^C)}^C) = a_{i(a)}) \wedge$

$\neg \exists C_{j(C)} (C_{j(C)} \in Cp_{j(C)} \wedge a_{out_{j(C)}} = a_{i(a)} \wedge a_{i(a)} \neq a_{out_{i(C)}})$;

АнализТелаЦикла($C_{i(C)}, p, \emptyset$);

end

Конец.

Алгоритм *АнализТелаЦикла*($C_{i(C)}, p, Ln$)

Начало

if $a_{cB_{i(C)}} = p$ or $a_{cond_{i(C)}} = p$ then

if $|Ln| \neq \emptyset$ then

$Cp_{i(C)} = Cp_{i(C)} \cup Ln$;

end

return;

end

if $\exists Br_i(Br) (a_{xB_{i(Br)}} = p)$ then

$Cp_{i(C)} = Cp_{i(C)} \cup Br_i(Br)$;

$np = a_{out_{i(Br)}}$;

if $|Ln| \neq \emptyset$ then

$Cp_{i(C)} = Cp_{i(C)} \cup Ln$;

end

АнализТелаЦикла($C_{i(C)}, np, \emptyset$);

return;

end

if $\exists C_{j(C)} (a_{cB_{j(C)}} = p)$ then

if $C_{j(C)} \notin Cp_{i(C)}$ then

$Cp_{i(C)} = Cp_{i(C)} \cup C_{j(C)}$;

```

end
 $np = a_{out_{i(C)}};$ 
if  $|Ln| \neq \emptyset$  then
 $Cp_{i(C)} = Cp_{i(C)} \cup Ln;$ 
end
АнализТелаЦикла( $C_{i(C)}, np, \emptyset$ );
return;
end
 $Ln = Ln \cup p;$ 
 $np = a_{i(a)} \mid \exists z_{i(Z^C)}^C (I_A(z_{i(Z^C)}^C) = p \wedge O_A(z_{i(Z^C)}^C) = a_{i(a)});$ 
АнализТелаЦикла( $C_{i(C)}, np, Ln$ );
Конец.
```

Описанные алгоритмы позволяют сформировать структуры циклов, позволяющих однозначно определить их наличие и положение в модели процесса, что на этапе преобразования облегчит внедрение иерархии в сеть путем замены целой структуры одной позицией.

Заключение. Предложенные в работе методы анализа расширенной семантическими связями сети Петри на предмет наличия циклических структур могут применяться в ходе оптимизации производственных процессов. Информация, полученная в результате применения описанных алгоритмов, позволит более эффективно распределять имеющиеся у предприятия ресурсы на этапе планирования модернизации производства или при планировании новых производств.

Исследование выполнено при финансовой поддержке гранта Президента Российской Федерации для государственной поддержки молодых российских ученых - кандидатов наук МК-1160.2020.9.

Список литературы

1. Сироткин М.Е. Методы моделирования производственных процессов предприятия машиностроения // Машиностроение и компьютерные технологии. 2011. №. 8. 14 с.
2. Дидманидзе Р.Н. Обоснование методов оптимального моделирования производственных процессов // Международный технико-экономический журнал. 2018. №. 2. С. 66-71.
3. Kuhn W. Digital factory-simulation enhancing the product and production engineering process // Proceedings of the 2006 winter simulation conference. IEEE, 2006. С. 1899-1906.
4. Затонский А.В., Уфимцева В.Н. Разработка объектных средств имитационного и многоагентного моделирования производственных процессов // Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика. 2018. №. 4. С. 56 – 62.
5. Куликов Г.Г., Ризванов К.А., Петров Ю.Е. Системный подход к построению структуры организационно-функциональной модели цифрового моделирования производственных процессов // Вестник Южно-Уральского государственного университета. Серия: Компьютерные технологии, управление, радиоэлектроника. 2018. Т. 18. № 2. С. 60 – 68.
6. De Filippis, Luigi Alberto Ciro, et al. ANN Modelling to Optimize Manufacturing Process // Advanced Applications for Artificial Neural Networks. IntechOpen, 2017. P. 201 – 225.

7. Westkämper E., Schmidt T., Wiendahl H. H. Production planning and control with learning technologies: Simulation and optimization of complex production processes // Knowledge-based systems. Academic Press, 2000. С. 839-887.
8. Седых И.А., Анисеев Е.С. Представление цементного производства иерархическими раскрашенными временными сетями Петри на основе окрестностных моделей // Вестник Липецкого государственного технического университета. 2017. № 1. С. 83 – 89.
9. Dong M., Chen F.F. Process modeling and analysis of manufacturing supply chain networks using object-oriented Petri nets // Robotics and Computer-Integrated Manufacturing. 2001. Т. 17. № 1-2. С. 121-129.
10. Wang J. et al. Modelling a remanufacturing reverse logistics system using fuzzy stochastic Petri net // International Journal of Industrial and Systems Engineering. 2015. Т. 19. № 3. С. 311-325.
11. Ломазова И.А. Вложенные сети Петри и моделирование распределенных систем // Труды международной конференции «Программные системы: теория и приложения», Переславль-Залесский. М.: Физматлит. 2004. С. 337 – 352
12. Van Hee K. M. et al. Nested nets for adaptive systems // International Conference on Application and Theory of Petri Nets. Springer, Berlin, Heidelberg, 2006. С. 241-260.
13. Zhang L., Rodrigues B. Nested coloured timed Petri nets for production configuration of product families // International journal of production research. 2010. Т. 48. №. 6. С. 1805-1833.
14. Волошко А.Г., Ивутин А.Н., Крюков О.С. Методы моделирования и анализа производственных процессов для разработки стратегии модернизации предприятия // Известия Тульского государственного университета. Технические науки. 2020. Вып. 12. С. 36-43.
15. Ivutin A.N., Troshina A.G., Semantic Petri-Markov nets for automotive algorithms transformations // 2018 28th International Conference Radioelektronika (RADIOELEKTRONIKA). IEEE, 2018. С. 1-6.
16. Ивутин А.Н., Трошина А.Г. Метод формальной верификации параллельных программ с использованием сетей петри. // Вестник Рязанского государственного радиотехнического университета 70 (2019). 2019. С. 15-26.
17. Волошко А.Г., Крюков О.С. Extended Petri Nets Based Approach for Simulation of Distributed Manufacturing Processes // The 9th Mediterranean Conference on Embedded Computing (MECO 2020). 2020. С. 508-511.
18. Кулагин В.П., Цветков В.Я. Философия сетей Петри // Вестник МГТУ МИРЭА 2014 г, № 4. 2014. С. 18-38.
19. Козюра В.Е. Реализация системы проверки моделей раскрашенных сетей Петри с использованием разверток. Новосибирск, 2002. [Электронный ресурс] URL: <https://static.freereferats.ru/avtoreferats/01002631845.pdf> (дата обращения: 10.02.2021).
20. Рудаков И.В., Пащенко А.В. Программный комплекс верификации алгоритмов программного обеспечения с помощью иерархических сетей Петри // Инженерный журнал: наука и инновации. 2013. №2. С. 1 – 10.
21. Wil M.P. van der Aalst, Arthur H.M. ter Hofstede. Verification Of Workflow Task Structures: A Petri-net-baset Approach // Information Systems, Volume 25, Issue 1. 2000. P. 43-69.
22. D. Lefebvre and C. Delherm. Diagnosis of DES with Petri Net Models // IEEE Transactions on Automation Science and Engineering, vol. 4, no. 1, 2007. P. 114-118.

Крюков Олег Сергеевич, студент, ol_kryukov97@mail.ru, Россия, Тула, Тульский государственный университет

STRUCTURAL ANALYSIS OF PROCESS MODELS AND IDENTIFICATION
OF CYCLIC STRUCTURES

O.S. Kryukov

The paper deals with the problem of optimization of production processes using the extended Petri nets methodology. An approach is proposed for determining cyclic structures in the process, taking into account branching.

Key words: Petri nets, simulation, manufacturing process, process structure, analysis, cycles.

Kryukov Oleg Sergeevich, student, ol_kryukov97@mail.ru, Russia, Tula, Tula State University

УДК 621.865.8

DOI: 10.24412/2071-6168-2021-4-330-338

СИСТЕМА УПРАВЛЕНИЯ РОБОТОМ ВЕРТИКАЛЬНОГО ПЕРЕМЕЩЕНИЯ

П.А. Груничев, С.А. Воротников, А.С. Самойлова

Рассмотрена система управления роботом вертикального перемещения (РВП) для движения по наклонной опорной поверхности. Показаны основные аспекты реализации системы управления модели РВП в программной среде V-REP. Представлены результаты моделирования различных вариантов походки по ровной вертикальной поверхности. Выполнена оценка результатов путем сравнения параметра отклонения РВП от заданного курса для различных вариантов походки. Предложен способ коррекции движения РВП.

Ключевые слова: робот вертикального перемещения, вакуумный захват, система управления, цикл шагания, гусеничная походка, диагональная походка, последовательная походка.

В настоящее время роботы вертикального перемещения (РВП) составляют отдельный класс робототехнических систем, предназначенных для самостоятельного выполнения различных технологических операций на сложных вертикальных и наклонных поверхностях [1]. К таким технологическим операциям относятся: дефектоскопия сварных швов и элементов строительных сооружений; чистка и покраска корпусов судов; контрольно-измерительные и инспекционные операции; опасные для человека аварийные работы и т.д. [2]. В частности, диагностика поверхностей зданий позволяет находить трещины и другие дефекты на ранних стадиях развития и предотвращать возможные аварии и разрушения [3]. Кинематические схемы роботов, используемых для данного типа работ, представлены в основном в виде шагающих механизмов, а для удерживания на вертикальной поверхности применяются управляемые вакуумные присоски [4, 5].

Заметим, что в большинстве случаев управление РВП осуществляется в ручном режиме с помощью пульта оператора, на интерфейс которого поступает информация о положении робота, состоянии модулей системы управления и параметрах окружающей среды. Данный тип управления требует от оператора специальных навыков, а ошибка может привести к падению и поломке РВП. Поэтому при выполнении производственных задач, например, дефектоскопии наружных стен, РВП страхуется с помощью троса, спущенного с крыши здания [6].