

УДК 519.876.5

Т. О. Голубєва; В. М. Дубовой, д. т. н., проф.

МОДЕЛЬ ВИРОБНИЧИХ ПРОЦЕСІВ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ЯК ДИНАМІЧНА СИСТЕМА УПРАВЛІННЯ

Розглянуто питання планування виробничих процесів розробки програмного забезпечення з точки зору невизначеності ряду параметрів та за умов впливу людського фактору. У результаті цього розроблена модель динаміки системи управління виробничими процесами розробки програмного забезпечення.

Ключові слова: виробничий процес розробки програмного забезпечення, планування, модель динаміки.

Вступ

В останні роки розробка програмного забезпечення перетворилася з мистецтва окремих особистостей на виробничий процес, у якому зайняті мільйони фахівців усього світу. Але виробничі процеси розробки програмного забезпечення (ВППЗ) мають свою специфіку, яка полягає в тому, що необхідно планувати розумову і, певною мірою, творчу працю. Крім того, такі процеси на цей час повністю виконують люди і їх автоматизація поки що малоімовірна. Але незважаючи на це, планування і прогнозування ходу ВППЗ вкрай необхідне для правильного керування ресурсами процесу і є **актуальною науковою проблемою**.

Наразі планування і прогнозування ВППЗ є стандартизованим [1 – 3]. Реалізовано ряд систем планування ВППЗ, наприклад [4, 5]. Проте, сучасні системи планування ВППЗ не враховують невизначеність параметрів та характер окремих складових процесів, що відбуваються під час ВППЗ. Зокрема, не враховується вплив управління ресурсами під час виконання ВППЗ, яке перетворює ВППЗ на замкнену динамічну систему. У результаті цього постає **завдання** розробки моделі системи управління виробничими процесами розробки програмного забезпечення як замкненої динамічної системи в умовах невизначеності.

Розв'язання завдання

Нехай є множина ресурсів $H = \{H_1, H_2, \dots, H_n\}$ та множина завдань $Z = \{Z_1, Z_2, \dots, Z_m\}$. Ресурси можна об'єднати у підмножини множини H відповідно до завдань. Для виконання одного завдання можуть бути використані лише ті ресурси, які спроможні виконати це завдання.

У проаналізованій літературі [1 – 5] не знайдено посилань на те, що завдання (tasks) є різними за своїми властивостями. Проте у ВППЗ все більшу увагу приділяють тому факту, що процеси, які відбуваються в ході виконання завдань, відрізняються від запланованих через низку причин. Як правило, відхилення від запланованого ходу виконання завдань зумовлюються існуванням ризикованих подій. Якщо така ризикована подія настає, то виникає необхідність виконання роботи з її усунення. У результаті до запланованої множини завдань Z додається множина робіт з усунення наслідків того, що ризикована подія настане (УНРПН) $R = \{R_1, R_2, \dots, R_k\}$. Тому слід зауважити, що в цьому випадку, як правило, кількість ресурсів не збільшується, що призводить до необхідності перепланування розподілу ресурсів вже на стадії виконання робіт. Наголосимо, що роботи УНРПН відрізняються від завдань тим, що завдання обов'язково має бути виконане, а кожна робота УНРПН характеризується певною ймовірністю того, що відповідна ризикована подія настане.

Для спрощення охарактеризуємо ВППЗ наступними твердженнями:

- процес складається з фази планування та фази виконання робіт;
- процес має заплановані дати початку (ДПР) $D_{ДПР}$ та кінця робіт (ДКР) $D_{ДКР}$;
- вважається, що до фази планування усі проблеми вирішено, тому вони не включаються в процес планування (наявність проблем та необхідність їх вирішення в ході процесу значно ускладнює модель);
- завдання між собою не пов'язані;

- кількість ресурсів та завдань є фіксованими величинами;
- одне завдання виконує один виконавець (якщо завдання має виконувати декілька виконавців, то на стадії планування це завдання може бути розбите на підзавдання, кожне з яких буде виконувати один виконавець);
- робота УНРПН не є такою, що може зупинити виробничі процеси на довгий час.

Відповідно до зроблених позначень, система управління ВПРПЗ матиме вигляд, як показано на рис. 1. Ця схема побудована з урахуванням наведених тверджень. На схемі показано, що спочатку керівник робіт (КР) визначає завдання, роботи УНРПН та наявні ресурси. При цьому КР розподіляє ресурси на підмножини за завданнями. Далі він оцінює необхідний час для виконання кожного з завдань та кількість наявних людино-годин для вирішення кожного завдання. Після цього ресурси розподіляються між завданнями та роботами УНРПН. У подальшому відбувається оптимізація розробленого плану із використанням автоматизованих систем управління підприємством та процесами. На цьому фаза планування завершується і КР отримує розклад процесу. Надалі процес переходить у фазу виконання процесу. На етапі виконання процесу відбувається внесення в розроблений розклад вже відпрацьованих годин з кожного із завдань та робіт УНРПН. Після кожного такого внесення КР отримує оновлений розклад та оцінює за ним стан процесу. Якщо процес не завершено та не зупинено, необхідно повторно оптимізувати розклад. При цьому певні ресурси можуть знову розподілятися між задачами. Після цього відбувається знову внесення в розроблений розклад вже відпрацьованих годин з кожного із завдань та робіт УНРПН та оптимізація оновленого розкладу, допоки процес з якоїсь причини не завершується (завершення процесу не обов'язково означає повне виконання завдань).

На рис. 1 пунктиром виділений адаптивний контур системи розподілу ресурсів під час розробки програмного забезпечення. Фактично цей контур є описаною в попередньому абзаці фазою виконання робіт.

Особливістю процесів розробки програмного забезпечення є те, що всі операції з розробки виконують люди. Довільна операція, де використовується людська праця характеризується тим, що час на її виконання задається приблизно керівником процесу і досить часто із значним запасом. Це призводить до неефективного використання людських ресурсів. Доцільнішим автори вважають визначення проміжку часу, за який може бути виконана ця робота. Таке представлення дозволить програмі вказувати який час буде оптимальним з точки зору моделювання. Це дозволить менеджеру переглянути тривалість деяких робіт для того, щоб оптимізувати розподіл ресурсів. На цей час ця робота виконується самим менеджером, базуючись на досвіді та знанні кожного окремого виконавця. Оскільки проміжок на виконання конкретно взятого завдання визначається експертно керівником процесу, то доцільним є застосування нечіткої логіки для описання часу на виконання завдання [6].

Але представлення за допомогою математичного апарату нечіткої логіки є неповним. Такі оцінки в деяких випадках можна конкретизувати за допомогою використання досвіду попередніх процесів та вже виконаної частини роботи з цього процесу. Це може допомогти у випадках повторюваних завдань протягом процесу. До таких належать більшість завдань тестування програмного забезпечення. Особливістю планування процесу тестування є те, що необхідно виконувати ідентичне завдання декілька раз. При цьому кожен раз на цей прохід буде витрачено тою самою людиною різний час. Актуальним є використання бази даних з тим, щоб використовувати попередню інформацію для подальшого планування. У цьому випадку ми отримуємо модель, де одні часові параметри задані менеджером, а інші – прораховані системою та задані стохастично

(див. рис. 1). При цьому можна оцінювати як приблизний необхідний час на виконання завдання, так і оцінювати час конкретного виконавця, витрачений на виконання певного завдання.

На рис. 1 показано, що в ході планування та виконання процесу відбувається як занесення даних до бази знань, так і використання вже накопичених знань для планування.

Задачі можуть виконуватись паралельно, тобто, якщо не має обмеження, що завдання мають виконуватись у певній послідовності, то декілька виконавців можуть одночасно виконувати декілька завдань, але при цьому один виконавець одночасно виконує лише одне завдання.

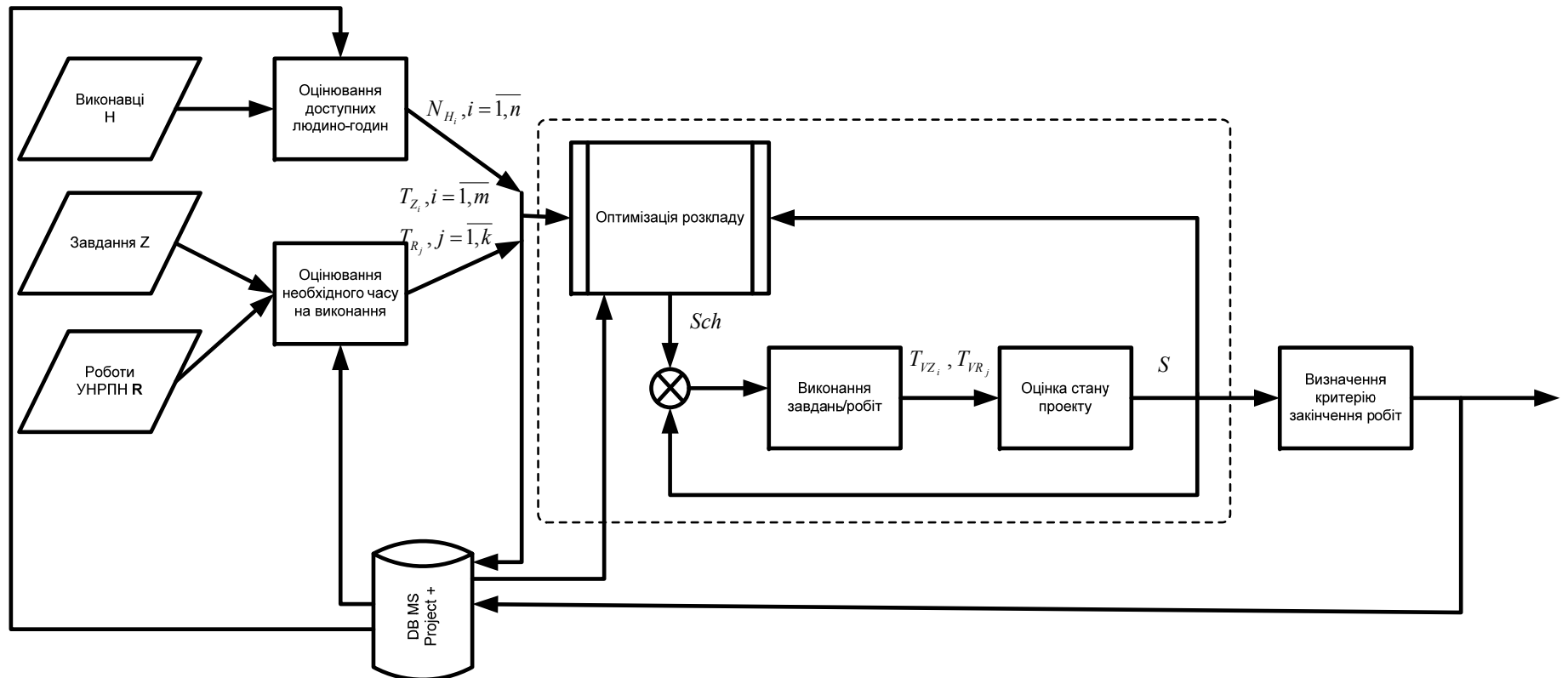


Рис. 1. Схема системи управління виробничим процесом розробки програмного забезпечення

На рис. 1 показана схема системи управління виробничим процесом розробки програмного забезпечення. Прогноз часу виконання процесу визначається динамікою замкненого контуру, який виділено пунктирною лінією. Розглянемо цю частину системи детальніше. Деталізована схема зображена на рис. 2.

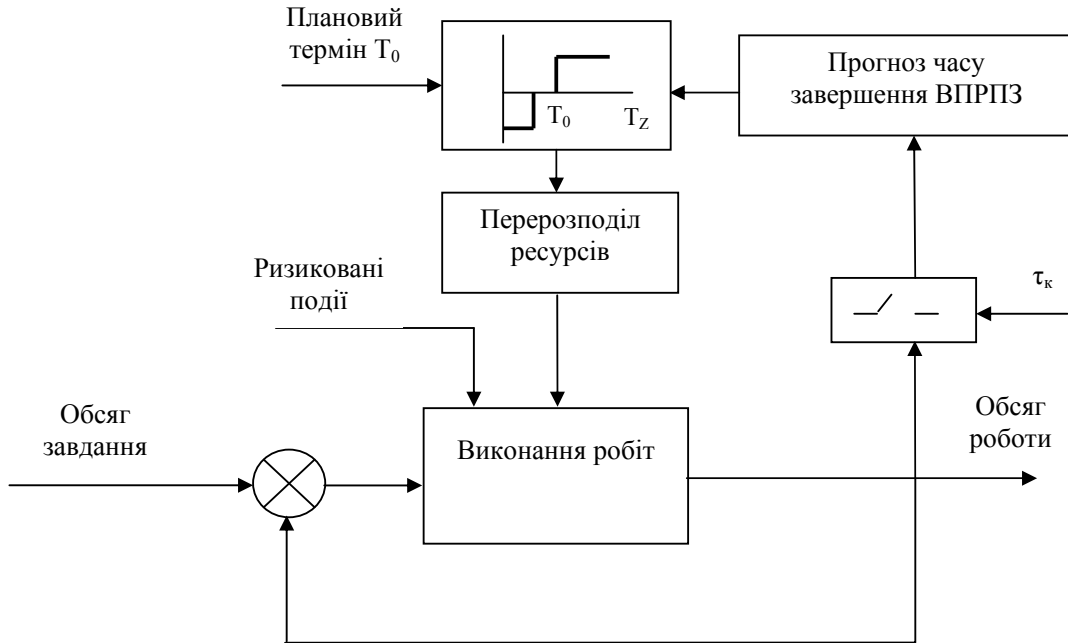


Рис. 2. ВПРПЗ як замкнена динамічна система

Розглянемо моделі динаміки кожної ланки системи.

Модель динаміки ланки «Виконання завдань» може бути представлена графічно фазовим портретом, зображеним на рис. 3.

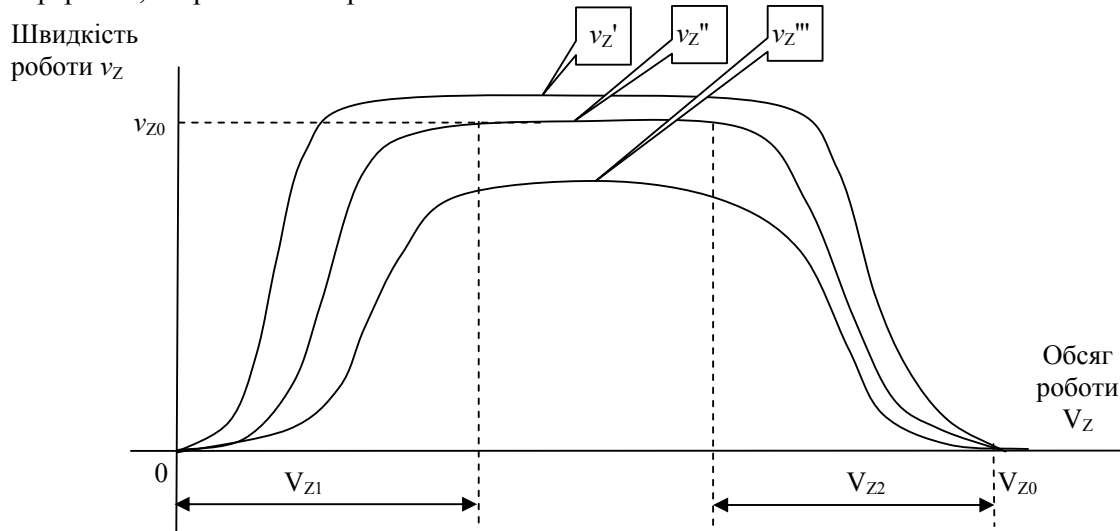


Рис. 3. Фазовий портрет ланки виконання робіт

Швидкість виконання завдання може бути визначена як функція від множини факторів

$$v_{Z_i} = f(V_Z, V_{Z0}, Sk_{Z_i}, Tp_{Z_i}, \{K_{E_j}, Mt_{E_j}, Rs_{E_j}, HV_{E_j}, SO_{E_j}, Hl_{E_j}, PS_{E_j}\}, Cm_{Z_i}), \quad (1)$$

де V_Z – виконаний обсяг завдання; V_{Z0} – загальний обсяг завдання; K_E – кваліфікація виконавця; Наукові праці ВНТУ, 2009, № 2

Sk_{z_i} – складність завдання, Tp_{z_i} – вид завдання, Mt_{z_i} – мотивація виконавця, $R_{S_{z_i}}$ – відповідальність виконавця, Sm_{z_i} – взаємодія в команді, HV_{z_i} – робоче обладнання, SO_{z_i} – самоорганізація виконавця, Hl_{z_i} – загальний стан виконавця, PS_{z_i} – здатність навчатися. Фазовий портрет побудований у координатах $v_z(V_z)$. Решта факторів визначають форму фазової траєкторії, яка характеризується трьома параметрами: довжиною початкової ділянки V_{z1} і відповідним часом налаштування виконавця на завдання, максимальною швидкістю виконання роботи v_{z0} , довжиною кінцевої ділянки V_{z2} і відповідним часом завершення роботи, який в свою чергу залежить від кількості помилок, зроблених під час виконання завдання.

Загальний час виконання роботи можна оцінити на основі інтегрування оберненої фазової траєкторії

$$T_Z = \int_0^{V_{z0}} \frac{dV_Z}{v_Z}. \quad (2)$$

Кожна робота УНРПН характеризується деякою ймовірністю p_{R_j} того, що її необхідно буде виконувати. Навіть якщо ризикована подія настане, робота УНРПН також може мати пріоритет виконання $w_{R_j} \in [0;1]$.

У результаті отримаємо зважені оцінки часу виконання завдання та робіт УНРПН:

$$T_{-zv_{z_i}} = w_{z_i} T_{z_i}; \quad (3)$$

$$T_{-zv_{R_j}} = w_{R_j} p_{R_j} T_{R_j}. \quad (4)$$

Другою складовою моделі динаміки є модель контролю стану виконання роботи, яка характеризується періодичністю τ_k і похибкою ΔV_Z . Оскільки контроль здійснюється експертом-керівником, то оцінка похибки є нечіткою. Періодичність контролю залежить від пріоритету роботи з урахуванням обмежень: не частіше, ніж раз на день і не рідше, ніж раз на тиждень.

Третім елементом моделі динаміки є прогноз терміну завершення роботи T . Прогноз здійснюється на основі залежності (1) і формули (2). Оскільки залежність (1) є нечіткою, то і інтеграл (2) є нечітким [8].

Четвертою складовою моделі динаміки є елемент порівняння прогнозованого терміну завершення роботи з плановим терміном $T_0 = \{T_{0min}, T_{0max}\}$ і прийняття рішення щодо необхідності перерозподілу ресурсів.

Прогноз терміну завершення і прийняття рішення є формальними процедурами, часом виконання яких можна знехтувати.

Перерозподіл ресурсів є пошуково-оптимізаційною задачею з обмеженнями. Він здійснюється на основі базової моделі (1), (2). Обмеження на перерозподіл ресурсів стосуються як самих ресурсів, так і часу перерозподілу, і залежать від пріоритету задачі — чим вищий пріоритет, тим менш жорсткі обмеження. Оскільки ресурси ВПРПЗ є дискретною величиною, то перерозподіл ресурсів здійснюється на основі багатокрокової стратегії, приклад якої зображений на рис. 4.

Отже, загальна модель динаміки подається системою рівнянь:

$$v_{z_i} = f(V_Z, V_{z0}, Sk_{z_i}, Tp_{z_i}, \{K_{E_j}, Mt_{E_j}, R_{S_{E_j}}, HV_{E_j}, SO_{E_j}, Hl_{E_j}, PS_{E_j}\}, Cm_{z_i});$$

$$T_Z = \sum_{k=1}^m \int_{V_{z_{kноч}}}^{V_{z_{kкін}}} \frac{dV_Z}{v_{z_k}};$$

$$m < (T_Z - t) / \tau_k; \quad (5)$$

$$u = \begin{cases} +1, & T_Z > T_{0\max}; \\ 0, & T_{0\min} \leq T_Z \leq T_{0\max}; \\ -1, & T_Z < T_{0\min}; \end{cases}$$

$$(m, [V_{Z_{k\text{поч}}}, V_{Z_{k\text{кін}}}], \{E_k\}): \min |T_Z - T_0|.$$

де m – кількість кроків стратегії розподілу ресурсів.

Система рівнянь (5) є нечіткою і жорсткою, а окремі її параметри задаються статистично. Для її розв'язання, з метою знаходження прогнозу часу виконання робіт, необхідне застосування спеціальних методів та алгоритмів [8 – 11].

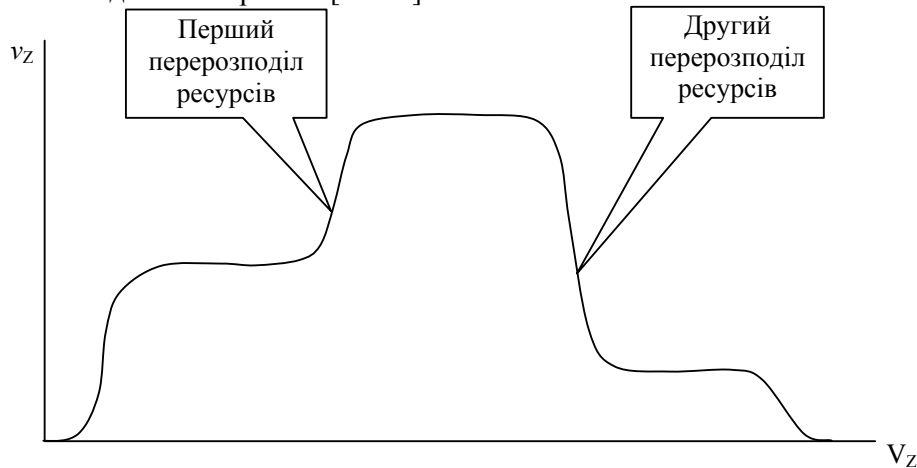


Рис. 4. Приклад багатокрокової стратегії розподілу ресурсів

Висновки

У статті розроблена модель динаміки системи управління виробничими процесами розробки програмного забезпечення. Ця система враховує невизначеність ряду параметрів виконавців та робіт, втручання керівника і перерозподіл ресурсів між завданнями, а також ризиковані події, що можуть виникнути в ході виконання робіт, і дозволяє підвищити точність і достовірність прогнозування термінів виконання робіт.

СПИСОК ЛІТЕРАТУРИ

1. ISO – Management standards. – Режим доступу : http://www.iso.org/iso/iso_catalogue/management_standards.htm.
2. Бушуев С. Д. Управление проектами. Основы профессиональных знаний и система оценки компетенции проектных менеджеров (National Competence Baseline, NCB UA Version 3.0) / С. Д. Бушуев, Н. С. Бушуева. – К. : ІРІДІ-УМ, 2006. – 208 с.
3. Типовые нормы времени на программирование задач для ЭВМ. – М. : НИИ труда, 1980. – 28 с.
4. Спайдер Проджект: Управление Проектами|Project Management|консалтинг|обучение|Spider Project – 2008. – Режим доступу : <http://www.spiderproject.ru/>.
5. Project Home Page – Microsoft Office Online – 2008. – Режим доступу : <http://office.microsoft.com/en-us/project/FX100487771033.aspx>.
6. Yousefli A. A. New Heuristic Model for Fully Fuzzy Project Scheduling / A. Yousefli, M. Ghazanfari, K. Shahanaghi, M. A. Heydari. – Режим доступу : <http://www.worldacademicunion.com/journal/jus/jusVol02No1paper07.pdf>.
7. Орловский С. А. Проблемы принятия решений при нечеткой исходной информации / С. А. Орловский. – М. : Радио и связь, 1981. – 286 с.
8. Раскин Л. Г. Нечеткая математика. Основы теории. Приложения / Л. Г. Раскин, О. В. Серая. – Х. : Парус, 2008. – 352 с.
9. Москвіна С. М. Аналіз алгоритмів розв'язання систем жорстких диференціальних рівнянь у пакетах моделювання систем автоматичного керування / С. М. Москвіна, Т. О. Голубева // Вісник Вінницького політехнічного інституту. – 2004. – № 2. – С. 21–25: іл. 5. – ISSN 1997–9266.
10. Москвіна С. М. Підхід до підвищення стійкості чисельних алгоритмів при моделюванні систем автоматичного керування / С. М. Москвіна, Т. О. Голубева // Вісник Вінницького політехнічного інституту. – 2003. – № 6. – С. Наукові праці ВНТУ, 2009, № 2

92–95: іл. 1. – ISSN 1997–9266.

11. Москвіна С. М. Про підвищення стійкості методів моделювання жорстких систем / С. М. Москвіна, О. М. Москвін, Т. О. Голубєва // Вісник Вінницького політехнічного інституту. – 2007. – № 1 – С. 84 – 88. – ISSN 1997–9266.

Голубєва Тетяна Олександрівна – аспірантка кафедри комп'ютерних систем управління.
gtat@bigmir.net.

Дубовой Володимир Михайлович – д. т. н., професор, завідувач кафедри комп'ютерних систем управління.
Вінницький національний технічний університет.