

(New York: Wiley, 1980). 3. Hu X. and Wang J. An improved dual neural network for solving a class of quadratic programming problems and its k -winners-take-all application, *IEEE Trans. on Neural Networks*, 19 (2008) 2022-2031. 4. Kwon T. M. and Zervakis M. A parallel sorting network without comparators: A neural-network approach, in: *Proc. Int. Joint Conf. on Neural Networks*, Vol. 1 (1992) 701-706. 5. Lippmann R. P., Gold B. and Malpass M.L. A comparison of Hamming and Hopfield neural nets for pattern classification, *MIT Lincoln Laboratory Technical report TR-769* (1987) 1-37. 6. Liu S. and Wang J. A simplified dual neural network for quadratic programming with its KWTA application, *IEEE Trans. on Neural Networks*, 17 (2006) 1500-1510. 7. Marinov C. A. and Hopfield J. J. Stable computational dynamics for a class of circuits with $O(N)$ interconnections capable of KWTA and rank extractions, *IEEE Trans. on Cir. and Syst. I: Fundamental Theory and Applications* 52 (2005) 949-959. 8. Tymoshchuk P. and Kaszkurewicz E. A Winner-take-all circuit based on second order Hopfield neural networks as building blocks, in: *Proc. Int. Joint Conf. on Neural Networks*, Vol. II (2003) 891-896. 9. Tymoshchuk P. and Kaszkurewicz E. A winner-take-all circuit using neural networks as building blocks, *Neurocomputing* 64 (2005) 375-396. 10. Urahama K. and Nagao T. K -Winner-take-all circuit with $O(n)$ complexity, *IEEE Trans. on Neural Networks* 6 (1995) 776-778. 11. Yang J. F. and Chen C. M. A Dynamic K -Winners-Take-All Neural Network, *IEEE Trans. on Syst., Man and Cyb.* 27 (1997) 523~526. 12. Yen J. C., Guo J. I. and Chen H.-C. A new k -Winners-take all neural network and its array architecture, *IEEE Trans. on Neural Networks* 9 (1998) 901-912.

УДК 004.942

Р. Вовк, В. Процюк, В. Шекета

Івано-Франківський національний технічний університет нафти і газу
кафедра програмного забезпечення автоматизованих систем

ФОРМАЛЬНИЙ ОПИС ПРОЦЕСУ КОНТРОЛЮ ЗАДОВОЛЕННЯ ТА ПОРУШЕННЯ ОБМЕЖЕНЬ В ІНТЕЛЕКТУАЛЬНИХ СИСТЕМАХ

© Вовк Р., Процюк В., Шекета В., 2011

Досліджено процес функціонування інтелектуальної навчальної системи засобами формальних методів теорії інформаційно-пошукових задач на основі обмежень. Побудовано множини, системи та ієрархії обмежень для статичних та динамічних навчальних проблем з виділеними станами, для яких описано можливі помилки рішень як результат порушення накладених обмежень з повідомленнями зворотного зв'язку.

Ключові слова – обмеження, ієрархія обмежень, правила, умова релевантності, умова задоволення.

The exploration of the functioning process for intelligible tutoring system by the means of constraints satisfaction problem formal methods is done. The sets, systems and hierarchies of constraints for static and dynamic learning problems with selected states is introduced with descriptions of possible solution errors, that are arising by violation of imposed constraints complemented with relevant feedback messages is proposed.

Keywords: constraints, constraints hierarchy, rules, relevancy condition, satisfaction condition.

Вступ

Інформаційно-пошукові задачі на основі обмежень (*CSP – constraints satisfaction problem*) [1, 2] формулюються шляхом накладання множини обмежень на деяку скінченну множину змінних. Для кожної змінної вводиться її домен, що містить множину її можливих значень. Інтерпретація обмеження полягає в контролі кортежів значень для відповідних наборів змінних. Обмеження за-яються явно або неявно, всі досліджувані множини змінних, доменів та обмежень є скінченними.

Інтелектуальні навчальні системи на основі обмежень – це вид інтелектуальних систем на основі підходу *Constraint-Based Modeling* [3], що дає змогу застосовувати їх для вирішення навчальних проблем шляхом виявлення помилок та хибних рішень.

Аналіз останніх публікацій і досліджень

Як показують дослідження [4, 5], інтелектуальна навчальна система на основі обмежень дає змогу реалізувати істотні переваги в області комп'ютеризованого навчання. Основі ідеї, які досліджуються, полягають в наступному: 1) підтримка індивідуалізованого навчання; 2) аналіз та діагностика дій об'єкта навчання засобами інтелектуальних систем; 3) створення та підтримка моделі об'єкта навчання; 4) забезпечення підтримки вирішення навчальних проблем на основі методів підтримки та оновлення моделей об'єкта навчання; 5) коректна генерація повідомлень зворотного зв'язку; 6) ведення та підтримка діалогів з об'єктом навчання. Отже, основне завдання системи полягає в створенні середовища підтримки об'єкта навчання при вирішенні ним навчальних (технологічних) проблем на основі інтелектуальних засобів виявлення помилок та генерації зворотного зв'язку. В [6] виконано формалізацію основних елементів архітектури навчальної системи на основі обмежень. Запропоновано сумарну формальну модель об'єкта навчання, отриману в результаті фіксації порушених обмежень при вивченні навчальних курсів, що розглядаються як сукупність навчальних проблем. У [7] досліджено модель навчальної системи з використанням підходу *Constraint-Based Modeling*. Отримана модель формалізує основні елементи архітектури навчальної системи на основі обмежень і є прототипом веб-базованої реалізації для вивчення мов програмування на основі аналізу типових помилок об'єктів навчання. В [8] досліджено структуру зворотного зв'язку в навчальній системі на основі обмежень. Запропоновано використання зворотного зв'язку у вигляді адаптивних впливів залежно від ваги порушеного обмеження.

Мета та завдання досліджень

Цілі статті. Основні цілі цієї статті є такими:

1. Структуризація та формалізація процесу функціонування механізму інтерпретації помилок рішень навчальних (технологічних) проблем в інтелектуальній системі на основі обмежень;
2. Введення способів впорядкування множин обмежень та помилок;
3. Опис процедури формування інтелектуального зворотного зв'язку з об'єктом навчання;
4. Дослідження представлень умов релевантності та умов задоволення в формі правил;
5. Побудова та інтерпретація формальних моделей профілів об'єктів навчання.

Невирішені проблеми. Виконані авторами дослідження проблеми побудови інтелектуальної навчальної системи на основі обмежень для інтерпретації помилок рішень навчальних (технологічних) проблем як невирішених проблем дають змогу сформулювати задачу прив'язки та інтерпретації множини помилок в контексті формально-логічного підходу інформаційно-пошукових задач на основі обмежень.

Основні результати досліджень

З погляду функціонування інтелектуальної навчальної системи, інтелектуальність якої полягає в наданні допомоги користувачу щодо вирішення певних навчальних (технологічних) проблем (станів проблем, навчальних курсів, побудованих з навчальних проблем та технологічних процесів, що розглядаються як сукупність технологічних проблем та пов'язаних з ними нештатних, надзвичайних ситуацій): $\{LC_i, LP_j, LPS_k\}_{i,j,k \in N}$, $\{TP_{i_1}, TP_{j_1}, TPS_{k_1}\}_{i_1, j_1, k_1 \in N}$, де LC_i (*LearningCourse*) – i -тий навчальний курс, LP_j (*LearningProblem*) – j -та навчальна проблема, LPS_k (*LearningProblemState*) – k -й стан навчальної проблеми, TP_{i_1} (*TechnologicalProcces*) – i_1 -тий технологічний процес, TP_{j_1} (*TechnologicalProblem*) – j_1 -та технологічна проблема, TPS_{k_1} (*TechnologicalProblemState*) k_1 -й стан технологічної проблеми.

Порядок на рівнях ієрархії визначається шляхом об'єднання відповідних сукупностей (множин): $LC_i = \bigcup_{j=1..j_{max}} LP_j$, $LP_j = \bigcup_{k=1..k_{max}} LPS_k$; $TP_{i_1} = \bigcup_{j_1=1..j_1^{max}} LP_{j_1}$, $TP_{j_1} = \bigcup_{k_1=1..k_1^{max}} TPS_{k_1}$.

Введені ієрархії є формальною основою для побудови *навчальних* профілів системи (вивчення мов програмування) та *технологічних* профілів (надзвичайні ситуації при бурінні нафти і газу). Оскільки формальна структура обидвох профілів в основному збігається, то подальший виклад буде побудований на основі ієрархії першого виду.

Необхідним є розгляд двох основних категорій навчальних проблем: *статичні* навчальні (технологічні) проблеми етапу наповнення бази знань системи та *динамічні* навчальні (техноогічні) проблеми етапу функціонування системи. На етапі формування бази знань кожна навчальна проблема розглядається як деяка скінченна множина її станів. Кожен стан $LPS_i, i = 1..k_{max}$, де k_{max} – максимальна кількість станів в проблемі LP_j , описується деякою множиною обмежень

$$ConstrSet_k = \{ c_1 : fb_1^{set}, c_2 : fb_2^{set}, \dots, c_n : fb_n^{set} \}_{n \in N},$$

де fb_i^{set} – множина повідомлень зворотного зв'язку, які прив'язані до обмеження $c_i, 1 \leq i \leq n, 0 \leq \# fb_i^{set} \leq n_i, n_i \in N$.

Після введення впорядкування множини обмежень за ступенем релевантності до проблеми отримаємо систему обмежень для k -го стану навчальної проблеми

$$ConstrSyst_{LPS_k} = \{ ConstrSet_{LPS_k, \geq rd} \}_{rd \in [0..1]},$$

яка також може бути представлена у вигляді

$$ConstrSyst_{LPS_k} = \{ c_1 : rd_1, c_2 : rd_2, \dots, c_l : rd_l \}_{l \in N}, \text{ де } rd_i \in [0..1], i_1 \in [0..1].$$

Процедура формування зворотного зв'язку для системи обмежень стану навчальної проблеми LPS_k матиме вигляд:

1. $FB_{LPS_k}^{max} = \bigcup_{k=1..l} fb_k^{set}, 1 \leq k \leq l$, де $FB_{LPS_k}^{max}$ – максимально можливий потік зворотного зв'язку

для стану LPS_k (тобто для граничного випадку, що відповідає порушенню всіх обмежень в системі). Проте очевидно, що буде порушуватись тільки деяка підмножина обмежень $[ConstrSyst_{LPS_k}]^V \subseteq ConstrSyst_{LPS_k}$. Тоді відповідно матимемо:

2. $FB_{LPS_k}^V = \bigcup_{\forall k_1} fb_{k_1}^{set}, \forall k_1, / \neq c_{k_1}$. На рівні навчальної проблеми відповідно матимемо:

$FB_{LP_j}^{max} = \bigcup_{k=1..k_{max}} FB_{LPS_k}^{max}, FB_{LP_j}^V = \bigcup_{k=1..k_{max}} FB_{LPS_k}^V$, де $FB_{LPS_{k_1}}^V \subseteq FB_{LPS_{k_1}}^{max}, k_1 = 1..k_{max}$. На рівні навчального

курсу: $FB_{LC_i}^{max} = \bigcup_{j=1..j_{max}} FB_{LP_j}^{max}, FB_{LC_i}^V = \bigcup_{j=1..j_{max}} FB_{LP_j}^V$, де $FB_{LP_{j_1}}^V \subseteq FB_{LP_{j_1}}^{max}, j_1 = 1..j_{max}$.

При формуванні бази знань в системі до кожного стану навчальної проблеми вказують множину помилок, яка може виникнути при вирішенні певного стану навчальної проблеми:

$$Err_k^{set} = \{ err_1, err_2, \dots, err_n \}_{n \in N}.$$

У технологічному профілі помилки розглядаються як позаштатні надзвичайні ситуації технологічного процесу (буріння нафти і газу). Проте в обох випадках виникнення помилок розглядається як наслідок порушення певних обмежень, їх множин та ієрархій:

$$\{ [ConstrSet_k]^V \neq \emptyset \} \Rightarrow [Err_k^{set} \neq \emptyset]$$

При введенні в базу знань, для помилок можуть вказуватись вагові коефіцієнти (ew – *error weight*). Введення відношення впорядкування на множині помилок дасть змогу отримати систему помилок

$$Err_k^{Syst.} = (Err_k^{set, \geq ew}) \text{ або } Err_k^{Syst.} = \{ err_1 : ew_1, err_2 : ew_2, \dots, err_n : ew_n \}_{n \in N}$$

Відповідно можна отримати ієрархію помилок до навчальної проблеми

$$Err_{LP_j}^{Hrch.} = \bigcup_{k=1..k_{max}} Err_k^{Syst.}.$$

При введенні помилок в базу знань і їх прив'язки до стану навчальної проблеми задається також їх прив'язка до системи обмежень накладеної на стан навчальної проблеми:

$$LPS_k : Err_k^{Syst.} \rightarrow ConstrSyst_k, \left\{ \begin{array}{l} \{ err_1, err_2, \dots, err_{n_1} \} \\ \{ c_1, c_2, \dots, c_{n_2} \} \end{array} \right., n_1, n_2 \in N.$$

Відповідно можна виконати ряд впорядкувань системи обмежень стосовно ступеня релевантності помилкам:

$$\left\{ \begin{array}{l} err_1 : \{ c_1 : rd_1^{err_1}, c_2 : rd_2^{err_1} \dots c_{n_2} : rd_{n_2}^{err_1} \} \\ err_2 : \{ c_1 : rd_1^{err_2}, c_2 : rd_2^{err_2} \dots c_{n_2} : rd_{n_2}^{err_2} \} \\ \dots \\ err_{n_1} : \{ c_1 : rd_1^{err_{n_1}}, c_2 : rd_2^{err_{n_1}} \dots c_{n_2} : rd_{n_2}^{err_{n_1}} \} \end{array} \right.$$

Нехай для обмеження c_m умовою релевантності є вираз $rc_m : cf_m^{rc}$, а умовою задоволення вираз $sc_m : cf_m^{sc}$. Тоді для обмеження c_m можна ввести додаткову мітку R_m , $R_m = [rc_m : cf_m^{rc}] \models [sc_m : cf_m^{sc}]$. На рівні навчальної проблеми, множина правил матиме вигляд: $Rules_{LP_j}^{set} = \{ R_m \}_{m=1 \dots C_{max}}$, де C_{max} – кількість обмежень в системі $ConstrSyst_{LP_j}$.

Означення 1. Статичною навчальною проблемою на основі обмежень будемо вважати кортеж $LP_j^{stat.} = (Err_{LP_j}^{Hrch.}, \bigcup_{k=1..k_{max}} LPS_k, ConstrHrch_{LP_j})$, де $Err_{LP_j}^{Hrch.}$ – ієрархія помилок до навчальної проблеми, LPS_k – k -й стан навчальної проблеми, $ConstrHrch_{LP_j} = \bigcup_{k=1..k_{max}} ConstrSyst_k$ – ієрархія обмежень j -тої навчальної проблеми, $ConstrSyst_k$ – система обмежень для k -го стану.

У режимі функціонування при вирішенні навчальних проблем кожне з обмежень у відповідній системі обмежень буде або задовольнятися, або порушуватися, тому отримуватимемо підмножини задоволених та порушених обмежень:

$$[ConstrSet_k]^S = \{ c_1, c_2, \dots, c_{m_k} \}_{m_k \in N}, [ConstrSet_k]^V = \{ c_1, c_2, \dots, c_{n_k} \}_{n_k \in N}.$$

Застосування впорядкування $\leq rd$ дасть змогу утворити відповідні системи задоволених та порушених обмежень: $[ConstrSyst_k]^S = \{ c_1 : rd_1, c_2 : rd_2, \dots, c_{m_k} : rd_{m_k} \}_{m_k \in N}$ $[ConstrSyst_k]^V = \{ c_1 : rd_1, c_2 : rd_2, \dots, c_{n_k} : rd_{n_k} \}_{n_k \in N}$, де $rd_t \in \{ rd_{i_t} \}_{i_t=1..l}, t \in [1..m_k]$.

Означення 2. Динамічною навчальною проблемою на основі обмежень будемо вважати кортеж $LP_j^{dyn.} = (Err_{LP_j}^{Hrch.}, \bigcup_{k=1..k_{max}} LPS_k, [ConstrHrch_j], [ConstrHrch_j^V])$, де $Err_{LP_j}^{Hrch.}$ – ієрархія помилок до навчальної проблеми, LPS_k – k -й стан навчальної проблеми, $[ConstrHrch_j]^S = \bigcup_{k=1..k_{max}} [ConstrSyst_k]^S$ – ієрархія задоволених обмежень для j -ї навчальної проблеми; $[ConstrSyst_k]^S$ – система задоволених обмежень для k -го стану; $[ConstrHrch_j]^V = \bigcup_{k=1..k_{max}} [ConstrSyst_k]^V$ – ієрархія порушених обмежень для j -ї навчальної проблеми; $[ConstrSyst_k]^V$ – система порушених обмежень для k -го стану.

Введена структура навчальної проблеми на основі обмежень може бути графічно представлена у деревоподібній формі (рис. 1).

Нехай при вирішенні стану навчальної проблеми LPS_k виникає певна кількість помилок, яка відповідає кортежу $(err_1, err_2, \dots, err_{k_1})$, де $k_1 \in [0..n]$, n – кількість елементів в $Err_k^{Syst.}$. Пропоноване рішення об'єктом навчання $SubmittedSolution$ оцінюється інтелектуальною системою з погляду набору помилок, що міститься у цьому рішенні $SubmittedSolution \models Err_k^{sol.} = (err_1, err_2, \dots, err_{k_1})$.

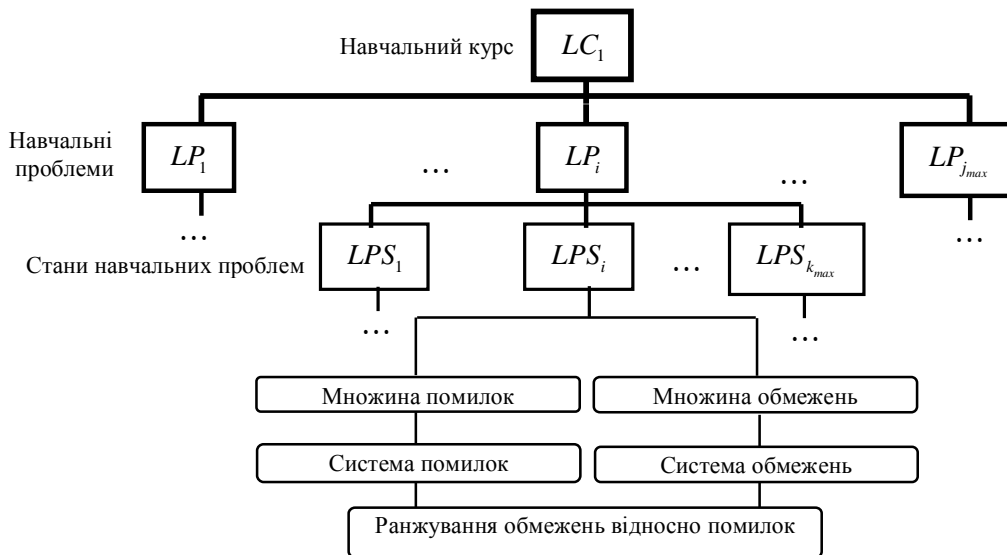


Рис. 1. Деревоподібна структура навчальних (технологічних) проблем

Означення 3. Присвоєнням на множині навчальних проблем (набором помилок пропонує ршень навчальної проблеми – SubmittedSolution) $LP^{set} = \{LP_j\}_{j=1..jmax}$ вважатимемо відображення Y^{LP_j} з множини змінних $Err_1 \hat{I} Err_{LP_j}^{Hrch.}$ в їх відповідні домени, тобто $Y^{LP_j}(err_m) \in D_m$ для $err_m \hat{I} Err_1$, $\{D_1 \times D_2 \times \dots \times D_m\} \subset \{LPS_k\}_{k=1..kmax}$. Множину всіх можливих присвоєнь для Err_1 позначатимемо $Y_{Err_1}^{LP_j}$.

Застосування присвоєння Y^{LP_j} в домені Err_1 для кортежів $(err_1, err_2, \dots, err_{k_1})$ дасть змогу отримати присвоєння $\{(err_1, d_{err_1}), \dots, (err_{k_1}, d_{err_{k_1}})\}$ і отримати кортеж значень $Y^{LP_j} = (d_{err_1}, \dots, d_{err_{k_1}})$, тобто кортеж значень помилок з інтерпретацією $d_{err_p} \rightarrow \{fb_p, R_p\}_{p \in \{1..k_1\}}$.

Означення 4. Присвоєння Y^{LP_j} вважатимемо повним, якщо відповідні йому домени значень $D_m, m_1 \in [1..m]$ не є порожніми для всіх станів навчальної проблеми LPS_k . В іншому випадку присвоєння вважатимемо частковим.

Означення 5. Множину всіх присвоєнь, означених для домену $Err_{LP_j}^{Hrch.}$ $Y_{Err_{LP_j}^{Hrch.}}^{LP_j} \equiv D_1 \times \dots \times D_{n_1}$, називатимемо простором інтерпретації помилок рішень навчальної проблеми LP_j .

Будемо розглядати присвоєння виду $Y^{LP_j} = (err_1, err_2, \dots, err_{k_1})$. Нехай стосовно обмеження $c_m \in ConstrHrch_{LP_j}$ цей набір помилок має ранжування виду $(err_1 : rd_1^{c_m}, err_2 : rd_2^{c_m}, \dots, err_{k_1} : rd_{k_1}^{c_m})$. Обчислимо середнє значення релевантності множини помилок з присвоєння обмеженню c_m :

$$AvgErr^{c_m} = (rd_1^{c_m} + rd_2^{c_m} + \dots + rd_{k_1}^{c_m}) / k_1$$

Означення 6. Будемо вважати, що обмеження c_m задовольняється присвоєнням Y^{LP_j} (позначатимемо це через $Y^{LP_j} \models^{SD^{c_m}} c_m$) зі ступенем $sd^{c_m} = 1 - AvgErr^{c_m}$. Відповідно обмеження c_m порушується присвоєнням Y^{LP_j} зі ступенем $vd = AvgErr^{c_m}$.

Часткове присвоєння Y^{LP_j} будемо вважати послідовним, якщо всі релевантні йому обмеження порушуються. Розглянемо ієрархію обмежень для навчальної проблеми LP_j

$$\text{ConstrHrch}_{LP_j} = \mathbf{U}_{k=1..k_{max}} \text{ConstrSyst}_k = \begin{pmatrix} c_1, c_2, \dots, c_{n_1} \\ c_1, c_2, \dots, c_{n_2} \\ \dots \\ c_1, c_2, \dots, c_{n_k} \\ \dots \\ c_1, c_2, \dots, c_{n_{k_{max}}} \end{pmatrix}.$$

Кожне присвоєння $y \in Y^{LP_j}$ задовольняє кожне обмеження $c_m \in \text{ConstrHrch}_{LP_j}$ з деяким ступенем задоволення $sd_y^{c_m}$ згідно з означенням 6. Тоді можна обчислити ступінь задоволення

присвоєння всієї ієрархії обмежень навчальної проблеми: $sd_y^{\text{ConstrHrch}_{LP_j}} = \left(\sum_{k=1}^{k_{max}} sd_y^{\text{ConstrSyst}_k} \right) / k_{max}$, де

$$sd_y^{\text{ConstrSyst}_k} = \left(\sum_{m=1}^{n_k} sd_y^{c_m} \right) / n_k.$$

Означення 7. Рішення $Sol_{Err^{set}}$ з множиною помилок Err^{set} будемо вважати коректним (правильним) рішенням навчальної проблеми LP_j із ступенем коректності $sd = sd_y^{\text{ConstrHrch}_{LP_j}}$ якщо деяке присвоєння $y \in Y^{LP_j}$, $y \subseteq Err_{Sol}^{set}$ задовольняє всі обмеження із ступенем $sd_y^{\text{ConstrHrch}_{LP_j}}$.

Означення 8. Обмеження $c_m \in \text{ConstrHrch}_{LP_j}$ навчальної проблеми LP_j будемо вважати послабленим, якщо в нього додано додаткове правило $R_m^{Add.} = [rc_m^{Add.} : cf_m^{RC^{Add.}}] \models [sc_m^{Add.} : cf_m^{SC^{Add.}}]$, тобто стосовно виділеного обмеження йдеться про додавання додаткової умови релевантності (rc – *relevancy condition*) та додаткової умови задоволення (sc – *satisfaction condition*) з відповідними коефіцієнтами визначеності (cf – *certainty factor*). Відповідно, якщо подібне правило можна видалити з обмеження, то тоді обмеження вважатимемо стиснутим.

Розпізнавання помилок виконується на основі наявних декларативних (описових) знань в об'єкті навчання. Теорія навчання на основі помилок використовує зворотний зв'язок для здійснення адекватних поточних, короткотривалих, довготривалих коректуючих впливів на основі виділення хибних (помилкових) знань в контексті аналізу помилок:

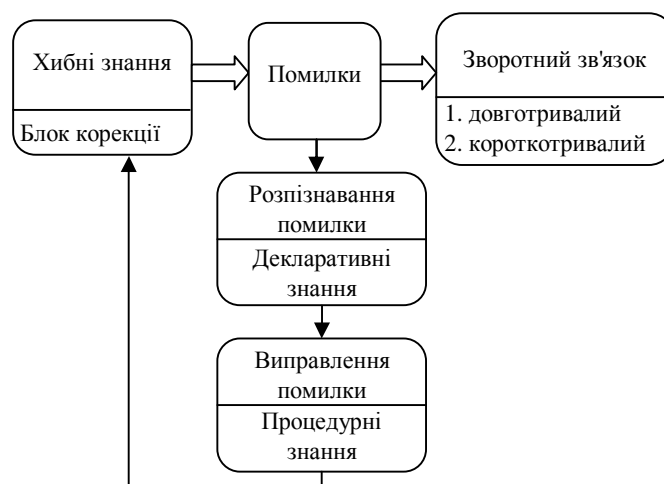


Рис. 2. Структура розпізнавання та виправлення помилок

Моделювання на основі обмежень [4, 5] є підходом до моделювання об'єкта навчання, що базується на теорії навчання на основі помилок. Основа даного підходу полягає в тому, що всі

коректні рішення (*CorrectSolution*) проблеми розглядаються як схожі (подібні). Їх схожість полягає в тому, що вони не порушують жодне обмеження в накладеній на домен множині обмежень *ConstrSet*. Розглядаються можливі послідовності станів в просторі проблеми (*LPS – LearningProblemStates*): $LearningProblemSpace = \{LPS_k\}_{k=1..k_{max}}, \{\dots LPS_{k_1} \dots LPS_{k_2} \dots\}_{k_1, k_2 \in [1..k_{max}]}$.

Виконуючи рішення навчальної проблеми, об'єкт навчання переміщується по певній послідовності станів із множини можливих і в кожен момент часу знаходиться в певному виділеному стані. Накладання множини обмежень дозволяє виділити класи еквівалентностей для множини станів проблеми. В межах класу еквівалентності система використовує одну і ту саму структуру та наповнення зворотного зв'язку, тому всі стани проблеми в межах класу еквівалентності розглядаються як тотожні з точки зору формування процедури зворотного зв'язку:

$$\{Class_m^E\}_{m=1..m_{max}}, Class_{m_1}^E = \{\dots LPS_{m_2} \dots LPS_{m_3} \dots\}_{m_1 \in [1..m_{max}], m_2, m_3 \in [1..k_{max}]};$$

Тому доцільним є прикріплення повідомлень зворотного зв'язку (*fb – feedback message*) безпосередньо до обмежень на рівнях множини обмежень *ConstrSet*, системи обмежень *ConstrSyst* та ієрархії обмежень *ConstrHrch*

$$ConstrHrch_{LC_i} = \bigcup_{j=1..j_{max}} ConstrSyst_{LP_j} = \bigcup_{k=1..k_{max}} ConstrSet_{LPS_k} = \{c_1 : fb_1, c_2 : fb_2 \dots c_n : fb_n\}_{k, n \in N}$$

Отже, модель домену *DomainModel* можна розглядати як сукупність (множину) правил: **якщо** $\langle \text{умова_релевантності} \rangle$ істинна з коефіцієнтом cf_1 , **то** $\langle \text{умова_задоволення} \rangle$ істинна з коефіцієнтом cf_2 , де $cf_1, cf_2 \in [0..1]$ і відповідно виражають коефіцієнт релевантності або ваговий коефіцієнт, ймовірнісний коефіцієнт, можливісний коефіцієнт або коефіцієнт преференції. У випадку $cf_1 = cf_2 = 1$ матимемо класичну інтерпретацію моделі домену в термінах абсолютної істинності та хибності.

Тобто, якщо рішення запропоноване об'єктом навчання відноситься до стану проблеми, визначеного умовою релевантності (абсолютно або з певним коефіцієнтом cf), то воно буде коректним (абсолютно або з певним коефіцієнтом cf) якщо воно також знаходиться в стані, що визначається умовою задоволення (абсолютно або з певним коефіцієнтом cf):

$$SubmittedSolution[LPS_{k_1}]^{rc:cf_1} \Rightarrow SubmittedSolution[LPS_{k_2}]^{sc:cf_2},$$

$$k_1 = k_2 \text{ або } k_1 \langle \rangle k_2, \text{ якщо } LPS_{k_1}, LPS_{k_2} \in Class_{m_1}^E; k_1, k_2 \in [1..k_{max}], m_1 \in [1..m_{max}]$$

де $rc : cf_1$ – умова релевантності (абсолютна або відносна), $sc : cf_2$ – умова задоволення (абсолютна або відносна).

Отже, моделі доменів визначатимуться відповідними множинами обмежень:

$$DomainModel_{LPS_k} = \{ConstrSet_k\}_{k \in [1..k_{max}]} = \{if [rc_i : cf_{i_1}] then [sc_i : cf_{i_2}]\}_{i_1, i_2 \in N}$$

Інтелектуальні системи на основі обмежень виконують оцінку рішень запропонованих об'єктом навчання шляхом зіставлення їх з відповідною доменною моделлю: $Match(SubmittedSolution, DomainModel)|_{ConstrSet}$, що визначається накладеною множиною обмежень.

Дана процедура виконується як послідовність кроків:

0. виділення шаблонів умов релевантності $[rc_{i_1} : cf_{i_1}].Pattern$ і шаблонів умов задоволення $[sc_{i_2} : cf_{i_2}].Pattern$ для $i_1, i_2 \in N$;

1. Зіставлення шаблонів релевантності із відповідним станом проблеми $Match([rc_{i_1} : cf_{i_1}].Pattern, LPS_k)_{i_1, k \in N}$;

2. Перевірка умови задоволення релевантних обмежень $c_{m_1}.[sc_{m_1} : cf_{m_1}], c_{m_1} \in ConstrSet_{LPS_k}$;

3. Обмеження вважається задоволеним, якщо шаблон його умови задоволення відповідає стану проблеми $\models c_{m_1} \text{ iff } c_{m_1}.[sc_{m_1} : cf_{m_1}] =_{Pattern} LPS_k$;

4. Обмеження вважається порушеним, якщо шаблон його умови задоволення не відповідає стану проблеми $\not\models c_{m_1} \text{ iff } c_{m_1}.[sc_{m_1} : cf_{m_1}] \neq_{Pattern} LPS_k$.

На основі введених означень виконується побудова поточної моделі об'єкта навчання, ($CM - CurrentModel$), короткотривалої моделі ($STM - Short-term model$) та довготривалої ($LTM - Long-term model$).

Означення 9. Під поточною моделлю будемо розуміти певну послідовність порушених обмежень при вирішенні k -того стану навчальної проблеми: $CM_{LPS_k} = \{\dots c_i^v, c_{i+1}^v \dots\}_{i \in N}$, де $c_i^v \in [ConstrSet_{LPS_k}]^V$.

Означення 10. Під короткотривалою моделлю будемо розуміти об'єднання підмножини задоволених та підмножини порушених обмежень: $STM_{LPS_k} = [ConstrSet_{LPS_k}]^S \cup [ConstrSet_{LPS_k}]^V$

Означення 11. Під довготривалою моделлю об'єкта навчання будемо розуміти об'єднання короткотривалих моделей:

$$\begin{cases} LTM_{LP_j}^1 = \bigcup_{k=1..k_{max}} STM_{LPS_k} \\ LTM_{LC_i}^2 = \bigcup_{j=1..j_{max}} LTM_{LP_j}^1 \end{cases}$$

Декларативні знання, як необхідна складова організації процесу комп'ютерного тьюторингу, дозволяють об'єкту навчання виконувати ідентифікацію помилок. Тому якщо такі знання відсутні для певного домену предметної області або відповідно не є достатніми, то ідентифікація помилок об'єкту навчання буде ускладнюватись, оскільки об'єкт навчання не зможе розрізняти хибні та релевантні знання і відповідно здійснювати ефективну корекцію власних помилок (рис. 3).

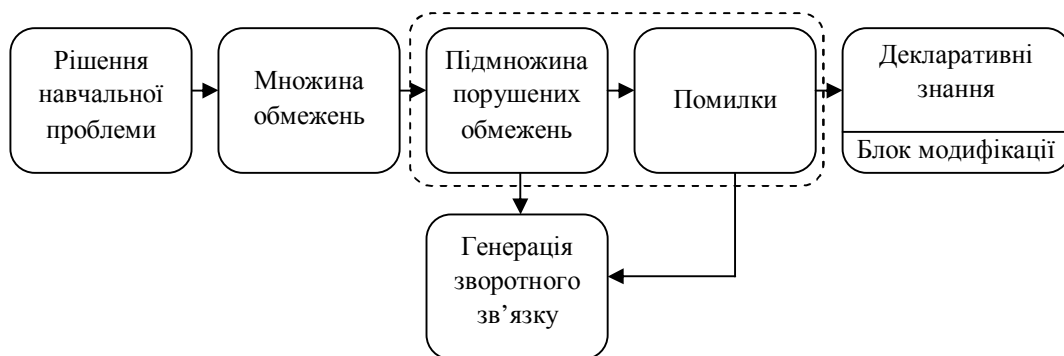


Рис.3. Співвідношення множини помилок і множини обмежень

Використання систем обмежень для навчальних проблем дозволяє виконувати аналіз помилок, їх класифікацію та формувати відповідні коректуючі впливи зворотного зв'язку, що забезпечує ефективне та адекватне розширення множини декларативних знань об'єкту навчання. Як видно з рис.3 порушені обмеження (множина порушених обмежень) є джерелом генерації зворотного зв'язку, оскільки відповідні повідомлення зворотного зв'язку прив'язуються безпосередньо до обмежень, порушення яких вказує системі на необхідність включення даних повідомлень в діалог зворотного зв'язку, що формується. Таким чином механізм функціонування системи полягатиме в співставленні рішень об'єкта навчання з множинами (системами, ієрархіями) обмежень, що описують поточну навчальну проблему та відповідно дозволяють виконувати оцінку рішення об'єкта навчання: $Match(SubmittedSolution, ConstrHrch[ConstrSyst[ConstrSet]])$.

Реалізація інтелектуальних функцій системи полягає в такому способі організації взаємодії з об'єктом навчання який би виконував активацію релевантних доменних знань для об'єкта навчання відповідно до профілю його зони найближчого розвитку ($PDZ - ProximalDevelopmentZone$) $PDZ \subset DomainModel$. Тому ефективність організації зворотного зв'язку в інтелектуальній системі може бути оцінена по таких критеріях: 1) вказання місця виникнення помилки (явне або неявне); 2) ідентифікація та класифікація помилки, співвіднесення її до певної ієрархії помилок, виділення складових помилки та її можливих рівнів; 3) виділення релевантних концепцій предметної області як основи побудови коректного рішення: $[DomainConception^{set}]^R \xrightarrow{ConstrSet} CorrectSolution^{set}$.

Таким чином, кожному обмеженню присвоюється одне або кілька наперед визначених повідомлень зворотного зв'язку, які використовуються при формуванні діалогів зворотного зв'язку як основи коректуючих впливів системи при порушенні певних обмежень:

$$ConstrSet \rightarrow \{ c_{i_j} : \{ fb \}_{j_i} \}_{i_j=1,2,\dots,n_1, j_i=0,1,\dots,n_2, n_1, n_2 \in N}, \text{FeedBack} \leftarrow \{ \{ c_k^v \}_{j_k}^{Jset} \}_{i_j}^{Syst.} \}_{i,j,k \in N}^{Hrch.}$$

Існуючі інтелектуальні системи, як правило, забезпечують визначену кількість рівнів зворотного зв'язку. Найбільш вживаними та ефективними рівнями є: 1) рівень I – зворотний зв'язок виду "так", "ні", "вірно", "хибно", "так:cf", "ні:cf", "вірно:cf", "хибно:cf", де $cf \in [0..1]$, $cf \rightarrow \{ релевантність, вагове значення, імовірнісне значення, можливісне значення, перевага \}$; 2) рівень II – використання методу прапорця помилок (*ErrorFlag*), для індикації місця виникнення помилки без виведення точного опису причини помилки; 3) рівень III – виведення повідомлення про суть виділеної помилки, що є наслідком порушення певного обмеження (*SingleErrorFeedback*); 4) рівень IV – виведення частини правильного рішення (*PartialSolutionFeedback*); 5) рівень V – виведення в якості зворотного зв'язку повного правильного рішення (*CompleteSolutionFeedback*); 6) рівень VI – виведення в якості зворотного зв'язку всіх помилок поточної сесії роботи з системою (*AllErrorsFeedback*); 7) рівень VII – візуалізація ієрархії помилок (*ErrorsHierarchyFeedback*) та візуалізація ієрархії порушених обмежень (*ViolatedConstraintsHierarchy*).

Як показує дослідження в даній області [3, 4], визначальне значення для побудови ефективного зворотного зв'язку має семантика та стилістика повідомлень, що будуть виводитися, їх доступність для розуміння поточним користувачам, тобто їх відповідність його зоні найближчого розвитку (*PDZ*). Очевидно, що в термінах задоволення та порушення обмежень, можемо розглядати концепцію очікуваної зони найближчого розвитку (*PDZ^{exp.}*):

$$PDZ^{exp.} \cong [DomainModel = ConstrSet] \setminus [LTM = \bigcup_{i_2} STM = \bigcup_{i_1} CM \}_{i_1, i_2}]$$

Нехай маємо навчальну проблему LP_i із заданою кількістю станів LPS_k . Ієрархія обмежень навчальної проблеми матиме вигляд:

$$ConstrHrch_{LP_i} = \left\{ \begin{array}{l} ConstrSyst_{LPS_1} = \{ c_1^1 : rd_1^1, c_2^1 : rd_2^1, \dots, c_{n_1}^1 : rd_{n_1}^1 \} \\ ConstrSyst_{LPS_2} = \{ c_1^2 : rd_1^2, c_2^2 : rd_2^2, \dots, c_{n_2}^2 : rd_{n_2}^2 \} \\ \dots \\ ConstrSyst_{LPS_k} = \{ c_1^k : rd_1^k, c_2^k : rd_2^k, \dots, c_{n_k}^k : rd_{n_k}^k \} \\ \dots \\ ConstrSyst_{LPS_{k_{max}}} = \{ c_1^{k_{max}} : rd_1^{k_{max}}, c_2^{k_{max}} : rd_2^{k_{max}}, \dots, c_{n_{k_{max}}}^{k_{max}} : rd_{n_{k_{max}}}^{k_{max}} \} \end{array} \right. ,$$

де $k = [1..k_{max}]$.

Відповідно до кожної системи обмежень можна обчислити середнє значення переваг обмежень $rd_k = (rd_1^k + rd_2^k + \dots + rd_{n_k}^k) / n_k$. Визначимо також середній крок зміни переваг для системи обмежень $ConstrSyst_{LPS_k}$ за формулою $rd_{step}^k = (rd_{n_k}^k - rd_1^k) / n_k$.

Виконаємо обчислення зони найближчого розвитку об'єкта навчання для стану LPS_k . Нехай стану LPS_k відповідає система обмежень $ConstrSyst_{LPS_k} = \{ c_1^k : rd_1^k, c_2^k : rd_2^k, \dots, c_{n_k}^k : rd_{n_k}^k \}$ і останнім задоволеним обмеженням є обмеження $c_{i_1}^k : rd_{i_1}^k$, тоді:

$$ConstrSyst_{LPS_k} = \{ \overbrace{c_1^k : rd_1^k, \dots, c_{i_1}^k : rd_{i_1}^k}^{PDZ_{LPS_k}^{max}}, \dots, c_{n_k}^k : rd_{n_k}^k \} \text{ або } PDZ_{LPS_k}^{max} = \{ ConstrSyst_{LPS_k} \setminus \{ c_{i_1}^k : rd_{i_1}^k \}_{i_1=1..n_k} \},$$

де n_k – кількість обмежень в системі обмежень $ConstrSyst_{LPS_k}$.

Значення $PDZ_{LPS_k}^{max}$ інтерпретуються як максимально повна зона найближчого розвитку для стану LPS_k . Відповідно можна ввести подібне обчислювальне значення з інтерпретацією мінімально достатньої зони найближчого розвитку об'єкта навчання:

$$ConstrSyst_{LPS_k} = \{ c_1^k : rd_1^k, \dots, c_{i-1}^k : rd_{i-1}^k, \overbrace{c_i^k : rd_i^k}^{PDZ_{LPS_k}^{min}}, \dots, c_{n_k}^k : rd_{n_k}^k \}$$

$$PDZ_{LPS_k}^{min} = \{ ConstrSyst_{LPS_k} \setminus \{ c_{i_3}^k : rd_{i_3}^k \}_{i_3=\{1..i-1\}} \cup \{ c_{i_4}^k : rd_{i_4}^k \}_{i_4=\{i..n_k\}} \} \text{ або } PDZ_{LPS_k}^{min} = \{ c_{i-1}^k : rd_{i-1}^k \}, \neq c_i.$$

Значення $PDZ_{LPS_k}^{min}$ може бути обчислене також на основі введеного означення для середнього кроку зміни переважень: $PDZ_{LPS_k}^{min} = \{ c_m^k \}_{m \in \{1..n_k\}}, rd_m^k = rd_i^k + rd_{step}^k, \neq c_i$. Шляхом об'єднання отриманих значень найближчого розвитку для станів навчальної проблеми, отримаємо значення для всієї навчальної проблеми і навчального курсу відповідно:

$$PDZ_{LP_j}^{max} = \bigcup_{k=1..k_{max}} PDZ_{LPS_k}^{max}, \quad PDZ_{LP_j}^{min} = \bigcup_{k=1..k_{max}} PDZ_{LPS_k}^{min}$$

$$PDZ_{LC_i}^{max} = \bigcup_{j=1..j_{max}} PDZ_{LP_j}^{max}, \quad PDZ_{LC_i}^{min} = \bigcup_{j=1..j_{max}} PDZ_{LP_j}^{min}$$

Висновки

Таким чином, досліджувана функціональність навчальних проблем на основі обмежень зводиться до виділення сукупності їх станів з введеними множинами можливих помилок рішень. Профіль об'єкта навчання формується як ієрархія навчальних проблем, що представляються статично або динамічно. Кожен із станів навчальної проблеми описується власною множиною обмежень які пізніше впорядковуються до рівня систем та ієрархій. Функціональність зворотного зв'язку системи з користувачем забезпечується прив'язкою до обмежень релевантних повідомлень, що дозволяє системі інтерпретувати виникнення помилок як наслідок порушення обмежень. Множини помилок в свою чергу впорядковуються у вигляді систем та ієрархій з прив'язкою до виділеної навчальної (технологічної) проблеми. Помилки також прив'язуються до відповідних систем обмежень, які можуть бути перевпорядковані по ступеню релевантності. Розгляд системою тільки релевантних обмежень відносно станів навчальної проблеми дає змогу розглядати їх як такі, що порушуються або задовольняються в кожному присвоєнні. Відповідно кожне рішення, що пропонується об'єктом навчання оцінюється з погляду набору можливих помилок, що містяться в ньому. Як присвоєння на множині навчальних проблем розглядаються набори помилок з пропонуванних рішень, що дає змогу формувати простір інтерпретації помилок у вигляді множини присвоєнь. Введена інтерпретація процесу порушення та задоволення обмежень з певним ступенем, дозволяє в кінцевому підсумку оцінювати коректність та повноту рішення навчальної проблеми з певною множиною помилок.

Подальші дослідження даного напрямку будуть спрямовані на імплементацію отриманого формально-логічного представлення в інтелектуальній системі навчальних та технологічних тренінгів.

1. Tsang E. *Foundations of Constraint Satisfaction*. – London and San Diego: Academic Press. – 1993. – 421p.
2. Barták R., Dechter R. *Constraint Processing*. – Morgan Kaufmann Publisher. – 2003. – 210p.
3. Ohlsson S. *Constraint-based student modeling* // In: Greer J.E., McCalla G. (Eds): *Student modeling: the key to individualized knowledge-based instruction*. – 1994. – P.167–189.
4. Ohlsson S., Mitrovic A. *Constraint-Based Knowledge Representation for Individualized Instruction*. – University of Illinois at Chicago. – 1998. – 22 p.
5. Mitrovic, A. *Experiences in Implementing Constraint-Based Modeling in SQL-Tutor*. – *Proceedings of ITS'98*. – P. 414–423.
6. Вовк Р.Б., Шекета В.І. *Формалізація сумарної моделі студента в інтелектуальній тьюторній системі на основі обмежень* // *Вісник Хмельницького національного університету. Технічні науки* – 2009., том 2., – С. 242 – 248.
7. Вовк Р.Б. *Модель навчальної системи на основі підходу задоволення обмежень* // *Системний аналіз та інформаційні технології: Матеріали XI Міжнародної науково-технічної*

конференції (26-30 травня 2009р., Київ). – К.: НТУУ “КПІ”, 2009. – С. 276. 8. Вовк Р.Б., Криль Л.Р., Королик В.М. Дослідження структури зворотного зв'язку тьюторної системи на основі обмежень // *Радиоэлектроника и молодежь в XXI веке: Материалы XIII Международного молодежного форума (30 марта – 1 апреля 2009 г., Харьков)*. – Х.: ХНУРЕ, 2009. – ч.2 – С. 167.

УДК 681.3:519.15

В. Різник, Д. Скрибайло-Леськів
Національний університет «Львівська політехніка»,
кафедра автоматизованих систем управління

СИНТЕЗ ЗАВАДОСТІЙКИХ КОДІВ НА ОСНОВІ БАГАТОПОЗИЦІЙНИХ ІДЕАЛЬНИХ КІЛЬЦЕВИХ В'ЯЗАНОК

© Різник В., Скрибайло-Леськів Д., 2011

Досліджуються методи побудови завадостійких кодів на основі багатопозиційних комбінаторних конфігурацій з кільцевою структурою типу ідеальних кільцевих в'язанок (ІКВ) для створення систем кодування, які виявляють та виправляють помилки, з поліпшеними якісними показниками за потужністю та завадостійкістю. Здійснено порівняльний аналіз параметрів запропонованого коду з параметрами загальновідомих кодів.

Ключові слова: код, кільцева структура, потужність коду, завадостійкість коду, ідеальна кільцева в'язанка.

This paper considers some ways of correcting code design based on the combinatorial configurations with ring structure, named Ideal Ring Bundles (IRB)s for create error protection coding systems with improved quality factors with respect to code size and noise immunity. The comparative analyze of parameters for proposed code and well – known codes is realized.

Keywords: code, ring structure, code size, code immunity, ideal ring bundle.

Вступ

Методи комбінаторної оптимізації широко застосовують в інформаційних технологіях під час кодування, перетворення та пересилання даних, в інформаційно-вимірювальній та обчислювальній техніці, радіотехніці, зв'язку та суміжних галузях науки і техніки. Приклади постановки таких задач: проектування завадостійких систем кодування, розроблення ефективних технологій захисту цінних паперів від несанкціонованого доступу, удосконалення компонентів комп'ютерних систем, проектування радіосистем з високою роздільною здатністю. Тому актуальними є створення систем кодування з поліпшеними якісними показниками за потужністю та швидкодією щодо виявлення і виправлення помилок в повідомленнях, що передаються каналами зв'язку. На порядку денному – дослідження комбінаторних властивостей ідеальних кільцевих в'язанок з метою побудови багатопозиційних кодів для створення інформаційних технологій та пристроїв комп'ютерної техніки з високими якісними показниками – такими, як забезпечення достовірності інформації, підвищення надійності, функціональної безпеки і живучості інформаційних та інформаційно-управляючих систем.

Аналіз загальновідомих кодів

Циклічні коди, які спроможні виправляти більше двох помилок, в літературі відомі як коди Боуза, Чоудхурі, Хоквінхема – авторів методики побудови цих кодів (скорочено – коди БЧХ). Суть методики полягає в побудові твірного полінома, вигляд якого залежить від двох параметрів: довжини кодового слова S та максимально можливої кількості виправлених помилок t . Решта параметрів, що беруть участь у побудові твірного полінома, можуть визначатися за допомогою