

Разработка Алгоритма Вычисления Структурных Элементов Нечетких Временных Сетей Петри

Валех Мустафаев¹, Шафахат Рагимов², Малахат Салманова³
^{1,2,3}Сумгаитский государственный университет, Азербайджан
³malaxat_70@mail.ru

Аннотация– Разработан алгоритм вычисления структурных элементов нечетких временных сетей Петри (НВСП). На основе предложенного алгоритма разработано программное обеспечение в среде Borland Delphi 7.0, с применением которого исследованы и разработаны сетевые модели управления.

Ключевые слова– алгоритм, трапециевидный нечеткий интервал, нечеткие временные сети Петри

I. ВВЕДЕНИЕ

При управлении объектов функционирующих в условиях неопределенности к ресурсам накладываются различные ограничения, в том числе временные. Эти ограничения соответствуют условиям изменения маркеров НВСП. Доступность маркеров входных и выходных позиций и время срабатывания переходов описываются нечеткими числами или интервалами, что усложняет вычисления структурных элементов сетевой модели. В связи с этим в докладе разработан алгоритм вычисления структурных элементов НВСП.

II. ПРЕДСТАВЛЕНИЯ МОДЕЛИ УПРАВЛЕНИЯ ПРОЦЕССОВ В ВИДЕ НВСП И ОПРЕДЕЛЕНИЕ ПРАВИЛ СРАБАТЫВАНИЯ ПЕРЕХОДОВ

Модели управления процессов представляется в виде НВСП.

Нечёткая временная сеть Петри определяется выражением [1].

$$C_T = (N, \mu_0, Z_T, S_T)$$

где $N = (P, T, I, O)$ – структура НВСП: $P = \{p_i\}$, ($i = 1, \dots, n$; n – число позиций) – множество позиций; $T = \{t_j\}$, ($j = 1, \dots, m$; m – число переходов) – множество переходов; $I: P \times T \rightarrow \{0,1\}$; $O: T \times P \rightarrow \{0,1\}$ – соответственно входная и выходная функция переходов;

$\mu_0 = (\mu_1^0, \mu_2^0, \dots, \mu_n^0)$ – вектор начальной маркировки, каждый компонент μ_i^0 который представляет собой трапециевидный нечеткий интервал (ТНИ)

$$\mu_i^0 = \langle \mu_{1i}^0, \mu_{2i}^0, \mu_{3i}^0, \mu_{4i}^0 \rangle (i \in \{1, 2, \dots, n\}).$$

ТНИ называется нормальный нечеткий интервал, функция принадлежности которого задается трапециевидной функцией f_T . ТНИ представляется в виде кортежа из четырех чисел: $A = \langle a, b, \alpha, \beta \rangle$, где a и b – соответственно нижнее и верхнее модальные значения ТНИ; α и β левый и правый коэффициенты нечеткости ТНИ.

$z = (z_1, z_2, \dots, z_n)$ – вектор параметров временных задержек маркеров в позициях НВСП, каждый компонент z_i который представляет собой ТНИ

$$z_i = \langle z_{1i}, z_{2i}, z_{3i}, z_{4i} \rangle, (i \in \{1, 2, \dots, n\}).$$

$s = (s_1, s_2, \dots, s_m)$ – вектор параметров времен срабатывания разрешенных переходов НВСП, каждый компонент s_j который представляет собой ТНИ

$$s_j = \langle s_{1j}, s_{2j}, s_{3j}, s_{4j} \rangle, (j \in \{1, 2, \dots, m\}).$$

Текущее состояние НВСП определяется вектором $\mu = (\mu_1, \mu_2, \dots, \mu_n)$, компоненты которого представляет собой ТНИ $\mu_i = \langle \mu_{1i}, \mu_{2i}, \mu_{3i}, \mu_{4i} \rangle$, ($i \in \{1, 2, \dots, n\}$) и интерпретируется как значения функции принадлежности нечеткого наличия одного маркера в соответствующих позициях $p_i \in P$ относительно времени, отсчитываемого от момента запуска данной НВСП.

Переход $t_k \in T$ НВСП называется разрешенным, при некоторой доступной маркировке μ , если выполнено следующее условие [2]:

$$\mu_i = \langle \mu_{1i}, \mu_{2i}, \mu_{3i}, \mu_{4i} \rangle > 0 / (\forall p_i \in P) \wedge (I(p_i, t_k) > 0).$$

Если переход $t_k \in T$ НВСП разрешен при некоторой доступной маркировке μ , то нечеткое срабатывание данного перехода, осуществляемое за время $s_k = \langle s_{1k}, s_{2k}, s_{3k}, s_{4k} \rangle$ приводит к новой маркировке μ_k , компоненты вектора которой определяется следующим образом:

для каждой из входных позиций $p_i \in P$ для которых $I(p_i, t_k) > 0$ по формуле:

$$\mu'_k = \langle \mu'_{1k}, \mu'_{2k}, \mu'_{3k}, \mu'_{4k} \rangle = \langle 0, 0, 0, 0 \rangle \quad (1)$$

$$/(\forall p_i \in P) \wedge (I(p_i, t_k) > 0)$$

для каждой из выходных позиций $p_j \in P$ для которых $O(t_k, p_j) > 0$, по формуле :

$$\mu'_j = \min\{\max\{\mu_i\} + s_k, \mu_j\} \quad (2)$$

$$(i \in \{1, 2, \dots, n\}) \wedge I(p_i, t_k) > 0)$$

$$(\forall p_j \in P) \wedge (O(t_k, p_j) > 0) \wedge (\mu_j \neq \langle 0, 0, 0, 0 \rangle),$$

где \max , \min – расширенная операция максимума и минимума для ТНИ, а “+” операция сложения ТНИ;

для каждой позиции $p_i \in P$, не являющейся входным и выходным по формуле:

$$\mu'_i = \langle \mu'_{1i}, \mu'_{2i}, \mu'_{3i}, \mu'_{4i} \rangle = \langle \mu_{1i}, \mu_{2i}, \mu_{3i}, \mu_{4i} \rangle$$

$$(\forall p_i \in P) \wedge (I(p_i, t_k) = 0) \wedge (O(t_k, p_j) = 0)$$

Если некоторые из позиций $p_j \in P$ являются одновременно входными и выходными для разрешенного перехода $t_k \in T$, то для них компоненты вектора новой маркировки рассчитываются последовательно, вначале по формуле (1), затем по формуле (2).

После нечеткого срабатывания перехода соответствующие маркеры становятся доступными только после окончания действия временных задержек, которые определяют доступную маркировку μ'_j по формуле:

$$\mu'_i = \mu_i + z_i / (\forall p_i \in P) \wedge (O(t_k, p_j) > 0) \wedge (\mu_j \neq \langle 0, 0, 0, 0 \rangle).$$

Для выходных позиций p_j перехода t_k , для которых $\mu_i = \langle 0, 0, 0, 0 \rangle$ по определению $\mu'_j = \langle 0, 0, 0, 0 \rangle$.

III. АЛГОРИТМ ВЫЧИСЛЕНИЯ СТРУКТУРНЫХ ЭЛЕМЕНТОВ НВСП

Начало алгоритма

Шаг 1. Создание входных и выходных матриц представляющих входную и выходную функции НВСП в размерности $m \times n : d_{ij}^-, d_{ij}^+ (i = \overline{1, n}; j = \overline{1, m})$.

Шаг 2. Определение размерности и ввод структурных элементов НВСП:

$$\mu_{ij}, z_{ij} (i = \overline{1, k}; j = \overline{1, n}); s_{ij} (i = \overline{1, k}; j = \overline{1, m}).$$

Шаг 3. Полагают: $k = 1$.

Шаг 4. Полагают: $j = 1; q = 1; i = 0$.

Шаг 5. Индекс i увеличивается на единицу: $i = i + 1$.

Шаг 6. Если выполняется условие $i \leq n$, то осуществляется переход к шагу 7, в противном случае к шагу 15.

Шаг 7. Создание промежуточной матрицы μ_{ii}^1 : $\mu_{ii}^1 = \mu_{ii}; (i = \overline{1, 4})$.

Шаг 8. Если выполняется условия $(\mu_{ii}^1 = 0)u(\mu_{2i}^1 = 0)u(\mu_{3i}^1 = 0)u(\mu_{4i}^1 = 0)$, то осуществляется переход к шагу 5, в противном случае к шагу 9.

Шаг 9. Вычисляют: $\mu_{ii}^1 = \mu_{ii} + z_{ii}; (i = \overline{1, 4})$.

Шаг 10. Если выполняется условие $i = d_{kj}^-$, то осуществляется переход к шагу 11, в противном случае к шагу 5.

Шаг 11. Если $j = 1$, то полагают:

$$\{\max_a = \mu_{ii}^1; \max_b = \mu_{2i}^1; \max_\alpha = \mu_{3i}^1; \max_\beta = \mu_{4i}^1\}$$

и осуществляется переход к шагу 12, в противном случае к шагу 13.

Шаг 12. Обнулирование элементов матрицы μ_{ii}^1 : $\mu_{ii}^1 = 0 (i = \overline{1, 4})$; индекс j увеличивается на единицу: $j = j + 1$ и осуществляется переход к шагу 5.

Шаг 13. Полагают: $a_1 = \max_a; a_2 = \mu_{ii}^1$.

Шаг 14. Если выполняется условие $a_1 \geq a_2$, то $\max'_a = a_1$, в противном случае $\max'_a = a_2$;

14.1 Полагают: $b_1 = \max_b; b_2 = \mu_{2i}^1$;

14.2 Если выполняется условие $b_1 \geq b_2$, то $\max'_b = b_1$, в противном случае $\max'_b = b_2$;

14.3 Полагают: $\alpha_1 = \max_\alpha; \alpha_2 = \mu_{3i}^1; \beta_1 = \max_\beta; \beta_2 = \mu_{4i}^1$; вычисляют: $\alpha'_1 = a_1 - \alpha_1; \alpha'_2 = a_2 - \alpha_2$;

14.4 Если выполняется условие $\alpha'_1 \geq \alpha'_2$, то $\max'_\alpha = \alpha'_1$, в противном случае $\max'_\alpha = \alpha'_2$;

14.5 Вычисляют: $\max_\alpha = \max'_\alpha - \max'_\alpha$; $\beta'_1 = b_1 + \beta_1; \beta'_2 = b_2 + \beta_2$;

14.6 Если выполняется условие $\beta'_1 \geq \beta'_2$, то $\max'_\beta = \beta'_1$, в противном случае $\max'_\beta = \beta'_2$;

14.7 Вычисляют: $\max_\beta = \max'_\beta - \max'_\beta$; осуществляется переход к шагу 12.

Шаг 15. Полагают : $i=1$.

Шаг 16. Если выполняется условие $i \leq n$, то осуществляется переход к шагу 17, в противном случае переход к шагу 25.

Шаг 17. Если выполняется условие $i = d_{kq}^+$, то осуществляется переход к шагу 18, в противном случае индекс i увеличивается на единицу: $i=i+1$ и осуществляется переход к шагу 16.

Шаг 18. Создание промежуточной матрицы μ_{li}^2 : вычисляют:

$$\left\{ \begin{array}{l} \mu_{li}^2 = \max_a + s_{1k}; \mu_{2i}^2 = \max_b + s_{2k}; \\ \mu_{3i}^2 = \max_\alpha + s_{3k}; \mu_{4i}^2 = \max_\beta + s_{4k} \end{array} \right\}.$$

Шаг 19. Если выполняется условия $(\mu_{li}^1 = 0)u(\mu_{2i}^1 = 0)u(\mu_{3i}^1 = 0)u(\mu_{4i}^1 = 0)$, то осуществляется переход к шагу 20, в противном случае переход к шагу 21.

Шаг 20. Полагают: $\mu_{li}^1 = \mu_{li}^2; (l = \overline{1,4})$; индекс i увеличивается на единицу: $i=i+1$ и осуществляется переход к шагу 16.

Шаг 21. Если выполняется условие $q=1$, то полагают:

$\{ \min_a = \mu_{1i}^2; \min_b = \mu_{2i}^2; \min_\alpha = \mu_{3i}^2; \max_\beta = \mu_{4i}^2 \}$ и осуществляется переход к шагу 22, в противном случае переход к шагу 23.

Шаг 22. Полагают $\{ \mu_{li}^1 = \min_a; \mu_{2i}^1 = \min_b; \mu_{3i}^1 = \min_\alpha; \mu_{4i}^1 = \min_\beta \}$; значение q увеличивается на единицу: $q=q+1$ и осуществляется переход к шагу 17.

Шаг 23. Полагают: $a_1 = \min_a; a_2 = \mu_{1i}^2$.

Шаг 24. Если выполняется условие $a_1 \leq a_2$, то $\min'_a = a_1$, в противном случае $\min'_a = a_2$;

24.1 Полагают: $b_1 = \min_b; b_2 = \mu_{2i}^2$;

24.2 Если выполняется условие $b_1 \leq b_2$, то $\min'_b = b_1$, в противном случае $\min'_b = b_2$;

24.3 Полагают: $\alpha_1 = \min_\alpha; \alpha_2 = \mu_{3i}^2$; $\beta_1 = \min_\beta; \beta_2 = \mu_{4i}^2$; вычисляют: $\alpha'_1 = a_1 - \alpha_1$;

24.4 Если выполняется условие $\alpha'_1 \leq \alpha'_2$, то $\min'_\alpha = \alpha'_1$, в противном случае $\min'_\alpha = \alpha'_2$;

24.5 Вычисляют: $\min_\alpha = \min'_\alpha - \min'_\alpha$; $\beta'_1 = b_1 + \beta_1$; $\beta'_2 = b_2 + \beta_2$;

24.6 Если выполняется условие $\beta'_1 \leq \beta'_2$, то $\min'_\beta = \beta'_1$, в противном случае $\min'_\beta = \beta'_2$;

24.7 Вычисляют: $\min_\beta = \min'_\beta - \min'_\beta$ и осуществляется переход к шагу 22.

Шаг 25. Новая маркировка принимается за текущую: $\mu_{lr} = \mu_{lr}^1; (l = \overline{1,4}; r = \overline{1,n})$.

Шаг 26. Значение k увеличивается на единицу: $k=k+1$. Если $k \leq m$, то осуществляется переход к шагу 4.

Конец алгоритма.

ВЫВОДЫ

Разработан алгоритм вычисления структурных элементов НВСП, который обеспечивает эффективную форму представления динамики состояния модели, пространство достижимых состояний и последовательность срабатывания переходов в виде совокупности векторов и матриц. На основе предложенного алгоритма разработано программное обеспечение в среде Borland Delphi 7.0.

ЛИТЕРАТУРА

- [1] В.В.Борисов, В.В.Круглов, А.С. Федулов. Нечеткие модели и сети.- Телеком, Москва, 2012, 284 с.
- [2] А.В. Леоненков. Нечеткое моделирование в среде MATLAB и fuzzy ТЕСН.- БХВ-Петербург, Санкт-Петербург, 2005, 717 с.