

ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И ПРИКЛАДНАЯ МАТЕМАТИКА

УДК 681.325.5

С.В. Челебаев, Ю.А. Челебаева

СТРУКТУРЫ НЕЙРОСЕТЕВЫХ ПРЕОБРАЗОВАТЕЛЕЙ ЧАСТОТНО-ВРЕМЕННЫХ ПАРАМЕТРОВ СИГНАЛОВ В ЦИФРОВОЙ КОД ДВУХ ПЕРЕМЕННЫХ НА ОСНОВЕ МОДИФИЦИРОВАННЫХ ПЕРСЕПТРОННЫХ СЕТЕЙ

Предложены структуры преобразователей частотно-временных параметров сигналов в цифровой унитарный и позиционный код двух переменных на основе модифицированных одно-, двух- и трехслойных перцептронных сетей. Структура нейросетевого преобразователя временного интервала в позиционный код двух переменных на основе модифицированного трехслойного перцептрона реализована на языке описания аппаратуры VHDL.

***Ключевые слова:** нейросетевой преобразователь, частота, временной интервал, код, перцептрон, язык описания аппаратуры VHDL.*

Введение. В настоящее время аппарат искусственных нейронных сетей (ИНС) является эффективным средством, на базе которого осуществляется синтез преобразователей формы представления информации [1, 2]. Эти устройства оперируют с переменными, заданными в виде частоты f , длительности временного интервала τ , периода T изменения сигнала, его амплитуды в виде напряжения U , число импульсного n или позиционного N кодов, и представляют собой аналого-цифровую нейросеть.

В известных публикациях [3 – 6] освещены вопросы синтеза нейросетевых линейных и нелинейных аналого-цифровых преобразователей, реализующих функциональные зависимости одной переменной. Однако не описаны вопросы синтеза нейросетевых преобразователей, реализующих функциональные зависимости двух переменных.

Необходимость воспроизведения функций двух и большего числа переменных обычно возникает при решении баллистических и навигационных задач, задач управления и контроля движущимися объектами, различными технологическими процессами, а также задач, связанных с исследованием и моделированием сложных систем авторегулирования.

Устройства для воспроизведения функций, функциональные преобразователи, самостоятельно или в составе различных управляющих, информационно-измерительных, контрольно-испытательных и других компонентов осуществляют вычисление требуемых зависимостей, линеаризацию характеристик, формирование корректирующих функций или моделирование нелинейностей.

В работе [7] предлагаются структуры нейросетевых преобразователей частотно-временных параметров сигналов в код двух переменных на основе перцептронных сетей. Однако эти структуры не учитывают мультипликативной зависимости между двумя переменными. Поэтому возникает необходимость модификации структур преобразователей на основе перцептронных сетей.

Цель работы – модификация структур преобразователей двух переменных на основе перцептронных сетей посредством добавления дополнительного слоя, вычисляющего взвешенное произведение входных переменных.

Теоретическая часть

В ходе исследований были разработаны структуры преобразователей частотно-временных параметров сигналов в цифровой унитарный и позиционный код на основе моди-

фицированных одно-, двух- и трехслойной персептронных сетей.

Структура преобразователя частотно-временных параметров сигналов в цифровой код двух переменных на основе применения модифицированного однослойного персептрона. Структура преобразователя $x_1, x_2 \rightarrow y^*$ двух переменных на основе модифицированного однослойного персептрона показана на рисунке 2.

Структура предназначена для решения конкретных задач преобразования аналоговых величин x_1 и x_2 , информативные параметры которых заданы в виде частот f_{x_1} и f_{x_2} или временных интервалов τ_{x_1} и τ_{x_2} , в цифровой эквивалент $y_n^* \equiv f(x_1, x_2)$ с унитарным способом кодирования эквивалента в диапазоне 000...0, 100...0, 110...0, ..., 111...1:

$$y_n^* = \alpha_1 \alpha_2 \dots \alpha_{k-1} = \sum_{i=1}^{k-1} \alpha_i, \quad (1)$$

где α_i – бит кода y_n^* ; $\alpha_i \in \{0, 1\}$; $i = \overline{1, k-1}$; k – количество участков аппроксимации; y_n^* – унитарный код.

В основу построения модели разрабатываемого преобразователя $x_1, x_2 \rightarrow y_n^*$ положен модифицированный однослойный персептрон. Его отличительной особенностью является добавление между входами и выходным слоем дополнительного слоя, вычисляющего взвешенное произведение входных переменных. Единственный нейрон 1-го слоя состоит из двух частей:

- множительного устройства;
- активационного устройства.

Аналитическое описание 1-го слоя предлагаемой нейронной сети (рисунок 1) имеет вид:

$$z_1 = F^{(1)}\left(\prod_{j=1}^3 w_{j,1}^{(1)} x_j\right),$$

где $F^{(1)}$ – линейная функция активации;

z_1 – выход нейрона 1-го слоя;

$w_{j,1}^{(1)}$ – коэффициент, задающий вес синаптической связи между j -м входом и нейроном 1-го слоя; $j = \overline{1, 3}$;

x_1 и x_2 – преобразуемые величины, а x_3 – эталонная (опорная) величина x_{on} .

Значение эталона x_{on} определяет меру преобразования (измерения). Эталон x_{on} обычно представляется в виде дискрета (кванта) преобразования $\Delta x_0 = x_{max}/k$ или максимального зна-

чения x_{max} . Для рассматриваемого далее варианта нейросетевой модели преобразователя принято: $x_{it} = \Delta x_0$.

При использовании данной сети на вход нейронов 2-го слоя поступают не только выход z_1 нейрона 1-го слоя, но и входные x_1, x_2, x_3 сигналы сети.

Аналитическое выражение для 2-го слоя сети (рисунок 1) имеет вид:

$$\alpha_i = F^{(2)}\left(\sum_{j=1}^3 w_{j,i}^{(2)} x_j + z_1\right), \quad i = \overline{1, k-1}, \quad (2)$$

где α_i – бит кода y_n^* (1);

$F^{(2)}$ – пороговая функция активации:

$$F^{(2)}(S_i^{(2)}) = \begin{cases} 1, & \text{если } S_i^{(2)} \geq \theta_i^{(2)}; \\ 0, & \text{иначе;} \end{cases} \quad (3)$$

где $S_i^{(2)} = \sum_{j=1}^3 w_{j,i}^{(2)} x_j + z_1$ – взвешенная сумма входных сигналов для i -го нейрона 2-го слоя; $w_{j,i}^{(2)}$ – коэффициент, задающий вес синаптической связи между j -м входом и нейроном 2-го слоя; $j = \overline{1, 3}$;

z_1 – выход нейрона 1-го слоя;

$\theta_i^{(2)}$ – пороговые значения нейронов 2-го слоя.

Результирующее выражение для 2-го слоя сети (рисунок 1) имеет вид:

$$\alpha_i = F^{(2)}\left(\sum_{j=1}^3 w_{j,i}^{(2)} x_j + F^{(1)}\left(\prod_{j=1}^3 w_{j,1}^{(1)} x_j\right)\right), \quad i = \overline{1, k-1}. \quad (4)$$

Исходно значение k , необходимое для обеспечения максимальной приведенной погрешности $\delta_{np} = \Delta y_{max}/y_{max}$ линейного преобразования $x \rightarrow y_n^*$, не превышающей заданного (допустимого) значения $\delta_{np_{oon}}$, выбирается из соотношения $k \geq \text{Int}(1/\delta_{np_{oon}})$, где Int – операция определения ближайшего большего целого. Поэтому фактически количество $\mu_2 = k-1$ нейронов в выходном слое модифицированного персептрона напрямую зависит от точности преобразования.

Для модифицированной однослойной структуры время преобразования T_{np} , затрачиваемое на формирование цифрового эквивалента y_n^* , находится как максимально возможное из времен T_{α_i} ($i = \overline{1, k-1}$) получения значения бита α_i на двух слоях:

$$T_{np} = \max_i \{T_{\alpha_i}\}.$$

При определении вида выходных сигналов нейронов $HU_1^{(0)}$, $HU_2^{(0)}$ и $HU_3^{(0)}$ нулевого (входного) слоя ИНС-преобразователя (рисунок 1) следует учитывать, что они должны соответственно иметь одинаковую с величинами x_1 , x_2 и x_{on} форму представления. Собственно, операцию аналого-цифрового преобразования осуществляют далее нейроны $HU_i^{(2)}$ ($i = \overline{1, k-1}$) его выходного слоя.

Значения функций активации, формируемые нейроузлами выходного слоя, могут быть представлены (закодированы) как в виде фиксированных значений бита, равных «0» или «1», так и в виде единичного импульса $l[t_v]$. При исполь-

зовании импульсной формы кодирования значений пороговой функции активации ее результат формируется в соответствии с выражением вида:

$$F^{(2)}(S_i^{(2)}[t_v]) = \begin{cases} 1[t_v], & \text{если } S_i^{(2)}[t_v] \geq \theta_i^{(2)}; \\ 0, & \text{иначе;} \end{cases} \quad (5)$$

где $S_i^{(2)}[t_v]$ – значение суммы на входе порогового элемента нейрона в дискретный момент времени (такт) $t_v = vT$, v – номер такта преобразования длительностью T ;

$\theta_i^{(2)}$ – пороговое значение i -го нейрона 2-го слоя; $i = \overline{1, k-1}$.

Обучение однослойного персептрона, как правило, осуществляется с помощью алгоритма Розенблатта [8, 9].

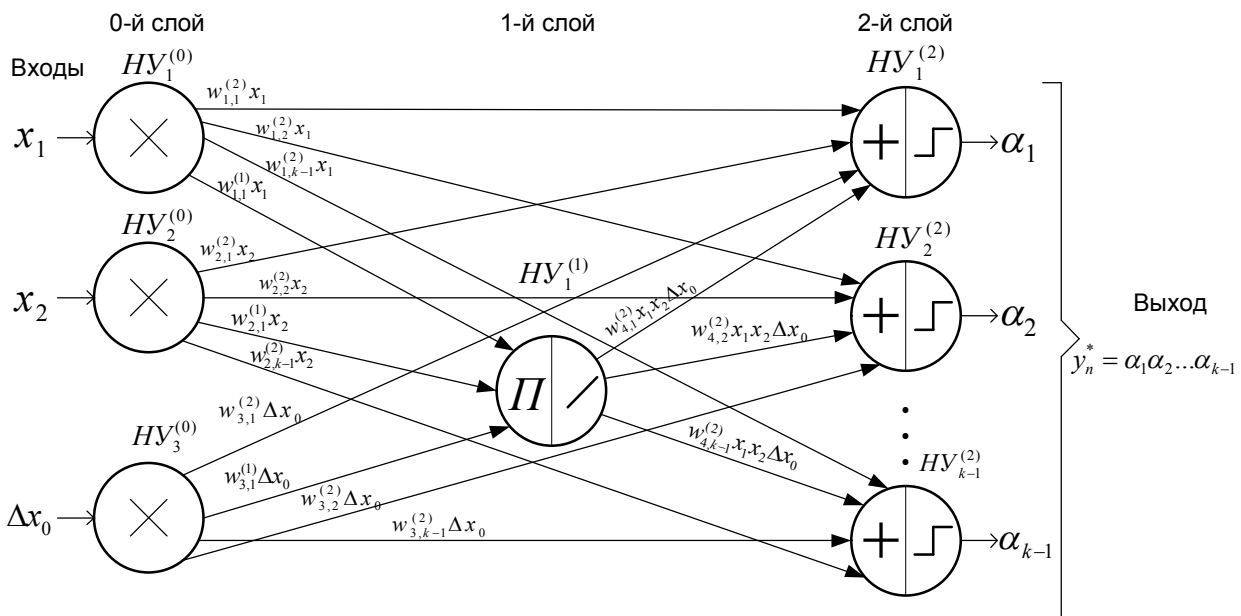


Рисунок 1 – Структура преобразователя $x_1, x_2 \rightarrow y_n^*$ на основе модифицированного однослойного персептрона

Структура преобразователя частотно-временных параметров сигналов в цифровой код двух переменных на основе применения модифицированного двухслойного персептрона. Структура преобразователя $x_1, x_2 \rightarrow y^*$ двух переменных на основе модифицированного двухслойного персептрона показана на рисунке 2. Она предназначена для получения результата y^* в виде унитарного кода в диапазоне 000...0, 100...0, 010...0, ..., 000...1 в форме $y_n^* = \psi_1\psi_2 \dots \psi_k$, где $\psi_i = \{0, 1\}$ – биты кода y_n^* .

Аналитическое описание для 1-го слоя сети (рисунок 2) имеет вид:

$$z_1 = F^{(1)}\left(\prod_{j=1}^3 w_{j,1}^{(1)} x_j\right), \quad (6)$$

где x_1 и x_2 – входные преобразуемые переменные; $x_3 = \Delta x_0 = x_{\max} / k$ – эталон преобразования; z_1 – выход нейрона 1-го слоя;

$w_{j,1}^{(1)}$ – коэффициенты, которые устанавливают веса синаптических связей между входами и нейроном 1-го слоя.

Аналитическое выражение для 2-го слоя сети (рисунок 2) имеет вид:

$$z_i^{(2)} = F^{(2)}\left(\sum_{j=1}^3 w_{j,i}^{(2)} x_j + z_1\right), \quad i = \overline{1, 2k}, \quad (7)$$

где $F^{(2)}$ – пороговая функция активации (3).

Результирующее выражение для 2-го слоя сети (рисунок 2), полученное на основе зависимостей (6) и (7), имеет вид:

$$z_i^{(2)} = F^{(2)}\left(\sum_{j=1}^3 w_{j,i}^{(2)}x_j + F^{(1)}\left(\prod_{j=1}^3 w_{j,1}^{(1)}x_j\right)\right), \quad i = \overline{1, 2k}. \quad (8)$$

Описание преобразователя $x_1, x_2 \rightarrow y_n^*$ функций двух переменных на основе модифицированного двухслойного персептрона может быть представлено как:

$$\psi_i = F^{(3)}\left(\sum_{r=1}^{2k} w_{r,i}^{(3)} F^{(2)}\left(\sum_{j=1}^3 w_{j,r}^{(2)} x_j + F^{(1)}\left(\prod_{j=1}^3 w_{j,1}^{(1)} x_j\right)\right)\right), \quad i = \overline{1, k}, \quad (9)$$

где $F^{(2)}$ и $F^{(3)}$ – пороговые функции активации

вида (3);

$F^{(1)}$ – линейная функция активации;

k – число разрядов кода y_n^* ;

$w_{j,r}^{(2)}$ – коэффициент, задающий вес связи между j -м входом и r -м нейроном 2-го слоя;

$w_{r,i}^{(3)}$ – коэффициент, задающий вес связи между r -м нейроном 2-го слоя и i -м нейроном 3-го слоя.

Обучение модифицированного двухслойного персептрона осуществляется с помощью алгоритма обратного распространения ошибки [8].

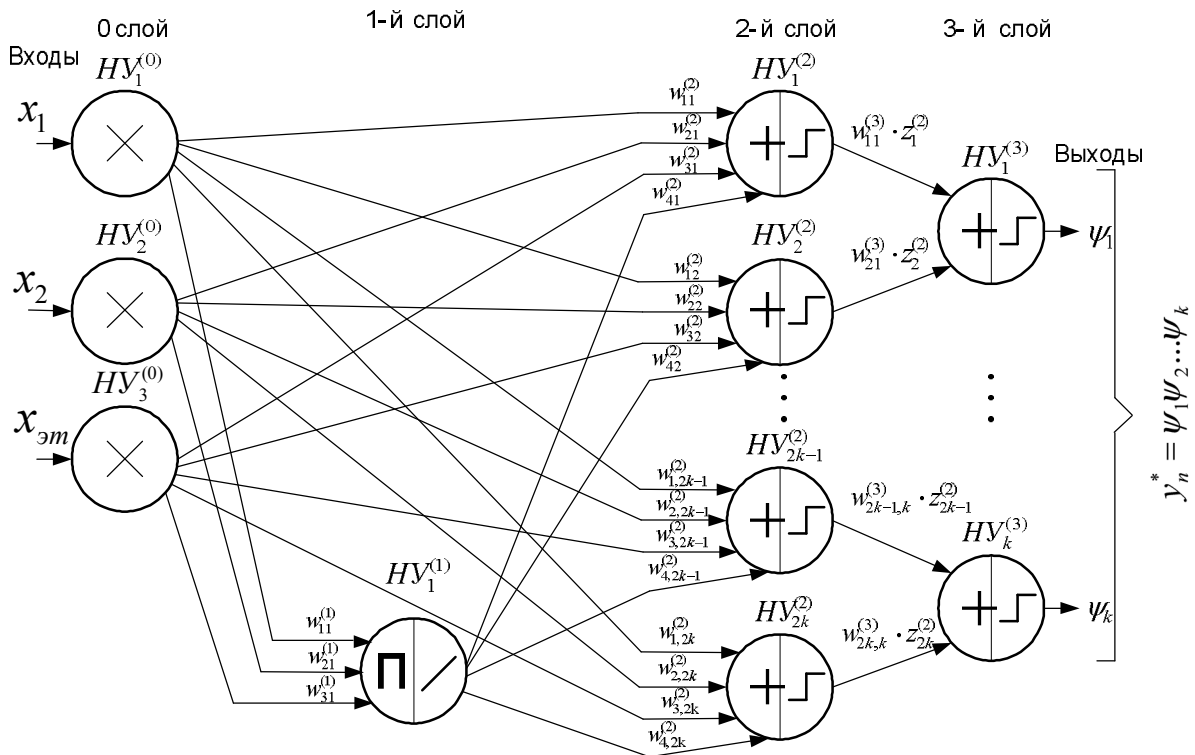


Рисунок 2 – Структура преобразователя $x_1, x_2 \rightarrow y^*$ на основе модифицированного двухслойного персептрона

Структура преобразователя частотно-временных параметров сигналов в цифровой код двух переменных на основе модифицированного трёхслойного персептрона. Структура преобразователя $x_1, x_2 \rightarrow y^*$ двух переменных на основе модифицированного трехслойного персептрона показана на рисунке 3. Структура предназначена для получения результата y^* в виде позиционного кода в диапазоне 000...0, 000...01, 000...10, ..., 111...11 в форме $y_N^* = \beta_m \beta_{m-1} \dots \beta_1$, где β_l – бит кода y_N^* ; $\beta_l \in \{0, 1\}$; $l = \overline{1, m}$.

Аналитическое описание для 1-го слоя сети (рисунок 3) имеет вид:

$$z_1 = F^{(1)}\left(\prod_{j=1}^3 w_{j,1}^{(1)} x_j\right), \quad (10)$$

где x_1 и x_2 – входные преобразуемые переменные; $x_3 = \Delta x_0 = x_{\max} / k$ – эталон преобразования; z_1 – вывод нейрона 1-го слоя;

$w_{j,1}^{(1)}$ – коэффициенты, которые устанавливают веса синаптических связей между входами и нейроном 1-го слоя.

Аналитическое выражение для 2-го слоя сети (рисунок 3) имеет вид:

$$z_i^{(2)} = F^{(2)}\left(\sum_{j=1}^3 w_{j,i}^{(2)} x_j + z_1\right), \quad i = \overline{1, \mu_2}, \quad (11)$$

где $F^{(2)}$ – пороговая функция активации (3);

μ_2 – количество нейронов 2-го слоя.

Результирующее выражение для 2-го слоя сети (рисунок 3) на основе выражений (10) и (11) имеет вид:

$$z_i^{(2)} = F^{(2)}\left(\sum_{j=1}^3 w_{j,i}^{(2)} x_j + F^{(1)}\left(\prod_{j=1}^3 w_{j,1}^{(1)} x_j\right)\right), i = \overline{1, \mu_2} \quad (12)$$

Аналитическое выражение для 3-го слоя сети (рисунок 3) имеет вид:

$$z_i^{(3)} = F^{(3)}\left(\sum_{r=1}^{\mu_2} w_{r,i}^{(3)} F^{(2)}\left(\sum_{j=1}^3 w_{j,r}^{(2)} x_j + F^{(1)}\left(\prod_{j=1}^3 w_{j,1}^{(1)} x_j\right)\right)\right), i = \overline{1, \mu_3},$$

где μ_3 – количество нейронов 3-го слоя.

Описание преобразователя $x_1, x_2 \rightarrow y_N^*$ функций двух переменных на основе модифицированного трехслойного персептрона может быть представлено как:

$$\beta_i = \left(\sum_{p=1}^m w_{p,i}^{(4)} F^{(3)}\left(\sum_{r=1}^{\mu_2} w_{r,p}^{(3)} F^{(2)}\left(\sum_{j=1}^3 w_{j,r}^{(2)} x_j + F^{(1)}\left(\prod_{j=1}^3 w_{j,1}^{(1)} x_j\right)\right)\right)\right), i = \overline{1, m}, \quad (13)$$

где $F^{(2)}$, $F^{(3)}$ и $F^{(4)}$ – пороговые функции активации вида (3); m – число разрядов кода y_N^* .

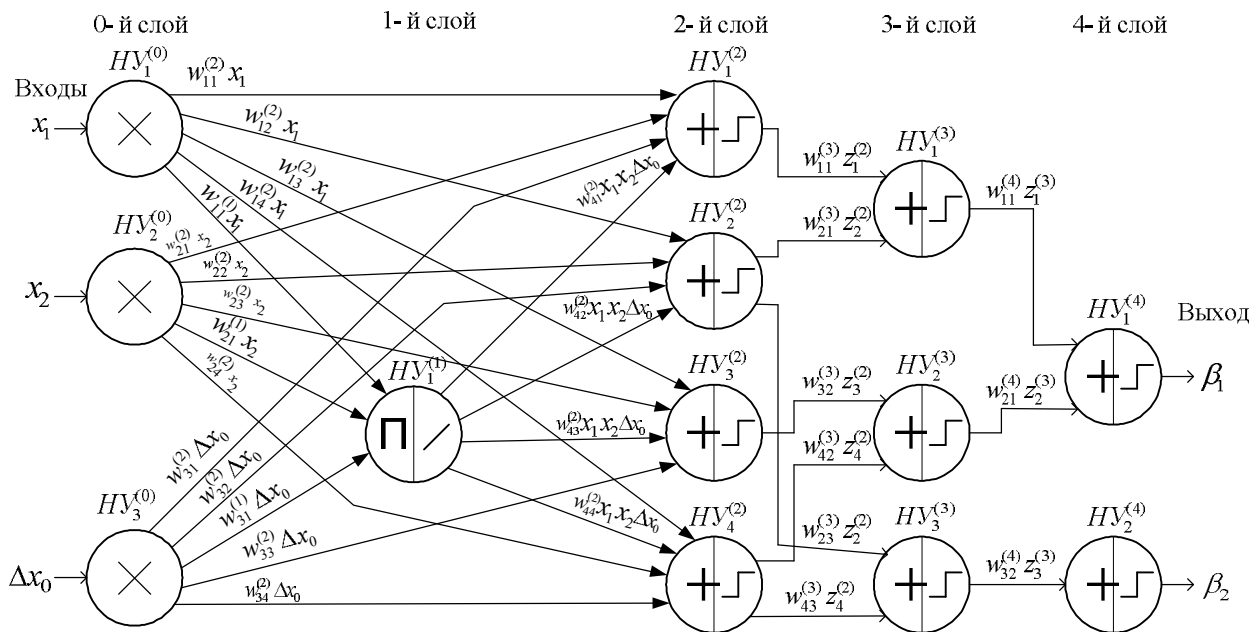


Рисунок 3 – Структура преобразователя $x_1, x_2 \rightarrow y^*$ на основе модифицированного трёхслойного персептрона (для $m=2$)

Экспериментальная часть

Структура нейросетевого преобразователя временного интервала в позиционный код двух переменных на основе модифицированного трехслойного персептрона реализована на языке описания аппаратуры VHDL. Для получения логических схем нейросетевых преобразователей на программируемых логических интегральных схемах (ПЛИС) может быть применен язык описания аппаратуры высокого уровня VHDL, который отличается простотой программ [9]. Компоненты, написанные на VHDL, могут быть использованы для ПЛИС разных производителей, таких как Xilinx, Altera и др.

Ниже приведен фрагмент программы, описывающий реализацию единственного нейрона 1-го слоя нейросетевого преобразователя временного интервала в позиционный код двух переменных на основе модифицированного трехслойного персептрона.

```
entity layer1 is
  Port (
    f0, tau1, tau2, start, finish: in std_logic;
    w1,w2 : in std_logic_vector(11 downto 0);
    z : out std_logic);
end layer1;
architecture Behavioral of layer1 is
  signal tau1_cnt, tau2_cnt, tau_rez_cnt :
  std_logic_vector(11 downto 0);
  signal tau1_t,tau2_t: std_logic;
  signal s1,s2 : std_logic_vector(11 downto 0);
  signal p1,p2 : std_logic_vector(23 downto 0);
  signal s3 : std_logic_vector(47 downto 0);
  signal z_in : std_logic;
begin
  process(f0)
  begin
    if (f0='1' and f0'event) then
      tau1_t <= tau1;
      tau2_t <= tau2;
```

```

end if;
end process;
process(f0)
begin
if (f0='1' and f0'event)
then if (start = '1')
then tau1_cnt <= "000000000000";
else if (tau1_t = '1')
then tau1_cnt <= tau1_cnt +
"000000000001";
end if;
end if;
end process;
process(f0)
begin
if (f0='1' and f0'event)
then if (start = '1')
then tau2_cnt <= "000000000000";
else if (tau2_t = '1')
then tau2_cnt <= tau2_cnt +
"000000000001";
end if;
end if;
end process;
process(f0)
begin
if (f0='1' and f0'event)
then s1 <= tau1_cnt;
s2 <= tau2_cnt;
end if;
end process;
p1 <= w1*s1;
p2 <= w2*s2;
s3 <= p1*p2;
process(f0)
begin
if (f0='1' and f0'event)
then if (start = '1')
then tau_rez_cnt <= "000000000000";
else if (tau_rez_cnt <= s3)
then tau_rez_cnt <= tau_rez_cnt +

```

```

"000000000001";
end if;
end process;
process(f0)
begin
if (f0='1' and f0'event)
then if (start = '1')
then z_in <= '0';
else if ((tau_rez_cnt > "000000000000") and
(tau_rez_cnt <= s3(11 downto 0)))
then z_in <= '1';
else z_in <= '0';
end if;
end if;
end process;
z <= z_in;
end Behavioral;

```

В приведенном фрагменте программы введены следующие обозначения: *tau1*, *tau2* – преобразуемые временные интервалы, *f0* – опорная частота преобразования, *start* и *finish* – управляющие сигналы начала и окончания преобразования, *w1*, *w2* – весовые коэффициенты, *theta* – порог нейрона, *z* – выходной сигнал нейрона, т.е. результирующий временной интервал.

Результаты моделирования показаны на рисунке 4. В качестве эталона задана опорная частота *f0* преобразования, равная 50 МГц (это соответствует периоду *T0*, равному 20 нс). Преобразуемые временные интервалы *tau1* и *tau2* заданы длительностью 40 нс и 60 нс соответственно, что соответствует величинам *2T0* и *3T0* соответственно. Весовые коэффициенты *w1* и *w2* равны единице, поэтому выходной временной интервал *z* нейрона должен быть равен величине *6T0*, что соответствует длительности *20*6=120 нс*. Эта длительность на рисунке 4 обозначена временными маркерами.

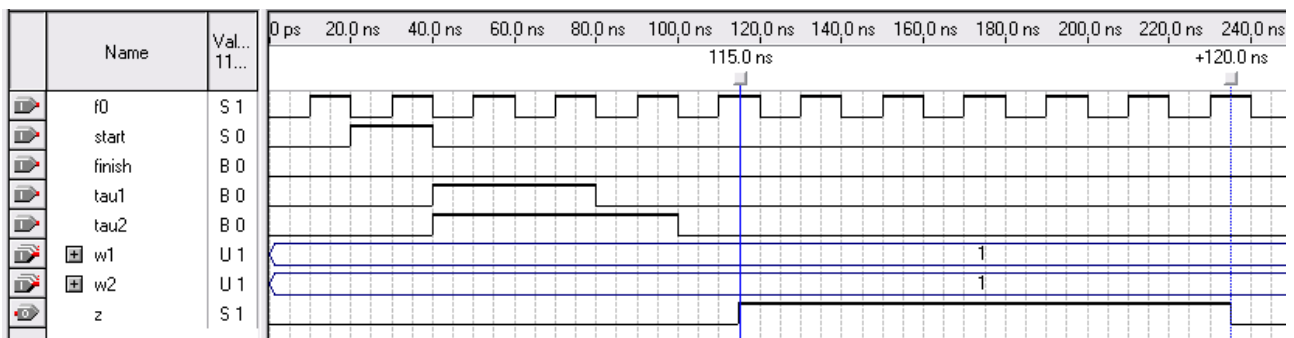


Рисунок 4 – Результаты моделирования нейрона 1-го слоя преобразователя $x_1, x_2 \rightarrow y^*$ на основе модифицированного трёхслойного персептрона

Из рисунка 4 видно, что временной интервал z сформирован верно, что подтверждает правильность программы.

Моделирование на языке C++ показало, что число нейронов скрытого слоя при реализации зависимости

$$y^* = \sin \frac{\pi}{2}(x_1 + 0,31) \cdot \cos \frac{\pi}{2}(x_2 + 0,31)$$

уменьшается на 10-20 % в зависимости от требуемой точности преобразования.

Заключение. Предложены структуры преобразователей частотно-временных параметров сигналов в цифровой унитарный и позиционный код двух переменных на основе модифицированных одно-, двух- и трехслойной персептронных сетей.

Предложенные структуры преобразователей частотно-временных параметров сигналов в цифровой код двух переменных на базе модифицированных персептронных сетей позволят понизить погрешность преобразования, сократить количество нейронов скрытых слоев за счет введения нейрона, реализующего мультипликативную зависимость между преобразуемыми переменными, а также сократить сроки проектирования подобных устройств.

Структура нейросетевого преобразователя временного интервала в позиционный код двух переменных на основе модифицированного трехслойного персептрона реализована на языке описания аппаратуры VHDL, осуществлено ее функциональное моделирование.

Работа выполнена при финансовой поддержке Министерства образования и науки РФ. Государственное задание № 2014/178.

Библиографический список

1. Галушкин А.И. Нейронные сети: основы теории. М.: Горячая линия–Телеком, 2010. – 496 с.

2. Локтюхин В.Н., Челебаев С.В. Нейросетевые преобразователи импульсно-аналоговой информации: организация, синтез, реализация / под общей редакцией А.И. Галушкина. – М.: Горячая линия – Телеком, 2008. 144 с.

3. David W. Tank, John J. Hopfield. Simple "neural" optimization networks: an A/D converter, signal decision circuit and a linear programming circuit // IEEE Circuit and Systems, vol. CAS-33, May 1986, p. 533-541.

4. Bang W. Lee, Bing J. Shen. Design of a neural-based A/D converter using modified Hopfield Network // IEEE Solid-State Circuits, vol. SC-24, Aug. 1989, p. 1120-1135.

5. Avitabile G., Forti M., Manetti S., Marini M. On a class of nonsymmetrical neural networks with application to ADC // IEEE Circuit and Systems, vol. CAS-38, Feb 1991, p. 202-209.

6. Локтюхин В.Н., Челебаев С.В., Антоненко А.В. Нейросетевые аналого-цифровые преобразователи // под общей редакцией А.И. Галушкина. – М.: Горячая линия–Телеком, 2010. – 128 с.

7. Chelebaev S.V., Chelebaeva Y.A. Two variables time-and-frequency signal parameters in a digital code neural network converters synthesis procedure on the perceptrons basis // 3rd Mediterranean conference on embedded computing (MECO). Budva, Montenegro. 2014. p. 140-143.

8. Ясницкий Л.Н. Введение в искусственный интеллект: учеб. пособие для студ. высш. учеб. заведений. – М.: Издательский центр "Академия", 2005. – 176 с.

9. Локтюхин В.Н., Челебаев С.В., Антоненко А.В. Алгоритмы настройки персептронных сетей на преобразование частотно-временных параметров сигналов в код // Вестник Рязанского государственного радиотехнического университета. 2009. № 30. С. 27 – 35.

УДК 004.822+004.89-027.31

Н.И. Цуканова

ОНТОЛОГИЧЕСКИЕ МОДЕЛИ ДОКУМЕНТА

Предложены три онтологические модели документа: «Последовательность», «Целое – Часть», «Вход – Выход», проведено их исследование с помощью редактора Protégé4 и сравнение. Показана возможность их применения для автоматического анализа документов.

Ключевые слова: онтологическая модель, дескрипционная логика, документ.

Актуальность. Если нам надо понять, о чем книга (или документ), мы обращаемся к содержанию. Содержание позволяет быстро охватить основные разделы, порядок их следования, в какой-то мере понять идею автора или к какой те-

матике принадлежит документ. Таким образом, содержание передает смысл документа. В настоящее время при автоматизации поиска информации задача понимания смысла стоит не только перед людьми, но и перед программами – аген-

тами, собирающими информацию. Для ее решения можно было бы использовать содержание, но оно должно быть представлено в формате, понятном для программ-агентов. В качестве такого формата можно использовать онтологическую модель [1 – 3], поэтому разработка различных онтологических моделей документа является в наше время актуальной задачей.

Цель работы – разработка онтологических моделей документа, передающих его смысл. В онтологической модели документ описывается формулами дескрипционной логики, записанными с помощью языка разметки *OWL*-языка *Semantic Web*, что позволяет автоматизировать процесс анализа документа с помощью процедуры вывода.

Введение. Основными элементами, передающими смысл документа, являются заголовок, автор, содержание, предметный указатель, список литературы. Содержание дает информацию об основных концептах (сущностях) описываемой в документе предметной области, о разделении документа на логически законченные и связанные фрагменты текста, об отношениях «Целое – Часть» между ними и о логическом следовании разделов друг за другом. Логическое следование показывает выводимость новой информации (нового раздела) на основе уже полученных в предыдущих разделах сведений и несет информацию о причинно-следственных отношениях, а также о развитии процесса во времени. В работе исследуются различные онтологические модели документа, которые могут быть построены на основе содержания, описываются связи, отражаемые этими моделями, и рассматривается вопрос, что может дать такая модель при выводе. Построение моделей и вывод осуществляются в редакторе *Protege 4*. В качестве примера документа взята глава «Описание предметной области с помощью программы на Прологе» из учебного пособия Цукановой Н.И., Дмитриевой Т.А. «Теория и практика логического программирования на языке *Visual Prolog 7*» [4].

Онтологическая модель «Последовательность» передает совокупность терминов и их следование в порядке чтения документа в виде иерархии вложенных классов. Если прочитан первый фрагмент, то освоены его концепты и можно перейти к следующему разделу, следующий раздел формирует знание новых концептов, и если он прочитан, то можно перейти к следующему и т.д. Эта связь описывается следующей формулой дескрипционной логики (ДЛ):

$$F_1 \sqsubseteq F_2 \sqsubseteq F_3 \dots \sqsubseteq F_n, \quad (1)$$

где F_1, F_2, F_3, F_n – фрагменты текста, заданные

своим наименованием. Здесь и далее знаки $\rightarrow, \equiv, \forall, \exists, \sqsubseteq, \sqsupset$ являются соответственно знаками следующих логических и множественных операций: импликация, тождество, квантор всеобщности, квантор существования, включение, пересечение. Например:

Конкретные_объекты \sqsubseteq *Абстрактные_объекты*

Абстрактные_объекты \sqsubseteq *Объекты*

Объекты \sqsubseteq *Отношения_связи*

Отношения_связи \sqsubseteq *Предметная_область*

На рисунке 1 представлена онтология, передающая логическое следование фрагментов документа. По виду она напоминает лестницу познания, где каждая новая ступень – это новая единица знания, приобретаемая на основе предыдущих ступенек.

В онтологической модели информация о предметной области описывается логическими формулами дескрипционной логики (ДЛ). Благодаря этому для получения новых знаний может быть использован один из алгоритмов вывода, известный в дескрипционной логике и положенный в основу работы машины вывода (резонера). Как известно [3], машина вывода применяется для получения иерархии классов на основе их свойств, а также при ответе на простые запросы.

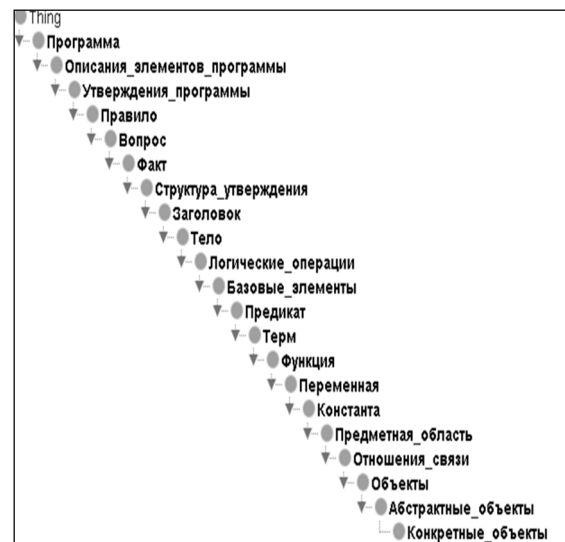


Рисунок 1 – Онтология документа «Описание предметной области на языке Пролог»

В модели «Последовательность» иерархия определяется изначально содержанием документа и описывается формулами (1). Выводимая иерархия подобна примитивной (исходной), т.к. не описаны свойства классов, связи между ними, и никакой новой информации не приносит. Однако машина вывода может быть использована при ответе на ряд простых, но полезных запросов ДЛ. Для заданного концепта могут быть по-

лучены 1) *Subclasses (Подкласс)* – ближайший концепт-предшественник; 2) *Descendant classes (классы-предки)* – предшествующие концепты, знание которых необходимо для понимания заданного концепта; 3) *Superclasses (Суперкласс)* – ближайший потомок, последователь – концепт, являющийся обобщением заданного концепта; 4) *Ancestor classes (классы-потомки)* – концепты, определение которых зависит от знания заданного концепта. Например, результатом работы машины вывода при ответе на вопрос *Descendant classes (Предшественники)* для концепта *Предикат* будет следующее множество концептов:

$D = \{\text{Конкретные_объекты, Абстрактные_объекты, Объекты, Отношения_связи, Предметная_область, Константа, Переменная, Функция, Терм}\}$

Эта автоматически получаемая информация несет дополнительные сведения о предметной области и может использоваться при информационном поиске, при индексировании документов, при дистанционном обучении.

Онтологическая модель «Целое – Часть».

Иерархия классов может быть построена по тексту содержания (рисунок 2), при этом хорошо передаются отношения «Целое – Часть», но теряется информация о порядке следования разделов внутри класса.

В этом случае описание содержания на языке дескрипционной логики выглядит следующим образом:

$$\Pi_{11} \sqsubseteq \Gamma_1; \Pi_{12} \sqsubseteq \Gamma_1; \dots \quad (2)$$

$$\Pi_{12} \sqsubseteq \Gamma_2; \Pi_{22} \sqsubseteq \Gamma_2; \dots \text{ и т.д.,}$$

где Π_{ij} – это параграфы, Γ_i – главы, все элементы объявлены непересекающимися:

$$\Gamma_1 \cap \Gamma_2 \cap \Gamma_3 \dots \cap \Gamma_n = \emptyset, \Pi_{11} \cap \Pi_{12} = \emptyset \text{ и т.д.} \quad (3)$$

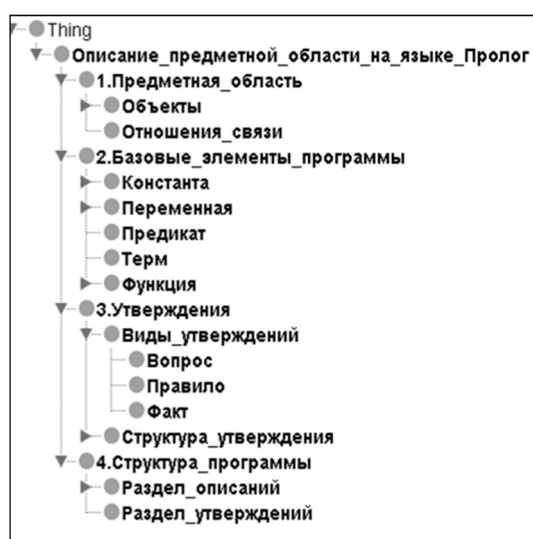


Рисунок 2 – Иерархия классов, построенная по тексту содержания

С помощью этих формул можно передать только отношение «Целое – часть». Из них следует, что глава 1 состоит из параграфов Π_{12} и Π_{11} , глава 2 – из Π_{21} и Π_{22} . Но из этих формул нельзя получить информацию о порядке следования глав первой и второй, параграфов 12 и 22. Для логических формул (2) не имеет смысла порядок их следования. Таким образом, порядок следования текста в документе нельзя передать с помощью модели «Целое – Часть», хотя эта информация очень важна – она определяет логику всего документа.

Рассмотрим, какую информацию можно получить путем вывода в этой модели. В качестве примера приведем ответы на запросы для концепта «Переменная».

Для концепта «Переменная» ответом на вопрос: «Какие элементы являются частью концепта?» (*Subclasses*) будет множество:

$S = \{\text{Значение_переменной, Имя_переменной}\}$;

а ответом на вопрос: «Какие понятия обобщают концепт (или частью каких понятий является заданный концепт)?» (*Ancestor classes*) получим элементы:

$A = \{2.\text{Базовые_элементы_программы, Описание_предметной_области_на_языке_Пролог}\}$

Можно попытаться объединить эти модели. Для этого нужно удалить условие непересекаемости классов (3) и задать порядок следования глав и параграфов следующими формулами:

$$\Gamma_1 \sqsubseteq \Gamma_2 \sqsubseteq \Gamma_3 \sqsubseteq \dots \sqsubseteq \Gamma_n; \Pi_{11} \sqsubseteq \Pi_{12}; \Pi_{21} \sqsubseteq \Pi_{22}, \dots \quad (4)$$

В этом случае автоматическая классификация выполняется без противоречий и приводит к интересным результатам. Так, например, на рисунке 3 приведена полученная с помощью резонера иерархия классов для смешанной модели (4).

Как видно из рисунка, в ней отражен порядок следования материала (1) – (4) снизу вверх, внутри каждого раздела определены его части.

Онтологическая модель «Логическая выводимость концептов» или «Вход – Выход». Документ разбивается на поименованные фрагменты (главы, параграфы, пункты, подпункты). Каждый фрагмент рассматривается как процедура вывода (получения) новой информации на основе известной. Поэтому у каждого фрагмента есть *входные* концепты, которые уже известны читателю и необходимы для формирования новых, так называемых *выходных* концептов, приобретаемых в процессе чтения фрагмента.

В результате модель содержит два класса сущностей: *Концепты (К)* и *Фрагменты (Фр)*, которые связаны между собой бинарными связями $R = \{r1, r2, \dots, r8\}$:

$r1$ – «концепт_вход_для_фрагмента»,
 $r2$ – «концепт_сформирован_фрагментом»,
 или инверсными им:
 $r3=r1^{-1}$ – «фрагмент_требует_знания_концепта»,
 $r4=r2^{-1}$ – «фрагмент_формирует_знание_концепта». На основе этих связей можно получить следующие производные транзитивные отношения:
 $r5$ – фрагмент_предшествует_фрагменту:
 $r5 \equiv r4 \circ r1$;
 $r6$ – фрагмент_следует_за_фрагментом:
 $r6 \equiv inverse(r5)$;
 $r7$ – концепт_предшественник_концепта:
 $r7 \equiv r1 \circ r4$;
 $r8$ – концепт_последователь_концепта:
 $r8 \equiv inverse(r7)$,
 где \circ – операция композиции отношений, $inverse()$ – инверсное отношение.

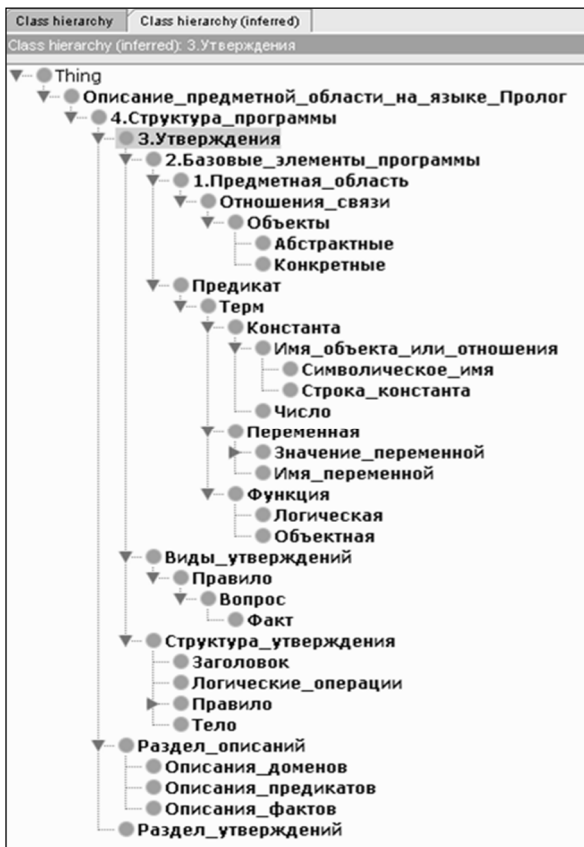


Рисунок 3 – Иерархия классов, полученная путем вывода для смешанной модели

Все эти связи определяются на вкладке *Object Property* редактора *Protege 4*. Эта модель достаточно сложная, но позволяет определить правильность логического следования материала.

Базовые классы *Фрагменты (Фр)* и *Концепты (К)* описываются следующими логическими формулами:

$K \rightarrow \forall r1. Фр \text{ П } \forall r2. Фр$;
 $Фр \rightarrow \exists r3. К \text{ П } \forall r3. К \text{ П } \exists r4. К \text{ П } \forall r4. К$.

Здесь $\forall r1. Фр$ – ограничение отношения квантором всеобщности, описывает множество сущностей, которые если имеют связь $r1$, то только с элементами множества $Фр$; $\exists r3. К$ – ограничение отношения квантором существования, описывает множество сущностей, у которых есть хотя бы одна связь $r3$ с элементами множества $К$; П – пересечение множеств.

Далее базовые классы заполняются индивидуальностями. В классе *Фрагменты (Фр)* каждая индивидуальность задается именем фрагмента (главы, параграфа, пункта, абзаца), в классе *Концепты (К)* – словами, соответствующими концепту. Например, для предметной области «Описание предметной области с помощью программы на Прологе» фрагменты задаются следующими именами:

Фр_Предметная_область,
Фр_Базовые_элементы_программы,
Фр_Утверждения, *Фр_Правило* и т.д., концепты именами: *Константа*, *Переменная*, *Предикат*, *Терм*, *Правило* и т.п.

Каждый фрагмент-индивидуальность описывается с помощью логических формул, в которых указываются входные концепты (они необходимы для понимания фрагмента) и выходные концепты (формируемые фрагментом). Например, для фрагмента *Фр_Предикат* формулы будут иметь вид:

Фр_Предикат $r3$ *Отношение*;
Фр_Предикат $r3$ *Предметная_область*;
Фр_Предикат $r3$ *Терм*;
Фр_Предикат $r3$ *Логическая_функция*;
Фр_Предикат $r4$ *Предикат*.

Уже на этом этапе с помощью машины вывода можно для каждой индивидуальности получать новую информацию. Так, свойства индивидуальностей-концептов не задаются, а получаются путем вывода. В качестве примера приведем сначала информацию, полученную путем вывода для концепта «*Правило*»: *Правило* – это «концепт_сформирован_фрагментом_Фр_Правило», «концепт_на_основе_которого_получен_концепт_Описание_доменов», «концепт_на_основе_которого_получен_концепт_Описание_фактов», «концепт_на_основе_которого_получен_концепт_Описание_предикатов», «концепт_на_основе_которого_получен_концепт_Раздел_утверждений», «концепт_вход_для_фрагмента_Фр_Структура_программы»;

затем для фрагмента *Фр_Правило*: «фрагмент_предшествует_фрагменту»

Фр_Структура_программы»,
 «фрагмент следует за фрагментом
 Фр_Базовые_элементы_программы»,
 «фрагмент следует за фрагментом
 Фр_Утверждения»,
 «фрагмент следует за фрагментом
 Фр_Предметная_область».

Из нее можно узнать, каким фрагментом сформирован концепт «Правило» и для какого фрагмента он является входным, какой фрагмент предшествует фрагменту «Фр_правило» и какие фрагменты следуют за ним. Вся эта информация характеризует правильность логического следования материала, что особенно полезно для документов, связанных с процессом обучения.

Следующим шагом является создание определяемых классов, которые позволяют индивидуальности из классов *Концепты* и *Фрагменты* разбить на группы, удовлетворяющие определенным условиям. Рассмотрим несколько примеров таких групп. Например, классы «*Концепты_входные*», «*Концепты_формируемые*», «*Концепты_промежуточные*» можно описать с помощью необходимых и достаточных условий (*Equivalent classes*) следующим образом:

Концепты_входные $\equiv \exists r1. \text{Фр}$;

Концепты_формируемые $\equiv \exists r2. \text{Фр}$;

Концеп-

ты_промежуточные $\equiv \text{Концепты_входные } \Pi$

Концепты_формируемые

В результате в первую группу войдут все понятия, используемые в документе для формирования новых концептов, во вторую – вновь сформированные документом понятия, а в третью – их пересечение, т.е. те концепты, которые в документе формируются и вновь используются для получения новых понятий.

Используя логические формулы, можно создать следующие определяемые классы: «*Концепты, на которых основан заданный концепт*», «*Концепты, имеющие связь с множеством заданных концептов*», «*Фрагменты, связанные с заданным фрагментом*». Для заданного концепта можно получить классы «*Концепты – предшественники*» и «*Концепты – последователи*», аналогично для заданного фрагмента – «*Фрагменты предшественники*» и «*Фрагменты – последователи*». Наполнение этих классов элементами выполняется автоматически при запуске резонера.

На рисунке 4 приведена выводимая иерархия созданных классов. Их наполнение рассмотрим на примерах. Для класса: «*Фрагменты_предшественники_фрагмента_Фр_Предикат*» элементами будут: *Фр_Объекты*, *Фр_Отношения*, *Фр_Предметная_область*,

Фр_Базовые_элементы_программы, *Фр_Константа*, *Фр_Переменная*, *Фр_Терм*; а для класса «*Фрагменты_последователи_фрагмента_Фр_Предикат*»: *Фр_Факт*, *Фр_Вопрос*, *Фр_Правило*, *Фр_Структура_утверждения*, *Фр_Структура_программы*.

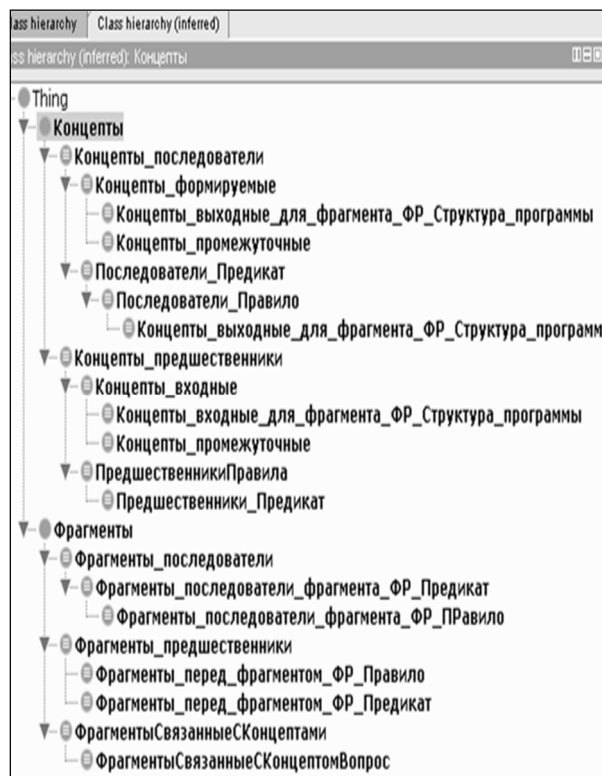


Рисунок 4 – Выводимая иерархия для модели «Вход - Выход»

Результаты. Предложенные в работе онтологические модели были разработаны для главы «Описание предметной области с помощью программы на Прологе» из учебного пособия Цукановой Н.И., Дмитриевой Т.А. «Теория и практика логического программирования на языке *Visual Prolog 7*» [4] в редакторе Protégé 4. Полученные путем вывода данные позволили найти несоответствия в логике изложения материала и впоследствии исправить найденные ошибки. Все онтологические модели сохраняются в формате языка *OWL* и могут быть использованы программами-агентами при поиске или индексировании документов.

Выводы. В работе предложены три онтологические модели документа. Первые две основаны на содержании документа, третья – на связи концептов и фрагментов, предназначенных для их формирования. Имена концептов могут быть взяты из текста или из предметного указателя, а имена фрагментов – из содержания.

Модель «Последовательность» передает логическое следование фрагментов документа друг

за другом, показывая процесс последовательного накопления новых знаний. Модель «Целое – Часть» описывает вложенность фрагментов друг в друга. В работе предлагается синтезировать эти две модели. Наиболее сложной и в то же время наиболее информативной является модель «Вход – Выход». Ее использование позволяет искать ошибки в изложении материала, находить логическую связь между концептами и фрагментами документа. Онтологические модели позволяют проводить анализ документов автоматически с помощью машины вывода, что способствует их использованию при поиске и индексировании документов, при передаче смысловой информации о документе по глобальной сети Интернет.

Библиографический список

1. Коровин С.Е. Семантико-прагматическая модель документа в нотации XML. Электронный журнал "ИССЛЕДОВАНО В РОССИИ" <http://1360://zhurnal.ape.relarn.ru/articles/2002/123.pdf>
2. Загорулько Ю.А., Загорулько Г.Б. О формализации семантики областей знаний в информационных и интеллектуальных системах на основе онтологий // Материалы IV международной научно-технической конференции OSTIS-2014. С.117-118.
3. Цуканова Н.И., Страхова З.В. Онтология учебно-методического комплекса // Вестник Рязанского государственного радиотехнического университета. 2013. № 1(43). С. 85 – 89.
4. Цуканова Н.И., Дмитриева Т.А. Теория и практика логического программирования на языке Visual Prolog 7. Учеб. пособие для вузов. М.:Горячая линия – Телеком, 2011. 232 с.: ил.

УДК 004.738.5, 81'322

А.В. Пруцков, Д.М. Цыбулько

ТЕОРЕТИКО-МНОЖЕСТВЕННОЕ ПРЕДСТАВЛЕНИЕ МЕТОДА ОБРАБОТКИ КОЛИЧЕСТВЕННЫХ ЧИСЛИТЕЛЬНЫХ ЕСТЕСТВЕННЫХ ЯЗЫКОВ И ОСОБЕННОСТИ ИХ ПЕРЕВОДА В РАЗЛИЧНЫХ СТРАНАХ

Предложено теоретико-множественное представление задачи обработки количественных числительных естественных языков. Решение задачи обработки числительных реализовано в разработанном авторами Интернет-приложении. Представлены и проанализированы направления перевода пользователей Интернет-приложения из Российской Федерации, Украины, Европейского Союза и Соединенных Штатов Америки. Анализ выполнен на основе статистики запросов к Интернет-приложению, включающей более 25 000 записей.

Ключевые слова: автоматическая обработка текста, машинный перевод, обработка числительных, сеть Интернет, лингвистические ресурсы, особенности перевода в различных странах.

Введение. В марте 2012 года начало свою работу разработанное авторами Интернет-приложение для обработки количественных числительных естественных языков (далее Интернет-приложение) [1], доступное по адресу <http://prutzkow.com/numbers/> в сети Интернет.

Интернет-приложение обладает широкой функциональностью и выполняет следующие операции:

1) перевод количественных числительных русского, английского, немецкого, испанского и финского языков в любом направлении для перечисленных языков;

2) преобразование количественных числительных в символьную (числительное) и цифро-

вую (число) формы (в терминах работы [2]);

3) генерация и определение форм слов числительных русского языка без учета контекста (в терминах работы [3]);

4) автоматическое распознавание языка введенного числительного;

5) проверка знаний правил образования и перевода числительных с динамической генерацией заданий [4].

Интернет-приложение имеет русско- и англоязычный интерфейс и доступно всем пользователям сети Интернет.

В основе Интернет-приложения лежит метод обработки количественных числительных естественных языков [2]. Метод предполагает, что

все преобразования числительных производятся через промежуточный этап, описанный трехуровневой моделью числительного [2].

Определения. Посетителем Интернет-приложения будем называть пользователя сети Интернет, который зашел на его страницу по адресу, указанному выше.

Пользователем Интернет-приложения будем называть посетителя Интернет-приложения, который использовал его для обработки числительных.

Целью статьи являются теоретико-множественное представление задачи обработки количественных числительных естественных языков, а также представление и анализ направлений перевода числительных пользователями Интернет-приложения из Российской Федерации, Украины, Европейского Союза (ЕС) и Соединенных Штатов Америки (США).

Теоретико-множественное представление задачи обработки количественных числительных естественных языков. Задача обработки числительных заключается в преобразовании числительных, записанных по правилам одного языка, в числительные, записанные по правилам другого языка.

Предлагается следующее теоретико-множественное представление задачи обработки числительных для его последующего практического применения.

Пусть определены L_1, L_2, \dots, L_{m-1} – множества числительных $m-1$ языков и L_m – множество чисел, соответствующих числительным естественных языков.

Эти множества конечны и имеют одинаковую мощность:

$$|L| = n,$$

так как не любое арифметическое число может быть записано числительным естественного языка. Примером такого арифметического числа является число $10^{1000000}$.

Между элементами множеств установлены взаимно однозначные соответствия, обозначаемые как

$$(L_i, L_j), i = 1, 2, \dots, m; j = 1, 2, \dots, m; i \neq j.$$

Обработка слов языка была ограничена только количественными числительными, так как они переводятся однозначно для языков, поддерживаемых Интернет-приложением.

Задача обработки числительных заключается в определении элемента $e_2 \in L_j$, соответствующего элементу $e_1 \in L_i$ при соответствии (L_i, L_j) . Таким образом, входными данными этой задачи являются элемент $e_1 \in L_i$ и множество L_j , а выходными данными – элемент $e_2 \in L_j$.

Для сокращения числа соответствий был введен промежуточный этап преобразований – множество M , в котором числительные записываются по правилам, определяемым трехуровневой моделью числительного.

Тогда соответствия между числительными сводятся к соответствиям вида:

$$(L_i, M), (M, L_i), i = 1, 2, \dots, m.$$

Введение промежуточного этапа увеличивает количество шагов, необходимых для преобразования числительного, но сокращает количество соответствий («каждый-с-промежуточным-этапом» вместо «каждый-с-каждым»).

Использование промежуточного этапа (трансфера) в машинном переводе текстов на неограниченном языке известно давно [5, 6]. Практическая реализация этого подхода затруднена из-за отсутствия адекватного аппарата представления текста на промежуточном этапе. Решением этой проблемы может стать семантическое представление текста [7].

С практической точки зрения декларативное описание всех соответствий нецелесообразно. Будем использовать для решения задач обработки числительных процедурный подход. Тогда для каждой пары множеств L_i и M необходимы два алгоритма:

1) алгоритм преобразования элемента $e_1 \in L_i$ в соответствующий ему элемент $e_2 \in M$;

2) алгоритм преобразования элемента $e_2 \in M$ в соответствующий ему элемент $e_1 \in L_i$.

Необходимость двух алгоритмов, а не одного, обусловлена различием правил обработки числительных в разных языках. Перевод числительного в направлении «русский язык – немецкий язык» не то же самое, что его перевод в направлении «немецкий язык – русский язык».

Введение промежуточного этапа при обработке количественных числительных расширило функциональность Интернет-приложения, реализующего данное теоретико-множественное представление. Расширение функциональности, в свою очередь, позволило получить преимущество над аналогичными ресурсами, что привело к увеличению количества пользователей Интернет-приложения.

Посетители и пользователи Интернет-приложения. Число пользователей Интернет-приложения с момента начала его работы составило более 3 000 и постоянно растет (рисунок 1).

Данные за 2012 год приведены в период с марта (с момента начала работы Интернет-приложения) по декабрь, а за 2014 год – с января по ноябрь (на момент окончания верстки статьи).

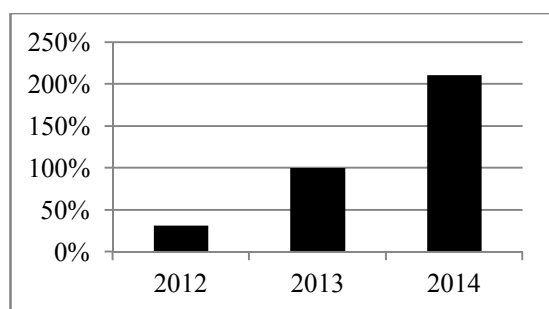


Рисунок 1 – Рост количества пользователей Интернет-приложения за последние 3 года

Посетителями Интернет-приложения являются жители всех постоянно обитаемых материков и континентов (рисунок 2).

Принадлежность пользователя Интернет-приложения стране определяется только по его IP-адресу, так как для работы с Интернет-приложением регистрации не требуется. Некоторые IP-адреса принадлежат не какой-то конкретной европейской стране, а ЕС в целом. Примером такого адреса является IP-адрес 193.110.85.45 (по состоянию на август 2014 года). IP-адреса, принадлежащие ЕС, используются, как правило, крупными компаниями с представительствами в европейских странах.

Рассмотрим особенности перевода количественных числительных в России, Украине, ЕС и США, в которых проживает наибольшее число пользователей Интернет-приложения (таблица 1).

Таблица 1

Страна	Доли от общего числа пользователей	
	III квартал, 2013 г.	III квартал, 2014 г.
Россия	74,8 %	27,8 %
Украина	8,7 %	4,9 %
ЕС	2,7 %	4,8 %
США	0,8 %	44,7 %
Остальные 89 стран	13,0 %	17,8 %

Год назад в работе [8] была проанализирована статистика использования Интернет-приложения по различным критериям и было намечено в качестве одного из направлений дальнейшего развития Интернет-приложения привлекать новых посетителей из зарубежных стран. За прошедший год доли пользователей из ЕС и США увеличились за счёт привлечения новых пользователей (см. рисунок 1).

Российская Федерация. Основные направления перевода пользователей из России представлены в таблице 2. Высокую долю направления перевода «число – русский язык» можно объяснить большой численностью эмигрантов или снижением уровня знания русского языка россиянами. Направление «число – английский язык» имеет относительно низкую долю из-за наличия конкурирующих Интернет-приложений, специализирующихся на этом направлении перевода.



Рисунок 2 – Последние 200 посетителей Интернет-приложения на 11 августа 2014 года (данные получены с помощью счетчика посетителей Statcounter.com)

Таблица 2

Направления перевода	Доля
«Число – русский язык»	44,6 %
«Число – немецкий язык»	32,5 %
«Число – английский язык»	13,2 %
«Число – испанский язык»	2,2 %
Остальные направления	7,5 %

Города России с наибольшим количеством пользователей Интернет-приложения представлены в таблице 3. Эта таблица, а также аналогичная таблица по США характеризуют широту охвата территории страны.

Таблица 3

Города России	Доля
Москва	20,1 %
Екатеринбург	15,2 %
Санкт-Петербург	6,1 %
Нижний Тагил (Свердловская область)	5,3 %
Новосибирск	3,5 %
Самара	2,2 %
Казань	1,6 %
Воронеж	1,5 %
Остальные 215 населенных пунктов	44,5 %

Высокий процент городов Екатеринбург и Нижний Тагил связан с пользователем, который в период с апреля 2013 года по настоящий момент сделал более 300 запросов на обработку числительных в направлении «число – русский язык» с IP-адресов, приписанных к этим городам.

Украина. Из наиболее популярных направлений перевода украинских пользователей можно сделать вывод, что Интернет-приложение в этой стране используется, прежде всего, для учёбы (таблица 4).

Таблица 4

Направления перевода	Доля
«Число – немецкий язык»	41,0 %
«Число – русский язык»	26,6 %
«Число – английский язык»	14,3 %
«Число – испанский язык»	7,4 %
Остальные направления	10,7 %

Украинские пользователи Интернет-приложения проживают в 48 населенных пунктах.

Европейский Союз. В таблице 5 представлены доли направлений пользователей из ЕС.

Учитывались только IP-адреса, принадлежащие ЕС, но не странам, входящим в ЕС.

Таблица 5

Направления перевода	Доля
«Число – русский язык»	55,4 %
«Число – немецкий язык»	31,3 %
«Число – английский язык»	10,2 %
Остальные направления	3,1 %

Соединенные штаты Америки. Одной из причин высокой доли испанского языка в направлениях перевода (таблица 6) является большая численность испаноговорящего населения (см. также таблицу 7).

Таблица 6

Направления перевода	Доля
«Число – испанский язык»	82,8 %
«Число – английский язык»	6,0 %
«Число – русский язык»	4,5 %
«Число – немецкий язык»	1,7 %
«Английский язык – испанский язык»	1,1 %
Остальные направления	3,9 %

Американские города представляют собой небольшие центры, вокруг которых расположены отдельные города-спутники, что отличает их структуры от структуры городов России. Поэтому доли городов США имеют небольшие значения. Например, 1,7 % пользователей из США – это жители Лос-Анджелеса, 1,3 % – жители Хьюстона (Техас), 1,1 % – живут в Нью-Йорке. Американские пользователи Интернет-приложения проживают в более чем 500 населенных пунктах.

Чтобы охарактеризовать охват США, приведем распределение американских пользователей по штатам (таблица 7). Также в таблице 7 представлена численность испаноговорящего населения в этих штатах в 2011 году [9]. Для штата Огайо указана численность испаноговорящего населения из [10] в 2004 году.

Таблица 7

Штаты	Доля	Численность испаноговорящего населения, млн чел.
Калифорния	12,3 %	14,4
Техас	8,6 %	9,8
Нью-Йорк	7,7 %	3,5
Флорида	4,8 %	4,4
Огайо	4,5 %	0,3
Иллинойс	4,3 %	2,1
Остальные 43 штата	57,8 %	

Активность пользователей из России,

Украины, ЕС и США по дням недели. Интернет-приложение используется для работы и учебы, так как большая часть пользователей посещает его в будние дни (таблица 8). Диаграмма с аналогичными данными по всем пользователям Интернет-приложения приведена в [8]. Пользователи из США одинаково активны в выходные дни, тогда как в других странах активность в воскресенье превышает субботную активность. Как было сказано выше, IP-адреса, приписанные к ЕС, принадлежат крупным компаниям. Поэтому в выходные дни активность пользователей из ЕС очень низка. Если в других странах в пятницу наблюдается спад активности, то активность американских пользователей остается на прежнем уровне.

Таблица 8

Дни недели	Россия	Украина	ЕС	США
Понедельник	18,4 %	18,1 %	18,9 %	16,9 %
Вторник	17,1 %	22,9 %	19,2 %	17,3 %
Среда	18,3 %	21,6 %	20,9 %	18,7 %
Четверг	16,7 %	14,4 %	23,1 %	18,1 %
Пятница	11,8 %	8,3 %	11,4 %	17,4 %
Суббота	7,0 %	5,2 %	1,7 %	5,7 %
Воскресенье	10,7 %	9,5 %	4,8 %	5,9 %

Список организаций, с IP-адресов которых были сделаны запросы к Интернет-приложению. Пользователи Интернет-приложения заходят с IP-адресов следующих организаций различных стран мира:

1) учебных заведений и научных институтов:

- Института микропроцессорных вычислительных систем РАН и Межведомственного суперкомпьютерного центра РАН (г. Москва, Россия);

- более 100 американских университетов: Йельского (г. Нью-Хэйвен, Коннектикут), Принстонского (г. Принстон, Нью-Джерси), Айовы (г. Айова-Сити), Алабамы (г. Тускалуса), Аризона (г. Тусон), Аризона (Северный) (г. Флэгстэфф), Арканзаса (г. Файетвилл), Вашингтона (г. Паллман), Вашингтона (Западный) (г. Беллингэм), Виргинии (г. Шарлоттсвилл), Висконсина (г. Ла-Кросс), Гавайев (г. Милилани), Джорджии (Южный) (г. Стэйтсборо), Иллинойса (г. Урбана), Иллинойса (Южный) (г. Карбондэйл), Индианы (г. Блумингтон), Калифорнии (г. Ла-Хойя,

г. Нортридж, г. Санта-Барбара), Калифорнии (политехнический) (г. Помона), Канзаса (г. Лоуренс), Кентукки (Восточный) (г. Ричмонд), Луизианы (г. Батон-Руж, г. Шривпорт), Миссисипи (г. Миссисипи-Стэйт), Миссури (г. Колумбия), Миссури (Центральный) (г. Уорренсбург), Мичигана (г. Энн-Эрбор), Мичигана (Западный) (г. Каламазу), Мэриленда (г. Колледж-Парк, г. Принсес-Энн), Невады (г. Рино), Нью-Йорка (г. Освего, г. Баффало), Огайо (г. Афины, г. Колумбус), Оклахомы (г. Норман), Орегона (г. Юджин), Пенсильвании (г. Индиана, г. Мэнсфилд, г. Вандергрифт), Род-Айленда (г. Кингстон), Северной Каролины (г. Гринсборо, г. Чэпел-Хилл, г. Шарлотт), Северной Каролины (Центральный) (г. Дурэм), Техаса (г. Арлингтон, г. Сан-Маркос), Техаса (A&M) (г. Корпус-Кристи), Техаса (Tech) (г. Лаббок), Флориды (г. Гэйнсвилл), Флориды (Южный) (г. Тампа), Южной Каролины (г. Колумбия), Юты (Южный) (г. Седар-Сити), Бостона (Массачусетс), Виллановы (г. Вилланова, Пенсильвания), Денвера (Колорадо), Ист-Струдсбурга (г. Ист-Струдсбург, Пенсильвания), Кента (г. Кент, Огайо), Клермонта (г. Клермонт, Калифорния), Луисвилла (Кентукки), Майами (г. Корал-Гэйблс, Флорида), Марри (г. Марри, Кентукки), Мемфиса (Теннесси), Нового Орлеана (Луизиана), Нотр-Дама (г. Нотр-Дам, Индиана), Палм-Бич Атлантического (Вест-Палм-Бич, Флорида), Портленда (г. Портленд, Орегон), Радфорда (г. Радфорд, Виргиния), Редландса (г. Редландс, Калифорния), Сан-Диего (Калифорния), Сан-Франциско (Калифорния), Санта-Клары (г. Санта-Клара, Калифорния), Сент-Луиса (Миссури), Сиракуз (г. Сиракузы, Нью-Йорк), Стоуни-Брук (г. Стоуни-Брук, Нью-Йорк), Тулсы (г. Тулса, Оклахома), Хьюстона (Техас), Хай-Пойнта (г. Хай-Пойнт, Северная Каролина), Цинциннати (Огайо), Шиппенсбурга (г. Шиппенсбург, Пенсильвания), Элизабет-Сити (г. Элизабет-Сити, Северная Каролина), Анджело (г. Сан-Анджело, Техас), Беллармайн (г. Луисвилл, Кентукки), Белмонт (г. Нэшвилл, Теннесси), Бентли (г. Уолтэм, Массачусетс), Болл (г. Манси, Индиана), Большой Долины (г. Аллендэйл, Мичиган), Брайнт (г. Смитфилд, Род-Айленд), Бренау (г. Гэйнсвилл, Джорджия), Бэйлор (г. Вако, Техас), ДеСэйлс (г. Сэнтер-Вэлли, Пенсильвания), Джеймса Мэдисона (г. Гариссонбург, Виргиния), Джорджа Мэйсона (г. Ферфакс, Виргиния), Дрейк (г. Дэс-Мойнс, Айова), Дрюри (г. Спрингфилд, Миссури), Дьюк (г. Дарэм, Северная Каролина), Кеннесо (г. Афины, Джорджия), Клемсон (г. Андерсон, Южная Каролина), Колгейт (г. Гамильтон, Нью-

Йорк), Корнерстоун (г. Грэнд-Рэпидс, Мичиган), Кристофер-Ньюпорт (г. Ньюпорт-Ньюс, Виргиния), Кэмпбелл (г. Буис, Северная Каролина), Лойола (г. Балтимор, Мэриленд), Лоуренс (г. Эпплтон, Висконсин), Майами (г. Оксфорд, Огайо), Маркетт (г. Милуоки, Висконсин), Мерсер (г. Макон, Джорджия), Побережного Каролинского (г. Конвэй, Южная Каролина), Пойнт-Лома-Назарин (г. Сан-Диего, Калифорния), Пэйс (Брайклифф-Мэнор, Нью-Йорк), Роуэн (г. Глассборо, Нью-Джерси), Рутгерс (г. Нью-Брансуик, Нью-Джерси), Северо-Западного (г. Эванстон, Иллинойс), Сент-Джозеф (г. Филадельфия, Пенсильвания), Сент-Джон (г. Колледжвилл, Миннесота), Сент-Кэтрин (г. Сент-Пол, Миннесота), Сент-Питер (г. Джерси-Сити, Нью-Джерси), Сент-Томас (г. Сент-Пол, Миннесота), Сент-Эмброуз (г. Дэйвенпорт, Айова), Темпл (г. Филадельфия, Пенсильвания), Тихоокеанского (г. Форест-Гроув, Орегон), Тулэйн (г. Новый Орлеан, Луизиана), Уитворт (г. Спокэйн, Вашингтон), Феррис (г. Биг-Рэпидс, Мичиган), Фордэм (г. Бронкс, Нью-Йорк), Фурман (г. Гринвилл, Южная Каролина), Фэйрли-Дикинсон (г. Тинек, Нью-Джерси), Хавьер (г. Колумбус, Огайо), Хардинг (г. Сирси, Арканзас), Чапмен (г. Оранж, Калифорния), Юго-Западного (г. Джорджтаун, Техас), Виргинского университета Содружества (г. Ричмонд);

- Медицинского института Джона Хопкинса (г. Балтимор, Мэриленд, США) и Американского колледжа кардиологии (г. Вашингтон, округ Колумбия, США);

- канадских университетов Ватерлоо (г. Ватерлоо, провинция Онтарио), Гуэлфа (г. Гуэлф, Онтарио), Западного Онтарио (г. Лондон), МакГилл (г. Монреаль, Квебек) и Моунт-Эллисон (г. Сэквилл, Нью-Брансуик);

- университетов Мельбурна (Австралия) Данди (г. Данди, Великобритания) и Сан-Франциско-де-Кито (г. Кито, Эквадор);

- Московского государственного института международных отношений (Университета) МИД России (МГИМО), Северного арктического федерального университета (г. Архангельск, Россия);

2) государственных учреждений:

- Федерального агентства по управлению государственным имуществом (г. Москва, Россия);

- Национальной службы здоровья (National Health Service) (Великобритания);

- Министерства образования Исландии;

- Национального управления по телекоммуникациям (Administración Nacional De

Telecomunicaciones) (г. Монтевидео, Уругвай);

- администраций американских штатов Коннектикут, Луизиана, Миннесота и Мэн;

3) всемирно известных компаний:

- информационно-технологических компаний Google и Microsoft (обе США);

- автомобилестроительной компании Peugeot-Citroen (Франция);

- фармацевтической компании Novartis (Швейцария);

- химической компании Clariant (Швейцария);

- машиностроительной компании Robert Bosch (Германия);

- сельскохозяйственной компании РусАгро (Россия);

- банка Union Bank Of Switzerland (UBS) (Швейцария);

- аудиторской компании Ernst & Young (CIS) (Россия).

Названия организаций, города и страны в этом списке определены по их IP-адресам.

Заключение. Авторами были получены следующие результаты:

- предложено теоретико-множественное представление задачи обработки количественных числительных естественных языков;

- представлены и проанализированы направления перевода числительных пользователями Интернет-приложения из России, Украины, ЕС и США.

Для обработки статистики авторами статьи было разработано специальное программное обеспечение.

Направлением дальнейшего развития Интернет-приложения является продолжение расширения числа пользователей с учётом особенностей перевода в различных странах.

Библиографический список

1. Пруцков А.В., Цыбулько Д.М. Интернет-приложение метода обработки количественных числительных естественных языков // Вестник Рязанского государственного радиотехнического университета. 2012. № 41. С. 70-74.

2. Пруцков А.В. Обработка числительных естественных языков с помощью формальных грамматик и нормальных алгоритмов Маркова // Вестник Рязанского государственного радиотехнического университета. 2009. № 28. С. 49-55.

3. Пруцков А.В. Определение и генерация сложных форм слов естественных языков при морфологическом анализе и синтезе // Известия Таганрогского государственного радиотехнического университета. 2006. Т. 70. № 15. С. 10-14.

4. Пруцков А.В. Статический и динамический подходы к проектированию подсистем проверки знаний автоматизированных обучающих систем //

Информационные ресурсы России. 2006. № 1. С. 27-29.

5. Искусственный интеллект. В 3 кн. Кн. 1. Системы общения и экспертные системы: справочник / под ред. Э.В. Попова. М.: Радио и связь, 1990. 464 с.

6. Марчук Ю.Н. Компьютерная лингвистика. М.: АСТ; Восток-Запад, 2007. 317 с.

7. Фомичев В.А. Формализация проектирования лингвистических процессоров. М.: Макс Пресс, 2005. 368 с.

8. Пруцков А.В., Цыбулько Д.М. Анализ статистики использования Интернет-приложения

обработки количественных числительных естественных языков // Вестник Рязанского государственного радиотехнического университета. 2013. № 4-1 (46). С. 130-134.

9. Mapping the Latino Population, By State, County and City. II. Ranking Latino Populations in the States. URL: <http://www.pewhispanic.org/2013/08/29/ii-ranking-latino-populations-in-the-states>. Дата обращения 24.07.2014.

10. Total Hispanic Population (most recent) by state. URL: http://www.statemaster.com/graph/peo_tot_his_pop-people-total-hispanic-population. Дата обращения 24.07.2014.

УДК 681.3.06,658.512.2.011.56

К.С. Кульга, П.В. Меньшиков

ОПТИМИЗАЦИЯ ГЕОМЕТРИЧЕСКОГО ПОКРЫТИЯ МНОГОСВЯЗНОГО ОРТОГОНАЛЬНОГО ПОЛИГОНА С ГРАНИЧНЫМИ ПРЕПЯТСТВИЯМИ С УЧЁТОМ КОНСТРУКТОРСКО-ТЕХНОЛОГИЧЕСКИХ ОГРАНИЧЕНИЙ

Рассматриваются результаты теоретических и экспериментальных исследований в области новых моделей и методов для создания автоматизированной информационной системы, предназначенной для решения прямой и обратной задачи оптимального геометрического покрытия многосвязного ортогонального полигона (обечайка) ёмкостного оборудования изготавливаемого на предприятиях нефтегазового и химического машиностроения.

Ключевые слова: *обечайка, оптимальное геометрическое покрытие, конструкторско-технологические ограничения, автоматизированная информационная система, 2D/3D – карта раскроя.*

Введение. Для эффективной работы в условиях рыночной экономики предприятиям нефтегазового и химического машиностроения необходимо принимать меры по повышению рентабельности выпускаемых изделий. Одна из этих мер – снижение себестоимости изделий и издержек, сокращение сроков технологической подготовки производства.

Одной из самых ответственных и дорогостоящих деталей ёмкостного оборудования (например, теплообменные и колонные аппараты), изготавливаемых предприятиями нефтегазового и химического машиностроения, является обечайка [1-3]. Обечайка – открытый с торцов цилиндрический или конический барабан (без днищ), являющийся заготовкой для паровых котлов, баков, резервуаров и других металлоконструкций из листового сортамента. Фрагмент чертежа обечайки колонного аппарата показан на рисунке 1. Обечайки колонного оборудования имеют значительные габаритные размеры (например, 60600x3200x18 мм), могут включать несколько

секций с разными размерами поперечного сечения, в том числе из разного материала. Эта деталь сваривается из группы листового сортамента (лент). Обечайки изготавливаются из дорогостоящих нержавеющей или биметаллических заготовок, и доля стоимости обечайки в плановой и фактической себестоимости изделия может достигать 70 %, а трудовые затраты на выполнение сварочных операций (кольцевые и продольные сварные швы обечайки – корпусные сварные швы) – до 50 % от суммарных норм времени на изготовление всего изделия.

В большинстве случаев (кроме самых простых изделий) требуется решать задачу расчёта карты раскроя обечайки. В этом случае определяется конфигурация корпусных сварных швов обечайки, при которой выполняются все конструкторско-технологические ограничения, учитываются технологические возможности оборудования и обеспечивается минимизация норм расхода основных и вспомогательных материалов, а также норм времени на заготовительные,

сварочные и сборочные операции.

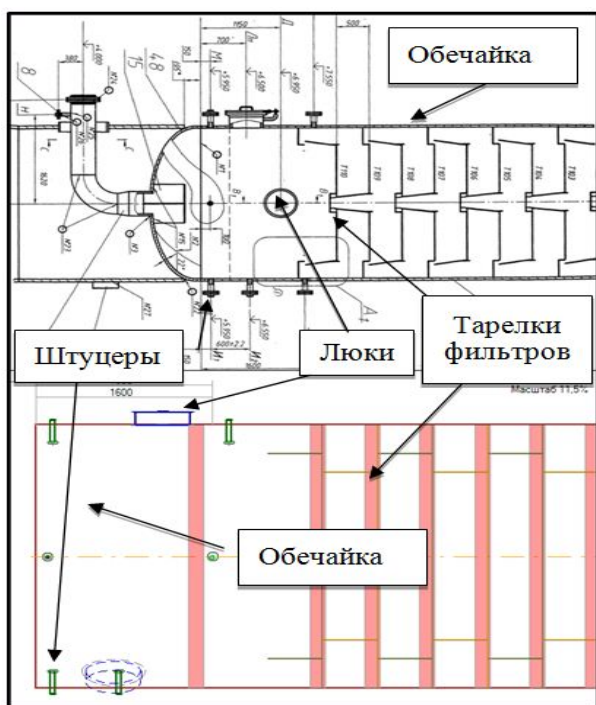


Рисунок 1 – Фрагменты чертежа и расчетной карты обечайки колонного аппарата

В дополнение к вышеуказанной задаче предприятия решают также следующие: предварительные расчёты плановой себестоимости изделий для принятия решения об участии в тендерах (на основе данных опросных листов); обоснование оптимальных размеров заготовки для изготовления изделия. Для решения этих задач требуется учёт деловых отходов заготовок на складах, соблюдение конструкторско-технологических требований, данные логистики поставок заготовок и себестоимости изделия.

Таким образом, для повышения рентабельности производства необходима автоматизация решения прямой и обратной (расчётное обоснование размера заготовок) задач оптимального раскроя обечайки. В общем случае эти задачи относятся к классу задач оптимального геометрического покрытия ортогонального многосвязного полигона (развертка обечайки) с препятствиями (технологические ограничения) набором прямоугольников заданных размеров (тип заготовки: лист или лента). Анализ публикаций [4-7] показал, что задача раскроя такой сложной детали, как обечайка, не может быть сведена к традиционной постановке задачи геометрического покрытия области с препятствиями по следующим причинам.

■ Препятствия в стандартной постановке рассматриваются только как зоны, исключенные из плана геометрического покрытия, в то время как при раскрое обечайки технологические огра-

ничения распространяются в основном на расположение корпусных сварных швов, т.е. границ покрываемых элементов, причем зоны этих ограничений тоже необходимо покрыть. Введём наименование такого вида препятствия – *граничное препятствие*. Учёт граничных препятствий не рассматривался в опубликованных работах, посвященных решению задач геометрического покрытия [4-7].

■ Разнородность препятствий. Большая часть технологических ограничений относится к расположению только кольцевых или только продольных сварных швов, а некоторые ограничения дополнительно обладают условием «срабатывания». Например, запрещено размещение кольцевого сварного шва под укрепляющим кольцом люка (см. рисунок 1), если диаметр люка меньше 300 мм. В работах [4-7] все препятствия рассматриваются как однородные.

■ Существуют динамические препятствия, т.е. ограничения, создаваемые во время раскроя покрываемыми элементами. Например, не допускаются крестообразные сварные швы или параллельные сварные швы, расстояние между которыми меньше определенного значения Δ_{\min} .

■ Необходимо решение не только прямой, но и обратной задачи покрытия.

■ Необходимость раскроя многосекционных аппаратов, в которых каждая секция может иметь разное сечение (диаметр и толщину стенок) и материал. Наряду с этим, необходим учёт нераскраиваемых секций, имеющих разную геометрическую форму (конус, цилиндр).

■ Учёт типа и параметров заготовки, например, величины кропления листового сортамента, анизотропии материала заготовки.

■ В качестве критерия оптимальности выступает не единственный параметр (например, суммарная площадь использованных заготовок), а набор разнородных критериев, относящихся к технологичности изготовления и себестоимости изделия (расход основных и вспомогательных материалов, объемы деловых и технологических отходов, периметры корпусных швов, нормы времени и др.).

■ Необходим учёт групп ёмкостных аппаратов, имеющих различные технологические требования [1, 2], а также параметров точности технологического оборудования, ширины корпусных швов, особенностей конструкции изделия, в которую входят следующие детали и сборочные единицы (ДСЕ): штуцеры, люки, цапфы, опоры, укрепляющие кольца и др. Для каждой ДСЕ изделия существует свой набор конструкторско-технологических ограничений на расположения корпусных сварных швов.

▪ Влияние на результаты геометрического покрытия конструкции и расположения *внутренних* ДСЕ обечайки, например тарелок фильтров [2].

▪ Необходима программная взаимосвязь исходных и результирующих наборов данных предлагаемой автоматизированной информационной системы (АИС) с интегрированной АИС (ИАИС) предприятия (например, конструкторская и технологическая документация, результаты нормирования расхода материалов и труда, данные учёта остатков и деловых отходов заготовок). В опубликованных работах [4-7] не реализована программная интеграция наборов данных в едином информационном пространстве с ИАИС предприятия.

▪ Наряду с составлением двухмерного (2D) плана геометрического покрытия, необходима генерация трёхмерной (3D) твердотельной параметрической модели раскроенного аппарата (3D карта раскроя) для обеспечения правильности сборки и изготовления изделия.

Цель работы – разработка моделей и методов создания АИС оптимального геометрического покрытия многосвязного ортогонального полигона (развертки обечайки ёмкостного оборудования), включая программное взаимодействие на уровне наборов данных и управления в едином информационном пространстве (ЕИП) предприятия с ИАИС *Stalker PLM v7.x* [9].

Теоретическая часть. Результаты исследований, проведенные в работе [5], показали, что в общем случае задачи покрытия относятся к *NP*-трудным, т.е. алгоритм точного решения таких задач за полиномиальное время не найден. Практически на всех наборах исходных данных методы поиска точного решения потребуют слишком много времени. Таким образом, для решения практических задач раскроя листового сортамента (ленты) для изготовления обечайки целесообразно использовать эвристические методы поиска оптимального решения.

Создание программного обеспечения (ПО) АИС оптимального геометрического покрытия обечайки основывалось на методологии, разработанной в монографии [9]. Данная методология предусматривает поэтапное решение задачи создания АИС на основе реинжиниринга существующих бизнес-процессов (БП), разработку объектно-ориентированной функциональной, а затем информационно-математической модели с последующим созданием на их основе ПО базовых информационных технологий АИС.

Реинжиниринг БП и создание функциональной модели (ФМ) АИС. Анализ существующих

АИС оптимального раскроя листового сортамента и результатов их применения на предприятиях нефтегазового и химического машиностроения выявил отсутствие комплексного применения критериев для решения задачи геометрического покрытия многосвязного ортогонального полигона (обечайки). В свою очередь, решение этой задачи на основе опыта квалифицированных инженеров предприятия требует значительных затрат времени (от трёх до семи дней по данным фотографии дня), при этом не гарантируется оптимальность полученного решения и отсутствие ошибок. На основе результатов реинжиниринга БП заготовительного производства ёмкостного оборудования разработана ФМ, отличающаяся тем, что формализовано описывает взаимодействие БП технической подготовки производства обечайки, а также интегрирует этапы функционального моделирования и создания ПО АИС [3].

Объектно-ориентированная ФМ разработана с применением методологии *Rational Unified Process (RUP)* и платформенно-независимого объектно-ориентированного языка *UML (Unified Modeling Language)*. На рисунке 2 приведён пример одной из диаграмм ФМ АИС: реализация БП преобразования технологических ограничений, связанных с тарелками фильтров, в граничные препятствия на расчётной схеме карты раскроя обечайки колонного аппарата.

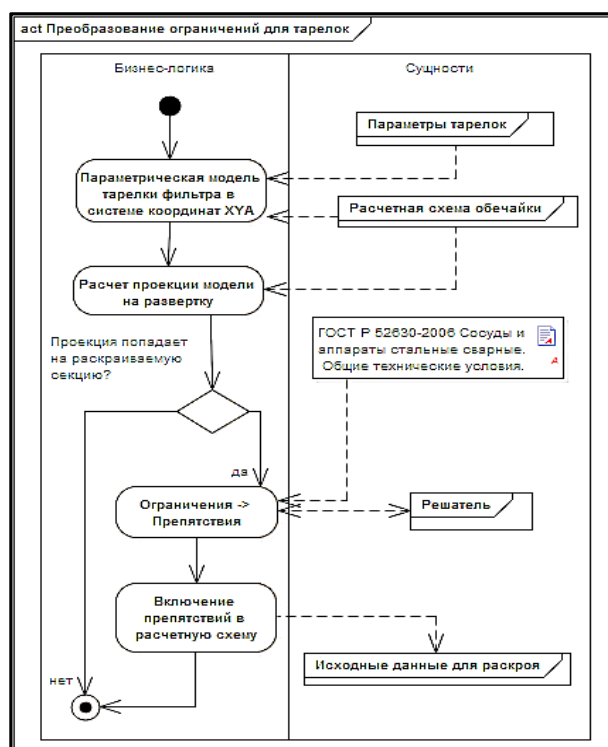


Рисунок 2 – Функциональная модель БП преобразования технологических ограничений для ДСЕ «Тарелка фильтра»

Единая информационно-математическая модель (ИММ) АИС. ИММ АИС отличается реализацией модифицированного метода ветвей и границ для решения прямой и обратной задач двумерного геометрического покрытия ортогонального многосвязного полигона с граничными препятствиями на основе определения конструкторско-технологических ограничений для размещения геометрических объектов, применением обобщенного критерия оптимальности, а также новым методом автоматического формирования параметрических 3D геометрических моделей карт раскроя обечайки с применением API (Application Programming Interface)-функций CAD (Computer Aided Designer)-систем [3].

Процессор АИС Stalker ROB. В качестве исходного набора данных для решения задачи оптимального геометрического покрытия ортогонального многосвязного полигона используется параметрическая 2D геометрическая модель карты обечайки. Для дополнительного контроля исходных данных реализована функция автоматического формирования 3D геометрической модели карты обечайки. Расположение ДСЕ задается визуально, а их параметры указываются с помощью интерактивных форм. Для уменьшения времени ввода реализованы функции привязки объектов к базовым поверхностям других объектов, ассоциативная связь активных полей формы с объектами сборочного чертежа колонного аппарата (см. рисунок 1) и операция копирования набора данных тарелок одинакового типа.

Группы ограничений относятся к расположению корпусных сварных швов, ортогональных по отношению друг к другу [1]. Это позволяет преобразовать любые ограничения независимо от их формы в прямоугольные боксы, стороны которых параллельны направлениям кольцевых и продольных сварных швов обечайки. У каждого бокса имеется условие «срабатывания» ограничения, например, ограничение только для кольцевых сварных швов, ограничение для сварных швов с катетом больше 20 мм или ограничение для Т-образных швов. На рисунке 1 показаны соответствующие фрагменты 2D геометрической модели карты обечайки и чертежа (на главном виде чертежа не показывается угловое положение ДСЕ). Представим обечайку колонного аппарата в виде развертки (рисунок 3).

В общем случае развертка будет представлять собой многосвязный ортогональный полигон. Каждой секции корпуса будет соответствовать прямоугольник с длиной L_s и шириной W_s , а также номенклатурным обозначением материала секции M_s , $s = \overline{1, n}$, где n – количество секций корпуса. Введем прямоугольную систему координат: оси Ox и Oy совпадают соответ-

ственно с нижней и боковой сторонами развертки первой секции, остальные секции также располагаются ортогонально по отношению к этой системе координат. Координаты нижнего и левого угла секции задаются параметрами X_s и Y_s ($X_1 = 0$, $Y_1 = 0$). Для каждой ДСЕ и геометрических объектов карты обечайки производится преобразование ограничений в прямоугольники, именуемые запретными зонами.

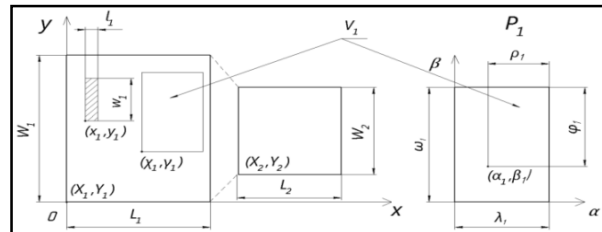


Рисунок 3 – Расчётная схема развертки обечайки

Классифицировано три типа запретных зон:

- ограничения на расположение кольцевых сварных швов Z_R ;
- ограничения на расположение продольных сварных швов Z_L ;
- ограничения на расположение Т-образных сварных швов Z_T , характеризуемых следующими параметрами:

$$(x_v, y_v), (l_v, w_v), v = \overline{1, \mu},$$

где x_v, y_v – координаты нижнего и левого угла ограничительной зоны, w_v – её ширина, l_v – её длина, μ – количество ограничительных зон этого типа.

При преобразовании рассчитывается точная проекция контуров ДСЕ на цилиндрическую поверхность обечайки, в том числе для случая с нерадиальным расположением ДСЕ.

К исходным данным относится информация о заготовке (группе заготовок), в том числе о деловых отходах, оставшихся от предыдущих сессий раскроя (ИАИС *Stalker PLM v7.x*. [9]).

Заготовки представляют собой множество прямоугольников:

$$P = \{P_1, P_2, \dots, P_m\}, P_u = \langle (\omega_u, \lambda_u), \mu_u \rangle,$$

где (ω_u, λ_u) – ширина и длина листа с учётом величины кромления заготовки, μ_u – обозначение материала заготовки.

Процессор АИС Stalker ROB. В задаче требуется найти множество прямоугольных боксов V :

$$V = \{V_1, V_2, \dots, V_q\},$$

$$V_j = \langle (\chi_j, \gamma_j), (\rho_j, \varphi_j), u_j, (\alpha_i, \beta_i) \rangle,$$

где (χ_j, γ_j) – координаты нижнеголевого угла бокса на развертке корпуса, (ρ_j, φ_j) – его длина и ширина, u_j – индекс заготовки P_u , частью которой является этот бокс, (α_i, β_i) – координаты нижнего левого угла бокса на заготовке, q – мощность множества.

Множество V должно удовлетворять следующим условиям.

1. Ребра боксов параллельны ребрам области:

$$(r_x^i = \rho_i) \cap (r_y^i = \varphi_i), i = \overline{1, q},$$

где r_x^i, r_y^i – проекции прямоугольника V_i на оси координат Ox и Oy .

Неперекрывание боксов с гранями области заготовки: $\forall i = \overline{1, q}$

$$(\alpha_i \geq 0) \cap (\beta_i \geq 0) \cap (\alpha_i + \rho_i \leq \lambda_{ui}) \cap (\beta_i + \varphi_i \leq \omega_{ui}).$$

2. Взаимное неперекрывание боксов на заготовке: $\forall i \neq j; i, j = \overline{1, q}$

$$\left((\alpha_i \geq \alpha_j + \rho_j) \cup (\alpha_j \geq \alpha_i + \rho_i) \right) \cap \left((\beta_i \geq \beta_j + \varphi_j) \cup (\beta_j \geq \beta_i + \varphi_i) \right).$$

3. Неperекрывание боксов с гранями ции: $\forall i = \overline{1, q}$,

$$(\chi_i \geq X_j) \cap (\gamma_i \geq Y_j) \cap (\chi_i + \rho_i \leq X_j + L_j) \cap (\gamma_i + \varphi_i \leq Y_j + W_j),$$

где j – й индекс секции, на которой располагается i -й бокс.

4. Взаимное неперекрывание боксов на развертке: $\forall i \neq j; i, j = \overline{1, q}$

$$\left((\chi_i \geq \chi_j + \varphi_j) \cup (\chi_j \geq \chi_i + \varphi_i) \right) \cap \left((\gamma_i \geq \gamma_j + \rho_j) \cup (\gamma_j \geq \gamma_i + \rho_i) \right).$$

5. Неperекрывание вертикальных граней боксов с ограничительными зонами на расположение кольцевых швов: $\forall i = \overline{1, q}, \forall j = \overline{1, \mu_R}$

Левая грань:

$$\left((\chi_i - \frac{\tau_R}{2} - \delta \geq x_j + l_j) \cup (x_j \geq \chi_i + \frac{\tau_R}{2} + \delta) \right) \cap A$$

Правая грань:

$$\left((\chi_i + \varphi_i - \frac{\tau_R}{2} - \delta \geq x_j + l_j) \cup (x_j \geq \chi_i + \varphi_i + \frac{\tau_R}{2} + \delta) \right) \cap A,$$

где $A = \left((\gamma_i \geq y_j + w_j) \cup (y_j \geq \gamma_i + \rho_i) \right)$, τ_R – катет кольцевых сварных швов, δ – минимальное допустимое расстояние между двумя швами.

6. Неperекрывание горизонтальных граней боксов с ограничительными зонами на расположение продольных сварных швов: $\forall i = \overline{1, q}, \forall j = \overline{1, \mu_L}$

Нижняя грань:

$$\left((\gamma_i - \frac{\tau_L}{2} - \delta \geq y_j + w_j) \cup (y_j \geq \gamma_i + \frac{\tau_L}{2} + \delta) \right) \cap A$$

Верхняя грань:

$$\left((\gamma_i + \rho_i - \frac{\tau_L}{2} - \delta \geq y_j + w_j) \cup (y_j \geq \gamma_i + \rho_i + \frac{\tau_L}{2} + \delta) \right) \cap A,$$

где $A = \left((\chi_i \geq x_j + l_j) \cup (x_j \geq \chi_i + \varphi_i) \right)$, τ_L – катет продольных сварных швов.

7. Непопадание углов боксов на ограничительные зоны на расположение Т-образных сварных швов: $\forall i = \overline{1, q}, \forall j = \overline{1, \mu_T}, \forall k = \overline{1, 4}$

$$\left((u_i^k \geq x_j + l_j) \cup (x_j \geq u_i^k) \right) \cap \left((v_i^k \geq y_j + w_j) \cup (y_j \geq v_i^k) \right),$$

где u_i^k, v_i^k – координаты по осям Ox и i -го угла бокса V_i .

8. Отсутствие крестообразных сварных швов, параллельные швы должны находиться на расстоянии не менее σ : $\forall i \neq j; i, j = \overline{1, q}, \forall k, m = \overline{1, 4}$

$$\left((u_i^1 - u_j^3) \geq \sigma \right) \cap \left((v_i^1 - v_j^3) \geq \sigma \right) \cap \left((u_i^2 - u_j^4) \geq \sigma \right) \cap \left((v_i^2 - v_j^4) \geq \sigma \right),$$

где u_i^k, v_i^k – координаты по осям Ox и i -го угла бокса V_i ,

u_j^m, v_j^m – координаты по осям Ox и i -го угла бокса V_j .

9. Обозначение заготовки и материала секции должно совпадать:

$$\forall i = \overline{1, q}, \mu_{ui} = M_j,$$

где j – й индекс секции, на которой располагается i -й бокс.

10. Прямоугольные боксы должны покрывать все секции:

$$\sum_{i=1}^q \varphi_i \cdot \rho_i = \sum_{j=1}^n L_j \cdot W_j.$$

Разработанный эвристический алгоритм, основанный на предлагаемом методе, отличается тем, что на каждом шаге отбирается N лучших вариантов по локальному критерию оптимальности, и для каждого из них определяются все возможные варианты следующего шага. Производится отсечение заведомо неудачных вариантов, а также одинаковых вариантов раскроя, полученных разными путями. Оптимальность варианта оценивается значением целевой функции, которая является сверткой частных критериев оптимальности на основе метода относительного отклонения от идеальной точки [8]:

$$F_T = \sqrt{\sum \lambda_i \cdot R_i^2}, F_T \rightarrow \min,$$

$$R_i = \begin{cases} \left(1 - \frac{F_i}{F_{iMAX}} \right), F_i \rightarrow \max \\ \left(1 - \frac{F_i}{F_{iMIN}} \right), F_i \rightarrow \min \\ 0, (F_i \rightarrow \min) \cap (F_i = 0), \end{cases}$$

где R_i – относительное отклонение i -го критерия от оптимального значения;

F_i – значение i -й целевой функции для данного варианта;

F_{iMIN}, F_{iMAX} – оптимальное значение i -го критерия целевой функции;

λ_i – весовой коэффициент для i -го критерия.

Использование безразмерных отношений типа $\frac{F_i}{F_{iMAX}}$ позволяет суммировать частные критерии разной природы в одном выражении.

К критериям оптимальности относятся следующие показатели.

1. Количество использованных в раскрое листов:

$$F_1 = \{ \{ P_i | \exists V_j, u_j = i \} \}, F_1 \rightarrow \min.$$

Для глобального критерия F_{1MIN} вычисляется, как количество листов максимальной площади среди доступных заготовок, полностью покрывающих развертку обечайки:

$$F_{1MIN} = \sum_{i=1}^n (L_i \cdot W_i) / S_{iMAX},$$

где L_i, W_i – длина и ширина i -й секции, S_{iMAX} – максимальное значение площади заготовки, подходящей по её свойствам (толщина и материал) для покрытия i -й секции:

$$S_{iMAX} = \text{MAX}_{u=1}^m (\omega_u \cdot \lambda_u), (\mu_u = M_i).$$

Для локального критерия величина F_{1MIN} равна минимальному количеству использованных листов на данной итерации (но не менее единицы).

2. Количество использованных целых листов:

$$F_2 = \{ \{ P_i | \exists V_j, (p_j = \lambda_i) \cap (\omega_j = \varphi_i) \} \}, F_2 \rightarrow \max.$$

Значение F_{2MAX} равно общему количеству использованных в раскрое листов (F_1).

3. Суммарная длина кольцевых сварных швов (швы, проходящие по границам секции, не учитываются):

$$F_3 = \sum_{i=1}^q \varphi_i - \sum_{j=1}^n L_j, F_3 \rightarrow \min.$$

Для глобального критерия величина F_{3MIN} вычисляется как минимально возможная длина кольцевых сварных швов для заданного набора заготовок:

$$F_{3MIN} = \sum_{i=1}^n W_i \cdot \frac{L_i}{\lambda_{iMAX}},$$

где λ_{iMAX} – максимальная ширина листа, подходящего по обозначению материала заготовки i -й секции.

Для локального критерия величина F_{3MIN} равна минимально достигнутому значению F_3 среди вариантов на данной итерации.

4. Суммарная длина продольных сварных швов:

$$F_4 = \sum_{i=1}^q \rho_i, F_4 \rightarrow \min.$$

Для глобального критерия величина F_{4MIN} равна сумме длин всех секций обечайки, так как

технологический процесс изготовления каждой секции обечайки включает операцию сварки как минимум одного продольного шва. Для локального критерия величина F_{4MIN} равна минимально достигнутому значению F_4 среди вариантов раскроя на данной итерации.

5. Суммарная площадь деловых и технологических отходов, например, величина последней площади рассчитывается по формуле:

$$F_5 = \sum_{i=1}^t S_{it}, F_5 \rightarrow \min$$

S_{it} – площадь i -го технологического отхода, t – общее количество технологических отходов.

Использование нескольких критериев оптимизации позволяет управлять решением задачи раскроя с помощью коэффициентов λ_i . Например, минимизировать количество используемого листового сортамента. Значения λ_i по умолчанию установлены методом экспертных оценок.

Для пользователя АИС *Stalker ROB* предлагаются следующие варианты настроек коэффициентов λ_i : сбалансированный; приоритет снижению технологической сложности; приоритет снижению нормы расхода материалов.

Постпроцессор АИС Stalker ROB. Основными функциями постпроцессора являются автоматическое построение геометрической модели 2D/3D – карты раскроя обечайки (рисунок 4), а также и формирование печатных форм отчетов с наборами данных для технологических операций резки и сварки секций.

Для апробации результатов научных исследований были проведены экспериментальные расчёты оптимального геометрического покрытия обечайки девяти видов ёмкостного оборудования (аппаратов) перед их запуском в производство. Результаты работы АИС *Stalker ROB* для аппарата №9 показаны на рисунке 4.

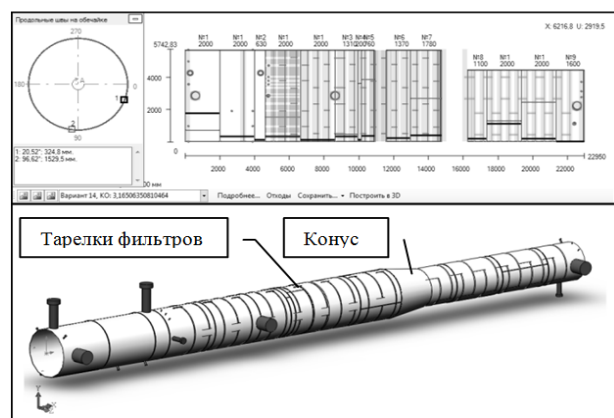
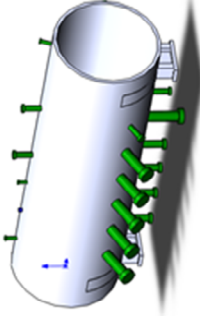
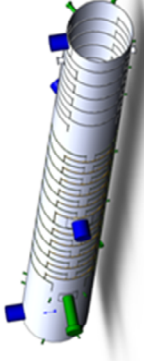
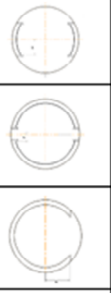
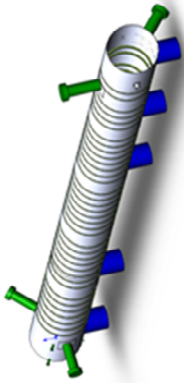

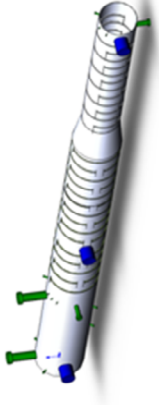



Рисунок 4 – 2D/3D – карты раскроя обечайки колонного аппарата

Исходные данные группы аппаратов для экспериментальных исследований приведены в таблице 1.

Таблица 1 – Исходные данные для экспериментальных исследований оптимального геометрического покрытия обечайки

№	Трёхмерная твердотельная геометрическая модель ёмкостного оборудования (аппарата)	Габаритные размеры аппарата, мм	Материал, типоразмер листового сортамента заготовок	Тип аппарата	Количество ДСЕ	Общее количество секций/нераскраиваемое количество секций	Наличие / тип тарелок фильтров
2		4840x1500x45	Биметалл 09Г2С и 08Х18Н10Т, 6000x1500x45 мм	Теплообменный (4256)	28	1/0	Нет
4		29500x2400x20	09Г2С, 8000x2000x20 мм	Колонна (2610)	82	1/0	Есть/Типы 1-3 
5		19600x1400x8	09Г2С, 6000x1500x8 мм	Колонна (7986)	53	3/1	Есть/Тип 3 
9		14700x1800x14, 8250x1400x12	09Г2С, 8000x2000x14 мм и 6000x1500x12 мм	Колонна (7935)	49	5/2	Есть/Типы 1 и 3 

Результаты экспериментов для группы аппаратов, проведённые квалифицированным инженером как вручную, так и с помощью АИС *Stalker ROB*, оценивались по двум критериям:

1) время, затраченное на выполнение операций оптимального раскроя обечайки, включая подготовку расчетной схемы, непосредственно расчёт, формирование карт раскроя и отчётов;

2) качественные показатели раскроя, формализованные в виде обобщенного критерия оптимальности (значения коэффициентов λ_i были приняты по умолчанию).

Для сравнительной оценки результатов решения задачи оптимального раскроя обечайки были разработаны:

– коэффициент сложности расчетной схемы, учитывающий количество ДСЕ и секций обечайки с разным сечением;

– коэффициент сложности раскроя, учитывающий процент площади обечайки, покрытый запрещенными зонами.

Результаты экспериментальных исследований аппаратов приведены в таблицах 2-4.

Таблица 2 – Результаты экспериментов

№	Сложность расчетной схемы	Время раскроя, мин	
		Ручной	АИС <i>Stalker ROB</i>
1	0,67	20	15
2	1,20	120	25
3	1,00	90	20
4	2,73	240	50
5	3,53	300	55
6	1,80	190	45
7	1,07	120	25
8	0,87	100	30
9	4,9	320	50

Таблица 3 – Результаты экспериментов

№	Сложность раскроя	Целевая функция F_T	
		Ручной	АИС <i>Stalker ROB</i>
1	0,56	0,530	0,530
2	0,81	0,502	0,415
3	0,58	0,451	0,392
4	0,34	0,342	0,278
5	0,54	0,418	0,356
6	0,51	0,405	0,366
7	0,69	0,510	0,475
8	0,46	0,441	0,331
9	1,1	0,531	0,381

Таблица 4 – Средние значения критериев

Способ раскроя обечайки аппарата	Время раскроя, мин	Целевая функция F_T
Ручной	166,7	0,46
АИС <i>Stalker ROB</i>	35	0,39
Изменение значения критериев	476,19%	17,2%

Заключение. На основе предложенных моделей и методов разработано и экспериментально апробировано ПО АИС *Stalker ROB*, реализующее прямую и обратную задачи оптимального геометрического покрытия ортогонального многосвязного полигона с граничными препятствиями на основе определения конструкторско-технологических ограничений на стадиях предварительных расчётов, технической подготовки, а также информационную интеграцию в ЕИП предприятия с ИАИС *Stalker PLM v7.x* [9].

Библиографический список

- ГОСТ Р 52630-2006. Сосуды и аппараты стальные сварные. Общие технические условия – М.: Стандартиформ. – 2007. – 68 с.
- ОСТ 26-805-73.Тарелки сетчатые колонных аппаратов. Тип и параметры. Конструкция и размеры. – Введен 1-4-1976. –М.: Министерство химического и нефтяного машиностроения. –1976. – 23 с.
- Кульга К.С., Меньшиков П.В. Прямая и обратная задачи двумерного раскроя многосвязного ортогонального полигона (обечайка) на основе определения конструкторско-технологических ограничений для размещения геометрических объектов и обобщенного критерия оптимальности // CAD/CAM/CAE Observer. #7(83). –2013.– С.70-75.
- Мухачева Э.А. Рациональный раскрой промышленных материалов. Применение АСУ. – М.: Машиностроение. –1984. – 176 с.
- Задача о покрытии множества: сложность, алгоритмы, экспериментальные исследования / Еремеев А.В., Заозерская Л.А., Колоколов А.А. // Дискретный анализ и исследование операций. – Сер. 2. – 2000. – Т. 7. – № 2. – С. 22-46.
- Хасанова Э.И. Проектирование размещения геометрических объектов на многосвязном ортогональном полигоне [Текст]: дис. ...канд. тех. наук: 05.13.12: Уфа. – 2010. – 188 с.
- Телицкий С.В. Оптимизация многокритериального геометрического покрытия полигона на основе условных оценок с учетом технологических ограничений [Текст]: дис. ... канд. тех. наук: 05.13.01: Уфа. – 2013.– 179 с.
- Гольдштейн А.Л. Метод отклонений для многокритериальных задач – Пермь: Изд. ПГТУ. – 2010. – 27 с.
- Модели и методы создания интегрированной информационной системы для автоматизации технической подготовки и управления авиационным и машиностроительным производством/ Кульга К.С., Кривошеев И.А. – М.: Машиностроение. – 2011. – 377 с.

УДК 681.3+004.85

В.Н. Азаров, С.С. Фомин**ВИРТУАЛЬНЫЕ ПРАКТИКУМЫ КАК ОСНОВНОЕ ЗВЕНО
НЕПРЕРЫВНОГО ОБРАЗОВАНИЯ В ОБЛАСТИ
ИНФОРМАЦИОННО-КОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ**

Анализируется состояние и возможности технологических компонентов систем дистанционного обучения для обеспечения непрерывного и целостного дистанционного обучения в области информационно коммуникационных технологий. Приводится обобщенная структура виртуального практикума. Обсуждается возможность организации обучения и переподготовки специалистов на рабочих местах.

Ключевые слова: непрерывное образование, дистанционное обучение, виртуальный практикум, виртуальная среда.

Введение. Современный этап развития вычислительной техники, информационных и коммуникационных технологий характеризуется большой динамикой. В связи с этим возникает необходимость подготовки большого количества специалистов и квалифицированных пользователей этих технологий.

В рамках вузовских образовательных программ не удаётся подготовить нужное количество специалистов необходимой квалификации. Причинами являются: нехватка часов в образовательных программах для практических работ и достижения студентами профессионального уровня, несоответствующее оснащение дисплейных классов, кризис преподавательских кадров вузов (старые не успевают, молодых нет).

Похожие проблемы возникают также в сферах повышения квалификации и переподготовки кадров. Однако специалисту в области ИКТ необходимо постоянно поддерживать профессиональный уровень в течение всего периода своей активности. *Цель работы* – определить подходы и схему организации сопровождения выполнения практических заданий при изучении дисциплин в области ИКТ в дистанционном режиме, проанализировать средства реализации таких практикумов, предложить пути решения проблемы.

Теоретическая часть. Около трех десятилетий назад благодаря усилиям ЮНЕСКО и Совета Европы появилась стратегия **образования в течение всей жизни (Lifelong Learning)** (в отечественной интерпретации – непрерывное образование). Это был ответ на ситуацию, когда люди учатся в течение всей жизни, в то время как возможности для получения образования предоставляются в основном на раннем этапе жизни,

когда преобладает в большей степени формальное образование [5, 6]. В настоящее время положение дел изменилось мало.

Основными чертами непрерывного образования являются следующие:

- Системный взгляд. Непрерывное образование должно планироваться как система, охватывающая весь жизненный цикл и состоящая из всех видов формального и неформального обучения.

- Центрированность на учащемся, т.е. смещение акцентов с поставляющей стороны на потребляющую сторону.

- Мотивация к учебе. Мотивация является основой для обучения, которое продолжается в течение всей жизни.

Сейчас концепция непрерывного образования на технологическом уровне и в содержательном плане для некоторых направлений образования в значительной степени поддерживается многочисленными образовательными инициативами на основе *дистанционных технологий обучения* [9, 10, 11, 12]. Как правило, такая поддержка достаточна для большинства (если не для всех) гуманитарных дисциплин. Простого контента (текст, гипертекст, аудио- и видеофрагменты) обычно достаточно для формирования содержательной части образовательных программ. Доступ к такому контенту осуществляется на уровне обычных браузеров с любых средств вычислительной техники, в том числе – мобильных устройств.

В области технических дисциплин (а особенно, ИКТ) дистанционной технологией поддерживается только часть цикла обучения – изучение теоретического материала, выполнение тестов. Выполнение практических заданий, ра-

бота с изучаемыми объектами (без навыков работы с изучаемыми объектами – нет специалиста), как правило, требует присутствия обучающего в дисплейных классах обучающей организации и сопряжено со значительным количеством трудностей по созданию и сопровождению таких практикумов. Однако с развитием систем виртуализации ситуация с практикумами стала меняться. Уже сейчас можно создавать блоки практических заданий (виртуальные практикумы), доступные в режиме дистанционного обучения.

Перечислим технологические компоненты системы дистанционного обучения, необходимые для построения системы непрерывного образования для подготовки специалистов и поддержки их профессионального уровня в области ИКТ.

1) *Компьютеры.* В результате развития технологий, появляются шести-, восьми-, шестнадцати-ядерные процессоры (и это еще не предел). Растет пропускная способность интерфейсов компьютеров, а также емкость систем хранения данных. Совершенствуются аппаратные средства поддержки виртуализации.

2) *Информационно-коммуникационные технологии.* Пропускная способность глобальных сетей и сетей уровня города постоянно растёт. Телекоммуникационные фирмы доводят до конечного потребителя высокоскоростные оптоволоконные каналы связи, снимая проблемы доступа практически к любому контенту дистанционных образовательных систем.

3) *Системы управления обучением.* С появлением свободно распространяемых систем управления обучением (LMS – Learning Management Systems) практически исчезла проблема выбора таких систем. Все эти системы обладают равноценной функциональностью, и выбор одной из них является делом вкуса. Так или иначе большинство обсуждений LMS в Интернет связано с системой **Moodle** [13], хотя Moodle не является единственно возможной системой для организации дистанционного обучения. Необходимо отметить, что LMS Moodle получила большое распространение благодаря значительному количеству расширений и открытому коду. Системы управления обучением имеют в своём составе средства общения слушателей друг с другом и слушателей с преподавателем. Как правило – это различного вида форумы.

4) *Мобильные устройства.* Бурное развитие мобильных устройств и их сегодняшние возможности позволяют предположить, что в ближайшем будущем они позволят получить доступ не только к простому контенту образовательных

систем, но и позволят достаточно удобно работать с изучаемыми объектами напрямую, используя технологии виртуализации. Учитывая уровень распространения мобильных устройств, можно сказать, что учебные материалы любого уровня будут доступны для любого желающего повысить свой профессиональный уровень.

5) *Средства виртуализации.* В целом выделяют следующие преимущества использования виртуализации [1, 2, 8].

- *Сокращение затрат на приобретение и поддержку оборудования.* Технология виртуализации позволяет иметь на одном физическом сервере несколько независимых друг от друга *виртуальных серверов* со своим набором служб и своими характеристиками, которые могут существовать как независимые узлы сети.

- *Сокращение компьютерного парка.* Применение технологии виртуализации позволяет значительно сократить количество физических ЭВМ. В результате на закупку и замену оборудования тратится меньше средств. Сокращаются площади, необходимые для установки оборудования.

- *Сокращение штата ИТ-сотрудников.*
- *Простота в обслуживании.* Обслуживание и модернизация физических серверов требует значительного времени. Аналогичные работы на уровне виртуальных серверов выполняются с помощью административных приложений и обычно сводятся к нескольким щелчкам мыши или командам администратора.

- *Клонирование и резервирование.* Еще одним достоинством виртуализации является простота клонирования виртуальных машин. Развертывание новой инфраструктуры на новом месте сводится к простому копированию образов виртуальных машин на сервер нового офиса, конфигурированию сетевого оборудования и изменению настроек в прикладном ПО.

- *Создание индивидуальных виртуальных сред.* Применение систем управления виртуализацией позволяет динамически создавать индивидуальные виртуальные среды, в которых функционируют объекты ИКТ и которые доступны пользователю в дистанционном режиме.

Анализ перечисленных возможностей компонентов систем дистанционного обучения позволяет сделать вывод о том, что они достигли необходимого минимального уровня для построения и сопровождения систем непрерывного образования в области информационно-коммуникационных технологий. Дело за соответствующими методиками и образовательным контентом.

Существенные роли в системе непрерывного

образования играют методические разработки для дистанционного обучения специалистов в области ИКТ и соответствующий образовательный контент.

Развитие методик и технологий дистанционного обучения. Общество в целом и академическое сообщество практически преодолели известное предубеждение против дистанционных и электронных методов обучения [14]. Сейчас те или иные элементы дистанционного образования достаточно широко применяются в вузах и при традиционных формах получения образования. Мировые тенденции показывают, что будущее за такими моделями образовательного процесса, в которых активно используются различные средства, методы и технологии, в том числе и дистанционные.

Использование информационно-коммуникационных технологий позволяет:

- создать и развивать целостную систему ДО;
- развивать новые современные методы обучения;
- создавать и адаптировать образовательные материалы в качественно новых формах.

Большинство высших учебных заведений России в настоящее время разрабатывают и применяют средства и методы дистанционного обучения.

Новые нормативные документы [15], вступившие в силу относительно недавно, создают правовую основу для использования дистанционных технологий в образовательном процессе и переносят большую долю ответственности за качество подготовки специалистов на головной вуз.

Организации самостоятельно определяют объем аудиторной нагрузки и соотношение объема занятий, проводимых путем непосредственного взаимодействия педагогического работника с обучающимся, и учебных занятий с применением электронного обучения, дистанционных образовательных технологий.

При реализации образовательных программ с применением исключительно электронного обучения, дистанционных образовательных технологий теперь допускается *отсутствие аудиторных занятий* [15].

Создание современного контента. Практически все известные системы управления обучением позволяют создавать и поддерживать типовой *учебный материал*:

- текстовые (гипертекстовые) материалы;
- статические иллюстрации (или фотографии);

- учебные видеофрагменты.

Кроме того, системы управления обучением обеспечивают проведение контрольных мероприятий:

- тестирование знаний;
- выполнение практических заданий.

Выполнение практических заданий с помощью виртуальных практикумов. Узким звеном в системе дистанционного обучения специалистов в области ИКТ является выполнение практических заданий. Основной целью обучения специалистов в области ИКТ является приобретение базовых **навыков** работы с изучаемыми объектами (языками программирования, операционными системами, базами данных, информационными системами, системами защиты информации, элементами корпоративных сетей и т.п.). Получить такую возможность позволяют системы управления виртуализацией, позволяющие предоставить пользователю личную виртуальную среду, в которой функционирует реальный изучаемый объект. Доступ к такой виртуальной среде возможен с компьютеров в дисплейных классах учебных заведений, домашних ПК, с современных мобильных устройств.

Организация взаимодействия систем управления обучением с виртуальными практикумами. При создании целостной платформы для изучения дисциплин области ИКТ необходимо связать систему управления обучением с системой управления виртуализацией. Интегрирующий компонент должен предоставить пользователю возможность подключаться к индивидуальным виртуальным средам для выполнения практических заданий в пределах сеансов работы в системе управления обучением и предоставлять ему возможности управления своими средами на функциональном (понятийном) уровне, не требующем знания особенностей тех или иных систем управления виртуализацией.

Структура виртуального практикума. Виртуальный практикум – программно-аппаратный комплекс поддержки виртуальных сред (включает серверы с системой виртуализации и программным обеспечением, автоматизирующим работу с виртуальными машинами) [3].

Виртуальный практикум должен позволять автоматически по запросу пользователя создавать и уничтожать виртуальные среды, отправлять уведомления о выполнении задания преподавателю, а также предоставлять данные для соединения с виртуальной средой.

Виртуальной средой (ВС) будем называть индивидуальную систему-тренажер, выделяемую студенту (или слушателю курсов повышения квалификации) для выполнения практиче-

ского задания.

Можно выделить три типа ВС:

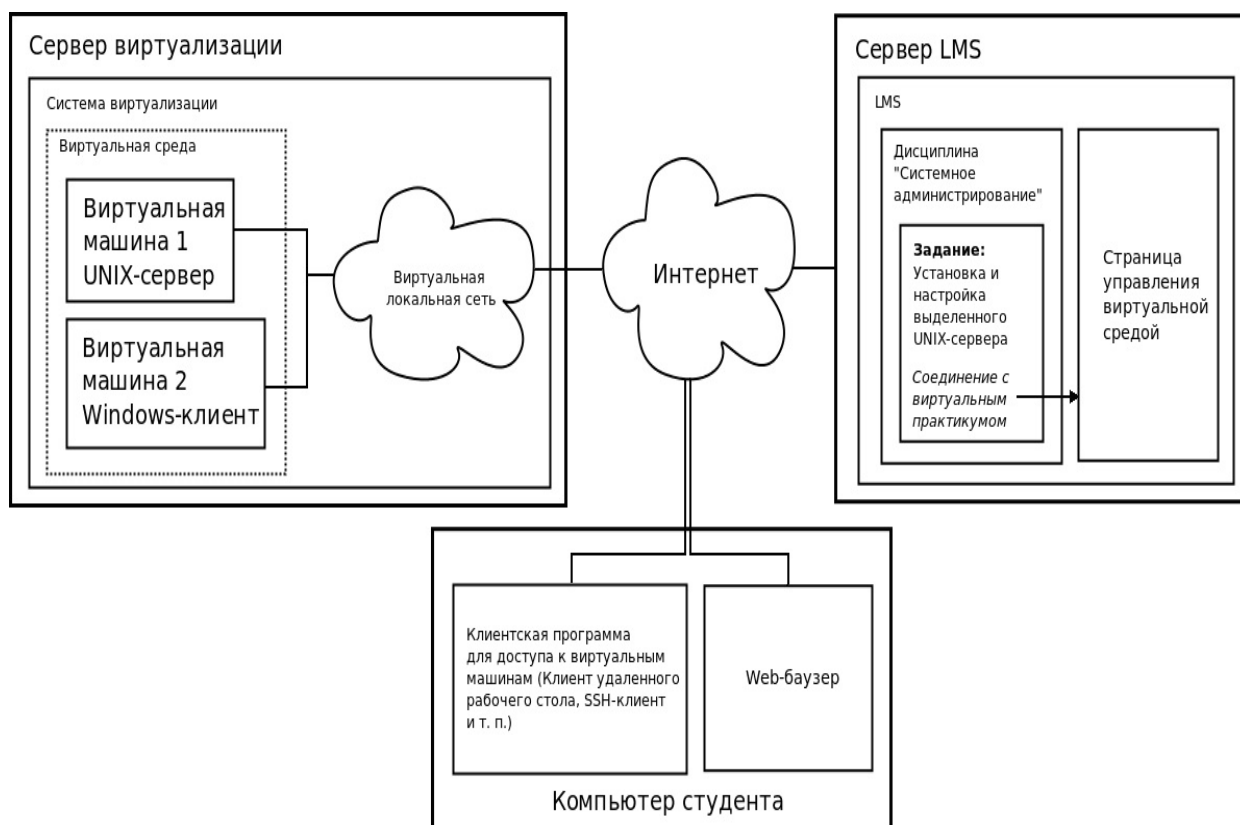
1) ВС на основе разделяемой виртуальной машины (VM). Это виртуальная машина, с которой могут работать несколько студентов. Студент работает в своем аккаунте независимо от других слушателей. Этот тип ВС может применяться для выполнения практических заданий, не связанных с модификацией системных объектов ОС (например, разработка и изучение прикладного ПО);

2) ВС на основе индивидуальной VM. Это виртуальная машина, предоставленная в единоличное распоряжение студента. Использование ВС этого типа удобно для выполнения заданий по системному программированию и системному администрированию, а также при работе с объектами, разделение которых между пользователями нецелесообразно (в силу лицензионной политики или технических трудностей);

3) Сложная ВС. Состоит из нескольких виртуальных машин, часть из которых может быть разделяемыми, а часть – индивидуальными. Студент имеет доступ к каждой виртуальной

машине в составе виртуальной среды. ВС этого типа может применяться для выполнения заданий по системному администрированию, сетевому программированию, изучению безопасности информационных систем, изучению компонентов корпоративных информационных систем.

Экспериментальные исследования. На рисунке приведена общая схема организации системы сопровождения практических заданий, выполняемых в виртуальных средах [4]. Используя веб-браузер, пользователь просматривает материалы дисциплины, размещенные в LMS, и управляет своей виртуальной средой. С помощью клиентской программы он получает доступ к виртуальной среде через Интернет и выполняет в ней практическое задание. На рисунке в качестве примера показана сложная ВС, состоящая из двух виртуальных машин и предназначенная для отработки умений по установке и настройке выделенного UNIX-сервера локальной сети. Эта же ВС может представлять собой фрагмент корпоративной сети.



Система сопровождения практических заданий, выполняемых в виртуальных средах

Использование виртуальных сред для выполнения практических заданий имеет следующие преимущества:

1) круглосуточный доступ к изучаемому объекту (компенсируется нехватка времени для практических занятий в дисплейном классе и

раскрывается потенциал самостоятельной работы студентов);

2) в случае необходимости преподаватель может получить доступ к виртуальной среде студента и проконсультировать его в затруднительной ситуации;

3) преподаватель имеет возможность дистанционно проверять работы студентов и отправлять им уведомление о доработке полученных результатов;

4) виртуальная среда предоставляет студенту и преподавателю стандартизованное окружение для выполнения задания (снимаются проблемы, часто возникающие при выполнении задания на одном компьютере и сдаче на другом);

5) в случае выхода виртуальной среды из строя (например, после неудачных экспериментов студента), она может быть удалена и заменена работоспособной виртуальной средой со стандартными исходными настройками (снимается проблема выхода из строя учебного сервера, рабочей станции в дисплейном классе или компьютера студента при выполнении на них заданий);

6) в одной виртуальной среде возможно объединение нескольких виртуальных машин, выполняющих разные роли (в случае использования физического оборудования задействовать несколько компьютеров для занятий каждого студента практически невозможно);

7) возможно выполнение заданий по установке и настройке той или иной ОС (на компьютерах дисплейного класса это либо невозможно, либо связано с постоянным ручным администрированием компьютеров – например, установленные ОС может понадобиться удалить, чтобы задание могла повторить следующая группа студентов);

8) возможна автоматическая организация терминального доступа студентов к одной виртуальной машине для выполнения в ней заданий, не подразумевающих оперирования с системными объектами (преподавателю не требуется заниматься администрированием учебного сервера);

9) виртуальные среды могут быть уничтожены автоматически после выставления преподавателем оценки за соответствующее задание или по запросу студента (преподавателю не требуется оперировать с программным обеспечением и документами в виртуальных средах вручную, как при подготовке к работе компьютеров в дисплейном классе).

В заключение коснёмся вариантов проведения обучения с применением дистанционной технологии.

В силу динамичности области ИКТ наибольший эффект повышение квалификации и профессиональная переподготовка могут дать при проведении их непосредственно на рабочем месте специалиста.

Обучение на рабочем месте [7]. Для многих

организаций, относящихся к разным сферам деятельности (торговля, сфера обслуживания, производство и др.), обучение на рабочем месте является основной формой обучения новых работников. При этом могут использоваться различные современные методы обучения. Перечислим основные из них: включение обучаемого в процесс деятельности другого человека, наблюдение за процессом работы, целенаправленная передача опыта, сопровождение процесса обучения, обсуждение опыта переноса полученных знаний в реальную практику (тьюторство).

В области информационных технологий не всегда удаётся построить полноценное обучение на рабочем месте в силу того, что функционирующие ИТ-системы весьма сложны и, как правило, не предназначены для проведения экспериментов с ними во время работы, а содержать вариант системы для обучения практически невозможно.

Обучение на рабочем месте для области ИКТ может означать договоренность (в плановом порядке) с руководством фирмы об использовании части рабочего времени сотрудника для изучения материала в системе дистанционного обучения обучающей организации и выполнения практических заданий на ресурсном сервере этой организации.

Таким образом, работники получают в свое распоряжение обучающие ресурсы прямо на рабочем месте или дома и могут планировать обучение по собственному графику, что значительно дешевле и эффективнее.

При этом возможно тьюторство и репетиторство со стороны более опытных сотрудников фирмы.

Продолжает также развиваться сочетание обучения на основе ИКТ с традиционными методами обучения на базе учебных заведений, что позволяет компенсировать отсутствие человеческого участия, возможное в обучении с применением только ИКТ.

Заключение. Современное состояние средств ВТ и специализированного ПО, которые применяются для построения систем дистанционного обучения позволяет создавать системы непрерывного и целостного дистанционного обучения в области информационно-коммуникационных технологий.

Основным звеном в таких системах являются виртуальные практикумы, позволяющие слушателю работать с реальными объектами через Интернет в любое время и в любом месте.

Наличие таких виртуальных практикумов существенным образом повышает качество обучения специалистов в области ИКТ и создаёт

надёжную базу для непрерывного образования в этой весьма динамичной отрасли.

Библиографический список

1. Фомин С.С., Кривошеев А.О., Сидоров С.И. Организация виртуального дистанционного практикума для системы повышения квалификации в области ИКТ // Ежеквартальный Научно-методический журнал «Информатизация образования и науки», № 4 (12) / 2011. С. 3-13.
2. Фомин С.С. Унифицированные виртуальные среды на базе свободного ПО для построения виртуальных практикумов по ИКТ дисциплинам // Труды конференции Телематика'2012. – СПб, 2012. С. 78-79.
3. Фомин С.С., Сидоров С.И. Повышение качества подготовки специалистов в области ИКТ // Ежемесячный Научно-практический журнал «Качество. Инновации. Образование», 2013. № 1 (92). С. 50—56
4. Фомин С.С., Сидоров С.И. Подготовка магистров по программе «Корпоративные информационные системы» // Ежемесячный Научно-практический журнал «Качество. Инновации. Образование», 2013. № 2 (93). С. 23—30
5. Реализация концепции обучения в течение всей жизни // <http://www.novsu.ru/dept/1108/i.122494?id=788901>.
6. Олейникова О.Н., Муравьева А.А., Аксёнова Н.М. Обучение в течение всей жизни как инструмент реализации Лиссабонской стратегии. – М.: РИО ТК им. Коняева, 2009 – 131 с.
7. Обучение на рабочем месте // Материал ООО «Скай Лайн Консалтинг». <http://www.mispnsk.ru/articles.html?id=447>.
8. Анализ современных технологий виртуализации // <http://habrahabr.ru/company/centosadmin/blog/212985/>.
9. Андреев А.А. Открытые образовательные ресурсы // Научно-педагогический журнал Министерства образования и науки РФ «Высшее образование в России». 2008. № 9. С. 114-116.
10. Стандарты обучения на основе технологий // ITU News Magazine, 2012, № 9. <https://itunews.itu.int/ru/Note.aspx?Note=3199>.
11. Coursera - проект в сфере массового онлайн-образования. <https://ru.wikipedia.org/wiki/Coursera>.
12. EdX is a massive open online course. // <http://en.wikipedia.org/wiki/EdX>.
13. 20 Most Popular Learning Management Systems. // <http://www.learndash.com/20-most-popular-learning-management-systems-infographic/>
14. Концепция развития системы дистанционного обучения (ДО) в Московском государственном университете имени М.В. Ломоносова // М. МГУ, 2007. С. 7.
15. Об утверждении порядка применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ. Приказ № 2 от 09.01.2014 г. // Министерство образования и науки РФ. Зарегистрирован в Минюсте России 04.01.2014 г. № 31823.

УДК 621.317.75:519.2

О.В. Миловзоров, А.В. Агузаров, Д.Ю. Тарабрин

МЕТОД СТРУКТУРИЗАЦИИ ДАННЫХ ДЛЯ АДАПТИВНЫХ СИСТЕМ НОРМИРОВАНИЯ МАШИНОСТРОИТЕЛЬНОГО ПРОИЗВОДСТВА

Разработаны метод и унифицированная модель данных, которые можно положить в основу адаптивной системы нормирования маршрутно-операционных технологических процессов. Выявлены основные принципы структуризации данных, принятых в общемашиностроительных справочниках. Предложена методика создания адаптивной системы нормирования, обеспечивающей работу предприятия на основе собственных нормативов времени, реализованных в виде СПП.

Ключевые слова: нормирование технологических процессов, адаптивная система, автоматизированное проектирование, САПР-система.

Введение. Определение обоснованных норм выполнения технологических операций является одной из важнейших задач конструкторско-технологической подготовки машиностроительного производства. От ее решения во многом зависит правильная оценка трудозатрат на изготовление отдельных деталей, узлов и машино-

строительного изделия в целом, а, следовательно, оценка себестоимости изделия и всех экономических показателей производства. *Цель работы* – на основе анализа контента общемашиностроительных справочников разработать унифицированную модель данных и предложить методику реализации адаптивных справочных систем

нормирования машиностроительного производства.

Теоретическая часть. Современные системы автоматизированного проектирования маршрутно-операционных технологических процессов (САПП-системы), имеющиеся на рынке САПП (TechCard, Вертикаль, T-Flex Технология, Спрут ТП, ADEM VX), решают эту задачу традиционным путем на основе специализированных подсистем [1, 2]. Как правило, такие подсистемы не являются гибкими. Информационной основой для них служат общемашиностроительные справочники, разработанные еще во время существования СССР в НИИ Труда. Нормы времени определялись путем хронометража типовых работ на различных видах станков, а также работ, выполняемых вручную. Значения норм времени заносились в многочисленные таблицы, из которых по соответствующим значениям параметров обработки технолог осуществлял выбор нужной нормы времени. Общее число только общероссийских справочников насчитывает несколько десятков. Эти нормы до сих пор остаются актуальными для многих машиностроительных предприятий.

В то же время в технологических процессах все шире применяются современные режущие инструменты, комплексные и комбинированные инструменты, использование которых позволяет существенным образом интенсифицировать процесс обработки. Все это, безусловно, требует корректировки устоявшихся нормативов времени, а в перспективе и разработки новых таблиц нормативов времени обработки. Существенным моментом является то, что нормативы могут быть специфичными, ориентированными на данное конкретное производство и предприятие. Такие нормативы обычно оформляются в виде СТП, причем форма представления данных, как правило, аналогична форме, принятой в общемашиностроительных нормативах.

В этих условиях унифицированная САПП-система с фиксированными видами обработки и нормативами времени уже не может отвечать требованиям потребителя. Возникает необходимость в создании таких подсистем нормирования, которые могли бы быть адаптированы под специфические требования конкретного производства.

Выявим основные принципы структуризации данных, принятых в общемашиностроительных справочниках, которые можно положить в основу унифицированной модели данных системы нормирования. Основной для нормирования

является карта, содержащая нормы времени для различных видов обработки в зависимости от различных факторов. Карты сгруппированы по виду нормы времени – неполное штучное время, вспомогательное время и подготовительно-заключительное время. Неполное штучное время – это, как правило, время выполнения одного рабочего прохода при обработке поверхности. Именно на его основе нормируется большая часть времени выполнения технологического обрабатываемого перехода, и, в конечном итоге, большая часть времени, затрачиваемого на обработку всей детали. Факторы, влияющие на эту составляющую штучного времени, можно разбить на две группы – группа, характеризующая технологию обработки, к которым можно отнести вид обработки (токарная, фрезерная и т.п.), обрабатываемый материал (сталь, чугун, сплавы), тип используемого инструмента, качество обрабатываемой поверхности, мощность станка и другие факторы. Ко второй группе относят параметры, характеризующие объем обработки, определяемые размерами обрабатываемой поверхности детали. Типовой вид карты нормирования, переведенной из бумажной формы в электронную в среде MS Excel, показан на рисунке 1. Среда MS Excel широко используется пользователями самой различной квалификации для оформления табличных данных. На ее основе в подразделениях конструкторско-технологической подготовки производства могут создаваться стандарты предприятия по нормированию технологических операций с учетом специфики производств и видов обработки, причем эту работу могут выполнять сотрудники, не имеющие квалификации программиста, но хорошо владеющие вопросами технологии производства. Таким образом, если информация о нормах времени будет оформлена как СТП в виде таблиц Excel, адаптивная система нормирования сможет использовать их в работе.

Рассмотрим информацию, содержащуюся в карте, на предмет составления унифицированной модели данных и выявления однотипных полей и атрибутов, на основе которых можно было бы сформировать оконный интерфейс системы нормирования. Будем представлять эти данные в виде набора структурированных множеств – кортежей.

Верхние поля карты всегда содержат *общую информацию* о технологической операции, которую можно использовать для распознавания и автоматического поиска нужной таблицы. Сюда относят:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V			
Горизонтально- и вертикально-фрезерные станки																								
Фрезы торцовые из стали Р6М5																								
Карты 9																								
1	Неполное штульное время																							
2	Фрезерование плоскостей, Rz 40, 11 квалитет																							
3	Сталь конструкционная углеродистая, $\sigma_b = 0,59 - 0,74$ ГПа																							
4	Единичное и мелкосерийное производство																							
5	Длина обрабатываемой поверхности l , мм, до																							
6	№ позиции	Мощность станка N_n , кВт, до	Диаметр фрезы D , мм, и число зубьев Z	Ширина обрабатываемой поверхности B , мм, до	Глубина резания t_f , мм, до	Время на рабочий ход, мин															Добавлять на каждые 100 мм свыше 950	п, об/мин	S_m , мм/мин	№, кВт, до
7						50	75	100	125	150	200	250	300	400	500	600	750	950	11,5	1,15				
8	1	4,8	80/10	48	3	1,4	1,65	1,95	2,4	2,65	3,1	3,65	4,05	5	6,5	7,5	9	11,5	1,15	190	96	1		
9	2	4,8	80/10	48	5-8	1,55	1,9	2,25	2,75	3,15	3,7	4,4	4,9	6,5	8	9,5	11	14	1,45	150	65	1,2-1,8		
10	3	4,8	100/10	60	3	1,55	1,9	2,25	2,75	3,15	3,7	4,4	4,9	6,5	8	9,5	11	14	1,45	150	75	0,9		
11	4	4,8	100/10	60	5-8	1,7	2,15	2,6	3,2	3,65	4,35	5	6	7,5	9,5	11,5	13,5	17	1,8	120	60	1,2-1,8		
12	5	4,8	125/14	75	3	1,4	1,7	2	2,45	2,7	3,1	3,7	4,1	5	6,5	7,5	9	11,5	1,15	120	96	1,4		
13	6	4,8	125/14	75	5-8	1,5	1,95	2,3	2,85	3,2	3,75	4,5	4,95	6,5	8	9,5	11	14	1,45	96	75	1,8-2,8		
14	7	10	80/10	48	3	1,15	1,35	1,55	1,85	2,05	2,3	2,7	2,95	3,7	4,4	5	6	7,5	0,72	190	150	1,3		
15	8	10	80/10	48	5-8	1,25	1,5	1,7	2,1	2,3	2,65	3,1	3,45	4,35	5	6	7,5	9,5	0,9	150	120	1,7-2,7		
16	9	10	100/10	60	3	1,25	1,5	1,7	2,1	2,3	2,65	3,1	3,45	4,35	5	6	7,5	9,5	0,9	150	120	1,3		
17	10	10	100/10	60	5-8	1,35	1,65	1,95	2,4	2,65	3,1	3,65	4,05	5	6,5	7,5	9	11	1,15	120	96	1,7-2,7		
18	11	10	125/14	75	3	1,4	1,7	2	2,45	2,7	3,15	3,7	4,1	5	6,5	7,5	9	11,5	1,15	120	96	1,4		
19	12	10	125/14	75	5-8	1,6	1,95	2,3	2,85	3,2	3,75	4,5	4,95	6,5	8	9,5	11	14	1,45	96	75	1,8-2,8		
20	13	10	160/16	112	3	1,8	2,15	2,55	3,05	3,4	3,95	4,65	5	6,5	8	9,5	11,5	14	1,45	96	75	1,7		
21	14	10	160/16	112	5-8	2,05	2,5	2,95	3,55	4	4,7	5,5	6	8	10	11,5	14	17,5	1,8	75	60	2,2-3,5		
22	15	10	200/20	140	3	2,4	2,95	3,5	4,25	4,8	5,5	6,5	7,5	9,5	12	14,5	17	21,5	2,25	60	48	1,2		
23	16	10	200/20	140	5-8	2,8	3,5	4,25	5	6	7	8	9	12	15	18	21,5	26,5	2,85	48	38	1,8-2,9		
24	17	15	100/10	60	3	1,25	1,5	1,7	2,1	2,3	2,65	2,95	3,45	4,35	5	6	7,5	9,5	0,9	150	120	1,3		
25	18	15	100/10	60	5-12	1,4	1,65	1,95	2,4	2,7	3,1	3,65	4,05	5	6,5	7,5	9	11,5	1,15	120	96	1,7-3,8		
26	19	15	125/14	75	3	1,4	1,65	1,95	2,4	2,7	3,1	3,65	4,05	5	6,5	7,5	9	11,5	1,15	120	96	1,4		
27	20	15	125/14	75	5-12	1,6	1,95	2,35	2,85	3,2	3,75	4,5	4,95	6,5	8	9,5	12	14	1,45	96	75	1,8-4,1		
28	21	15	160/16	112	3	1,8	2,15	2,55	3,05	3,45	3,95	4,7	5	6,5	8	9,5	12	14	1,45	96	75	1,7		
29	22	15	160/16	112	5-12	2,05	2,5	2,95	3,6	4,05	4,7	5,5	6	8	10	12	14	17,5	1,8	75	60	2,2-5,0		
30	23	15	200/20	140	3	2,1	2,55	3	3,6	4,05	4,75	5,5	6	8	10	11,5	14	17,5	1,8	75	60	1,8		
31	24	15	200/20	140	5-12	2,4	2,95	3,55	4,25	4,85	5,5	7	7,5	9,5	12	14,5	17	21,5	2,25	60	48	2,3-5,4		

Рисунок 1 – Пример типовой карты норм неполного штульного времени для операции фрезерования плоскостей, выполняемой на горизонтально- и вертикально-фрезерных станках, представленная в среде MS Excel

- тип норматива (штучное или неполное штучное время);
- вид обработки и обрабатываемой поверхности с указанием качества;
- обрабатываемый материал;
- серийность производства;
- используемое металлорежущее оборудование;
- используемый металлорежущий инструмент;
- номер карты и листа (если карта занимает несколько страниц).

Данные всех этих полей (за исключением номера листа) целесообразно выдавать в описании нормируемой технологической операции. Как правило, эти поля занимают верхние строки таблицы в представлении MS Excel. Данные поля обозначим, как A_i , $i=1..I$, где I – общее количество полей, отводимых под общую информацию об операции. Как правило, общее количество полей, содержащих общую информацию (и соответствующее количество ячеек Excel), не превышает 8. В результате общая информация о технологической операции будет описана кортежем $A = \{A_1, A_2, A_i, \dots, A_I\}$.

После полей, содержащих общие сведения о технологической операции, следует собственно тело информационной таблицы. Среди информативных атрибутов первыми следуют атрибуты, конкретизирующие условия обработки, а также геометрические параметры обрабатываемой поверхности. Таковыми могут быть:

- мощность станка;
- диаметр и число зубьев режущего инструмента;
- базовый геометрический параметр, характеризующий условия обработки (диаметр, ширина, высота обрабатываемой поверхности, модуль зубчатого колеса и т.д.);
- глубина резания.

Обозначим эти атрибуты через C_k , $k=1..K$, где K – общее количество информативных атрибутов, конкретизирующих условия обработки. Тогда информация, конкретизирующая условия обработки, будет описана кортежем $C = \{C_1, C_2, \dots, C_K\}$.

Затем следует целая группа атрибутов, обеспечивающих конкретизацию геометрии обрабатываемой поверхности, по сути, определяющих объем обработки. Как правило, это длина обрабатываемой поверхности, реже – ширина или глубина обработки. Данные этих атрибутов определяют диапазон, в котором может изменяться параметр, для которого при укрупненном

нормировании задается фиксированная норма времени. Общее количество таких атрибутов может колебаться в достаточно широких пределах. Обозначим эти атрибуты через D_l , $l=1..L$, где L – общее число атрибутов, определяющих объем обработки. Соответствующий кортеж примет вид $D = \{D_1, D_2, \dots, D_L\}$.

Дополнительно к этой группе атрибутов необходимо ввести атрибут, обеспечивающий аддитивную добавку к выбранной норме времени DD . При этом возможны два варианта расчета. Первый вариант – если объем обработки превышает диапазон установленных картой значений.

Наименование соответствующего столбца в бумажном варианте карты нормирования обычно имеет вид «Добавлять на каждые ... мм свыше ...» (рисунок 2, а, б).

		Режим				
950	Добавлять на каждые 100мм свыше 950	п. об/мин	180	На каждые последующие 25 мм длины добавлять, мин	300	При обработке закрытого паза добавлять, мин
			9,1	1,2		
			10,9	1,4		
			12,9	1,6		
			14,9	1,8		
11	1,1	190				
14	1,45	150				
14	1,45	150				
17	1,8	120				

а

б

в

Рисунок 2 – Примеры столбцов в картах нормативов, определяющих аддитивную добавку

В этом случае атрибут DD определяет аддитивную добавку к норме времени, определяемой атрибутом D_L , т.е. нормы, установленной для максимальной границы диапазона длин обработки. Значение добавки рассчитывается исходя из нормы времени по обработке добавочных 100мм (либо какой-то другой нормативной величины) поверхности. Второй вариант расчета – если к любой из норм времени D_l , определяемой рядом значений длин обработки, необходимо ввести фиксированную аддитивную добавку. Такая добавка может быть связана с условиями обработки (например, фрезерование открытого или закрытого паза) (рисунок 2, в). Аддитивную составляющую нормы времени опишем кортежем $DD = \{TDD, PRR, NV\}$, где TDD – наименование атрибута DD (текстовая строка, описывающая действия технолога – нормировщика (рисунок 2); PRR – признак расчета: 1 – если добавка к норме времени рассчитывается исходя из длины обработки, превышающей максимальную границу диапазона длин, имеющихся в карте нормирования; 0 – если добавка увеличивает норму по карте нормирования на фиксированную величину.

ну; NV – добавочная длина обработки, которой соответствует указанная в атрибуте добавочная норма времени.

Последними атрибутами в картах нормирования следуют нормативы, определяющие рекомендованные режимы обработки – частоту вращения шпинделя, подачу, скорость резания, мощность резания, обеспечивающие данный норматив. Обозначив через R_m m -й атрибут режима обработки, получим кортеж $R = \{R_1, R_2, R_m, \dots, R_M\}$, описывающий информацию о режимах обработки.

Кроме информативных данных, определяющих расчет нормы времени, в картах также есть поля, которые содержат текстовую информацию и являются обобщенными заголовками, например, «Длина обрабатываемой поверхности», «Режимы резания» либо дополнительную инструктивную информацию для технолога-нормировщика – какое именно время является нормой – время одного прохода при обработке на станках, обработки одного зуба при обработке зубчатых колес и т.д. Обозначив данные поля через TD для атрибутов D , TR для атрибутов R и TV для инструктивного поля, уточняющего вид нормы времени, получим кортеж заголовков $T = \{TD, TR, TV\}$.

При работе технолога-нормировщика важно обеспечить высокую эргономичность работы с системой нормирования, поскольку нормирование заключается в многократном повторении рутинных однообразных действий по вводу исходных для расчета данных, приводящих к быстрому утомлению. В качестве диалогового окна целесообразно предложить окно, сочетающее в себе текстовые поясняющие поля, выпадающие списки, поля для ручного ввода недостающих данных (при необходимости), поля для вывода результатов расчета и некоторое количество управляющих элементов. На рисунке 3 показано типовое диалоговое окно, которое можно было бы предложить для нормирования основного времени.

Можно установить однозначную взаимосвязь между кортежами данных и элементами управления данного окна. Так, верхний контейнер «Описание» содержит общие сведения о выполняемой технологической операции, которые, по сути, содержатся в кортеже $A(I)$. Контейнер «Входные данные», где осуществляется ввод в систему технологических параметров, определяющих расчет нормы времени, можно составить из элементов типа "выпадающих список", в которых заголовки и наполняющая информация формируются на основе данных атрибутов C и

D . В элементах контейнера «Выходные данные» осуществляется вывод результатов расчета – норм времени и значений параметров режимов резания.

Рисунок 3 – Пример диалогового окна для нормирования основного времени

Выпадающие списки для вывода данных атрибутов C и D располагаются «справа-налево» и «сверху-вниз», при этом необходимо выполнять ассоциативную связь между ними. Архитектура списков иерархическая. Первый (верхний левый) список определяет состав элементов для второго списка, расположенного правее, второй – для третьего и т.д. Элемент списка, выбранный пользователем из правого столбца таблицы, определяет, какие элементы из столбца, расположенного правее, должны отображаться в следующем по порядку выпадающем списке.

Сформулируем принципы связи выпадающих списков. Обозначим через C_{vk} элемент атрибута C_k , выбранный на предыдущем шаге пользователем или системой (при автоматическом нормировании) в качестве действующего для поиска нормы времени. Тогда условие формирования массива элементов параметров для следующего выпадающего списка атрибута C_{k+1} , выдаваемого в диалоговом окне на последующем шаге можно представить, как

$$\forall C_{pk+1} \in M_{C_{k+1}} (\exists (C_{pk} = C_{vk}) \vee (C_{pk+1} \neq C_{jk+1})) , \\ p = 1 \dots P \quad j = 1 \dots p - 1$$

где P – количество информативных строк в карте;

$M_{C_{k+1}}$ – массив элементов для выпадающего списка атрибута C_{k+1} .

Т.е. в выпадающем списке должны выво-

даться лишь те элементы, которые располагаются в строках, в которых имеется элемент C_{vk} .

Последним выбранным элементом становится элемент атрибута C_{pK} , однозначно определяющий конкретную строку таблицы нормирования p , из которой по атрибуту D выбирается норма времени и комплект параметров режимов резания.

Таким образом, модель карты нормирования основного времени можно представить в виде набора структурированных множеств - кортежей:

$$K_n = \{A(i)_{i=1...L}; C(p,k)_{p=0...P, k=1...K}; D(p,l)_{p=0...P, l=1...L}; DD; R(p,m)_{p=0...P, m=1...M}; T\}$$

и правил формирования выпадающих списков.

Системе нормирования должна передаваться информация об адресах конкретных ячеек страницы книги MS Excel – карты справочника норм времени, хранящих те или иные данные и в соответствии с этим элементами кортежей становятся адреса этих ячеек MS Excel. На основе такого описания система нормирования может сгенерировать карту в виде внутреннего файла, уже не привязанного с MS Excel. Весь комплект карт будет образовывать внутреннюю базу данных.

Информация, содержащаяся в кортеже A , может быть использована для выстраивания иерархии в базе данных, и в процессе работы системы обеспечивать поиск нужной карты нормирования.

Рассмотрим, каким образом наиболее целесообразно выстраивать порядок следования элементов кортежа A . Основным структурным элементом маршрутно-операционной технологии является технологическая операция, наименование которой, как правило, определяется типом станка, соответственно элемент A_1 должен соответствовать типу оборудования (для станочной обработки) или типу операции. Элемент A_2 должен определять серийность производства:

- единичное и мелкосерийное;
- среднесерийное;
- крупносерийное и массовое.

Элемент A_3 должен определять тип нормы времени: неполное штучное, штучное время, или основное время на обработку некоторого фиксированного расчетного размера поверхности (для среднесерийного производства). Кроме того, справочниками предусмотрены нормы для подготовительно-заключительного времени и вспомогательного времени.

Элементу A_4 должно соответствовать описание технологического преобразования – опера-

ции или перехода, т.е. описание обработки с указанием получаемого качества поверхности (для слесарно-сборочных технологических процессов – описание выполняемого действия).

Элемент A_5 описывает тип обрабатываемого материала. В традиционных системах нормирования базовые нормы времени определяют для трех типов материалов:

- чугун серый;
- сталь конструкционная углеродистая;
- медные сплавы и алюминиевые.

В зависимости от категории твердости материала нормы времени корректируются с помощью поправочных коэффициентов.

Элементы A_6, A_7 – поля с описанием режущего инструмента (в картах для среднесерийного производства информация об инструменте может содержаться более чем в одном поле).

Элемент A_8 – поле с номером карты.

Таким образом, каждая карта будет описана кортежем A , информация в котором позволит системе осуществлять поиск нужной карты и формировать дерево расчетов.

Для окончательного формирования описания необходимы данные о размерах контента информационной части таблицы нормирования в виде номеров первой и последней строк p_1 и p_p .

После того как оператором (или системой) будет выбран нужный расчет, на основе информации в кортежах A, C и D, R и T может осуществляться формирование диалогового окна, включающего всю описательную часть и логическую часть в виде выпадающих списков и текстовых полей и интерактивный расчет нормы времени.

Обозначим адрес ячейки Excel с соответствующей информацией в виде $TC_{\text{индекс}}$, где индекс – символ элемента кортежа (A, C, D или R). Тогда описание карты нормирования основного времени будет иметь вид:

$$\left\{ \begin{array}{l} A(I) = \{TC_{A1}, \dots, TC_{AI}\} \text{ (кортеж ячеек общих сведений об операции);} \\ C(K) = \{TC_{C1}, \dots, TC_{CK}\} \text{ (кортеж наименований атрибутов C);} \\ D(L) = \{TC_{D1}, \dots, TC_{DL}\} \text{ (кортеж наименований атрибутов D);} \\ DD = \{TC_{TDD}, PRR, NV\}; \\ R(M) = \{TC_{T1}, \dots, TC_{RM}\} \text{ (кортеж наименований атрибутов R);} \\ T = \{TD, TR, TV\} = \{TC_{TD}, TC_{TR}, TC_{TV}\}; \\ p_1 = NR_{p1}; p_p = NR_{pp} \text{ (номера первой и последней строк).} \end{array} \right.$$

Например, для карты на рисунке 1, ее описание представится в виде:

$$\left\{ \begin{array}{l} A(7) = \{Q1, A4, A1, A2, A3, Q2, V4\}; \\ C(4) = \{B5, C5, D5, E5\}; \\ D(13) = \{F6, G6, H6, I6, J6, K6, L6, M6, \\ N6, O6, P6, Q6, R6\}; \\ DD = \{S6, 1, 100\}; \\ R(3) = \{T6, U6, V6\}; \\ T = \{TD, TR, TV\} = \{F5, T5, F7\}; \\ p_1 = 8; p_p = 31. \end{array} \right.$$

Рассмотренный подход можно распространить и для формализации описания систем поправочных коэффициентов, примечаний, карт вспомогательного и подготовительно-заключительного времени. При этом должны быть учтены особенности формирования этих данных.

В целом предлагаемые метод структуризации справочных данных и модель описания данных справочников нормирования позволяют создать адаптивную систему нормирования, которая может быть составляющей подсистемой таких САПР-систем, как Вертикаль, Т-Flex Технология и обеспечивать работу предприятия на основе собственных нормативов времени, реализованных в виде СТП в среде MS Excel. Такая система будет состоять из нескольких взаимодействующих программных модулей:

- генератора внутренней базы данных;
- модуля импорта дерева техпроцесса из

САПР-системы;

- набора модулей, обеспечивающих нормирование основного, вспомогательного и подготовительно-заключительного времени;
- модуля расчета поправочных коэффициентов;
- модуля экспорта результатов расчета норм времени в дерево техпроцесса САПР-системы.

Заключение. В статье предложен метод структуризации данных и информационная модель данных для адаптивных систем автоматизированного нормирования маршрутно-операционных технологических процессов. Предложена структура системы, основанная на данной информационной модели.

Библиографический список

1. Миловзоров О.В., Давыдов А.П., Ухова Т.И., Павлов А.В., Пахомов А.В. Комплексы мини-САПР нормирования как средство создания САПР, полностью адаптированной к конкретному производству // САПР и графика №10, 2007//
URL: <http://sapr.ru/article.aspx?id=18693&iid=847>
2. Миловзоров О.В., Давыдов А.П., Ухова Т.И., Павлов А.В., Пахомов А.В., Шлыков Е.С. Новые возможности САПР-системы Т-FLEX Технология по нормированию технологических процессов // САПР и графика №3, 2008//
URL: <http://sapr.ru/article.aspx?id=18833&iid=874>