УДК 519.876.5

В. М. Дубовой, д. т. н., проф.; Е. Д. Никитенко; О. В. Глонь, к.т. н., доц.

ЭКВИВАЛЕНТНОСТЬ НЕОПРЕДЕЛЕННЫХ АЛГОРИТМОВ

Рассмотрены свойства алгоритмических моделей, в частности, эквивалентность неопределенных алгоритмов. Предложен подход для определения степени эквивалентности алгоритмов в условиях неопределенности с помощью операторного метода.

Ключевые слова: *алгоритмические модели, эквивалентность неопределенных алгоритмов, условия неопределенности, степень эквивалентности.*

Повышение эффективности информационных систем (ИС) становится все более актуальной проблемой в связи с глобальной информатизацией всех сторон жизни общества, усложнением и увеличением масштабов ИС. Повышение эффективности предусматривает оптимизацию структуры ИС, распределение задач по подсистемам ИС и т.п. Одним из перспективных направлений решения этой проблемы есть оптимизация алгоритмической модели ИС с последующей ее реализацией программно-аппаратными средствами [1, 2].

Оптимизация алгоритмической модели (АМ) осуществляется с помощью системы эквивалентных преобразований [3]. Эквивалентным называют преобразование, в результате которого получают эквивалентную алгоритмическую модель. Две алгоритмические модели называют функционально эквивалентными, если при одинаковых входных данных они дают одинаковые результаты.

Значительные результаты в исследовании эквивалентности алгоритмов принадлежат А. А. Ляпунову, который ввел понятие схем программ [4]. На базе стандартных схем алгоритмов введены основные понятия и свойства, связанные с алгоритмическими моделями, главное из которых – отношение функциональной эквивалентности алгоритмических моделей.

Идеи Ляпунова были развитые в конце 50-х и в 60-е гг. А. П. Ершовым, Н. А. Криницким, Л.А. Калужниным, Р. И. Подловченко и Ю. И. Яновым, который в [5] формализовал понятие схемы программы, определил отношение эквивалентности схем и исследовал проблему эквивалентности для класса схем, которые получили со временем название схем Янова. Н. А. Криницкий [6] исследовал проблему эквивалентности и эквивалентных преобразований стандартных схем, причем для подкласса схем без циклов (т. е. схем, граф которых не содержит контуров), нашёл алгоритм распознавания эквивалентности и построена полная система преобразований, которая разрешает любую пару эквивалентных схем автоматически превратить одна в одну. Графовая форма схем была предложена Л. А. Калужниным [7].

Алгоритмические модели информационных систем большей частью рассматривалось в детерминированных условиях [4 – 8]. Но в большинстве практических задач функционирование ИС происходит в условиях неопределенности (УН) входных данных, причем степень и происхождение неопределенности может существенным образом отличаться, в частности ее причинами могут быть: недостаточное знание предметной области, нехватка точной информации о значении данных, неопределенность целей и т. п.

Учет неопределенности открывает новые возможности проектирования и оптимизации ИС на основе алгоритмических моделей.

Пусть есть две информационных системы $ИС_1$ и $ИС_2$, причем $ИС_2$ есть менее затратным вариантом $ИС_1$ того же назначения $C_1 > C_2$, где C_1, C_2 – затраты соответствующих информационных систем $ИС_1$ и $ИС_2$. Системам отвечают алгоритмические модели $АМ_1$ и $АМ_2$, которые определяют преобразование входных данных X на результат функционирования Y. Поскольку модели систем отличаются, то и результаты функционирования будут разными, значит:

$$Y_1 = AM_1(X) \text{ и } Y_2 = AM_2(X).$$

Охарактеризуем результаты Y_1 и Y_2 функциями неопределенности $\beta_1(Y_1)$ и $\beta_2(Y_2)$ соответственно, которые показаны на рис. 1. Достоверность совпадения результатов в условиях неопределенности

$$B(Y_1 = Y_2) = \int\limits_{\Omega_{Y_1} \cap \Omega_{Y_2}} \beta_1(Y)\beta_1(Y)dY,$$

где Ω_{Y_1} и Ω_{Y_2} – соответственно области значений результатов Y_1 и Y_2.

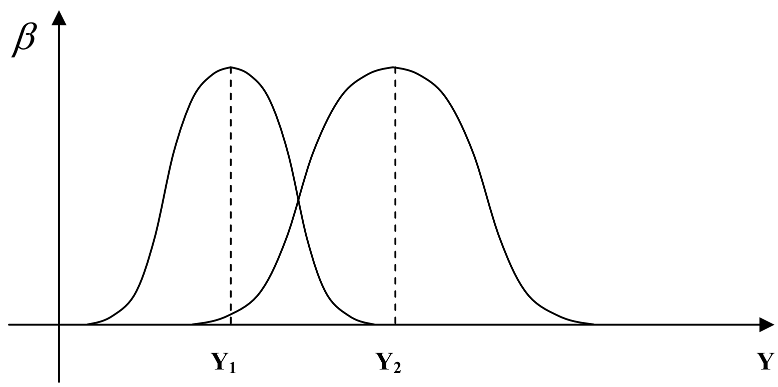

Рис. 1. Функции неопределенности результатов Y_1 и Y_2

Исходя из определения эквивалентности алгоритмических моделей, можно сказать, что системы ИС$_1$ и ИС$_2$ в условиях неопределенности являются эквивалентными с достоверностью $B(Y_1 = Y_2)$. Тогда выбор эффективного варианта ИС сводится к оцениванию стоимости риска $Q = (1 - B)/(C_1 - C_2)$.

Таким образом, для проектирования информационных систем актуальной есть задача исследования эквивалентности алгоритмических моделей в условиях неопределенности.

Задача определения эквивалентности алгоритмических моделей в условиях неопределенности не имеет общепризнанных подходов к ее решению. Усовершенствование методов анализа эквивалентности алгоритмических моделей с целью учета в них условий неопределенности функционирования систем является *задачей* данной работы.

Класс стандартных схем характеризуется базисом класса *B* и структурой схемы. Базис класса фиксирует символы, из которых строятся схемы, определяет их роль (переменные, функциональные символы и т. п.), задает вид выражений и операторов схемы [8]. Фиксация конкретной интерпретации превращает стандартную схему в конкретную алгоритмическую модель. Интерпретацией базиса *B* в области интерпретации *D* называется функция *I*, которая сопоставляет каждому элементу (переменным, константам, функциональным символам, предикатам) из базиса *B*, некоторые всюду определенные функции и элементы из области интерпретации *D*. Пара (*S, I*), где *S* – схема в базисе *B*, а *I* – интерпретация этого базиса, называется интерпретированной стандартной схемой алгоритма или алгоритмической моделью.

В [8] введено отношение эквивалентности для стандартных схем алгоритмов в одном базисе. Если схемы S_1 и S_2 построены в двух разных базисах B_1 и B_2, то можно их "привести к одному базису", в качестве которого взять объединение базисов B_1 и B_2. Стандартные схемы S_1 и S_2 в базисе *B* функционально эквивалентные ($S_1 \sim S_2$), если для любой интерпретации *I* базиса *B* программы (S_1, *I*) и (S_2, *I*) или обе зацикливаются, или обе останавливаются с одинаковым результатом, т. е. $val(S_1, I) \approx val(S_S, I)$. Эквивалентными преобразованиями алгоритмической модели будем называть такую последовательность операций над моделью, которая не изменяет содержание результатов работы системы.

Введенные понятия разрешают перейти к определению эквивалентности алгоритмических моделей в условиях неопределенности.

В условиях неопределенности это понятие имеет размытые границы. Учитывая ограниченную достоверность результата работы алгоритма, получаемого в условиях неопределенности, можно говорить лишь об эквивалентности алгоритмов с заданной достоверностью или о степени эквивалентности алгоритмических моделей. Рассмотрим возможный подход к оценке этой достоверности.

Неопределенность может описываться разными способами. Воспользуемся функциональным способом описания. При функциональном способе неопределенность стохастичного типа описывается распределениями вероятности, а нечеткого типа – функциями принадлежности. Метод обобщающих функций [9] учитывает неопределенность разного типа. Под обобщающей функцией понимают положительно определенную функцию на промежутке возможных значений аргумента, которая обозначается $\beta(x)$ и характеризует возможность π или вероятность P принятие аргументом значения из определенного интервала $[x_1, x_2]$, $x_1 \in B$, $x_2 \in B$, по правилам:

$$p = \frac{\int_{x_1}^{x_2} d[\beta(x)]}{\int_B d[\beta(x)]}, \qquad \pi = \frac{\int_{x_1}^{x_2} d[\beta(x)]}{\max\limits_B \int_{[x_{i-1}, x_i]} d[\beta(x)]}, \tag{1}$$

где $x_{i-1}, x_i \subset B$, $i = \overline{1, n}$, n – количество интервалов разбивки B.

Для обобщающей функции определены также правила обобщения математических операций. Все операции разделены на три группы: нелинейные унарные, нелинейные бинарные, интегро-дифференциальные. За основу определения этих операций принят операторный метод преобразования, который использует интегральные операторы вида:

$$\beta_Y(y) = \int_{-\infty}^{+\infty} \overset{n}{\cdots} \int_{-\infty}^{+\infty} \beta_X(\bar{x}) \psi(\bar{x}, y, F, Q) \mathrm{d}\bar{x}, \tag{2}$$

где ψ – ядро оператора, F и Q – характеристики операции, которая выполняется; n –кратность интегрирования, которая зависит от размерности вектора $\bar{x}$ и характеристик операции F и Q.

Функции неопределенности результатов работы двух алгоритмов, степень эквивалентности которых исследуется, может быть полученная с помощью операторного метода. Для этого алгоритмы подаются в алгебраической форме со следующим преобразованием на операторную форму записи.

Для преобразования детерминированной модели (R-форма) операторов $\tilde{u}$ в обобщенную (G-форма) исходную модель записывают в виде ряда символов, которые образуют определенную математическую формулу в системе R. При записи модели используются неопределенные сменные, знаки операций {+, -, *, /}, знаки элементарных функций, обозначения интегро-дифференциального (динамического) преобразования в форме интеграла Дюамеля $I(x*g)$, где x – исходная функция, g – ядро преобразования (импульсная переходная функция динамического преобразования), разделители.

Примеры типичных записей приведены в табл. 1.

Таблица 1

Примеры типичных записей алгоритмической модели в алгебраической и операторной формах

Запись на алгоритмическом языке	Алгебраическая форма (R-форма)	Комментарий	Операторная форма (G-форма)	Комментарий
Проверка условия и разветвление if (a) then $n1$ else $n2$	<<$E_1(n_1)$ $w2$ $(i>a)\_\vert 1$ $E_2(n_2)$>>	a – логическая перем.; $n1 \in N$ – номер оператора алгоритма, к которому осуществляется переход при истинном a, $n2 \in N$ – при ложном; N – нумерованное множество операторов алгоритма	Операторная модель будет иметь вид двухкомпонентного вектора $b(g)=\begin{Bmatrix} b(a1)\cdot(n1), \\ b(a2)\cdot b(n2) \end{Bmatrix}$	$b(a)=k\cdot\delta[a-a1]+$ $+(k-1)\cdot\delta[a-a2],$ k – вероятность истинного значения условия a
Вычисление функции $p_2=f(p_1)$	$I(f(p_1)\,/\,p_2)$	p_1 – исходные данные; p_2 – результат вычисления ; f – формула вычисления	$b(p2)=F\,(1,f)[b(p1)]$	$F\,(1,f)$ – оператор n-го порядка; нелинейная бинарная операция; $b(p_1),b(p_2)$ – обобщающие функции
Инициализация констант # *define* p_2, p_1	$C(p_1/p_2)$	p_1 – значение константы; p_2 – имя константы	$b(p_1)=\delta[p_1]$ $b(p_2)=F(1,1)[b(p_1)]$	$\delta[p2-p1]=$ $=\left\lvert\begin{array}{l} 0, при\ p1\neq p_2 \\ \infty, при\ p_1=p_2 \end{array}\right.$ $\int\limits_{-\infty}^{+\infty}\delta[p_2-p_1]dp_2=1$
Измерение *import* (&p_1) $p_2=f(p_1)$ $p_3=\varepsilon$	$Im(p_1,\varepsilon\,/p_2,p_3)$	p_1 – измеряемая величина; p_2 – результат измерения; p_3 – погрешность измерения	$b(p_2)=F(2,+)[b(p_1),b(p_2)]$	$b(p_3)=\frac{1}{2\pi p_3}e^{-\frac{(p_3)^2}{2\varepsilon^2}}$ $b(p_2)=\delta[p_1]$ Допускается нормальное распределение погрешности измерений
Экспертные данные *scan* (&p_1, &p_2); $p_3=(p_1+p_2)/2$; $p_4=(p_1-p_2)/6$	$Ex(p_1,p_2\,/\,p_3,\,p_4)$	p_1, p_2 – левая и правая границы оценки эксперта ;	$b(p_3)=F(1,N)[e(p)]$	$e(p)=\left\lvert\begin{array}{l} 1, -0.5\le p\le +0.5 \\ 0, \lvert p\rvert>0.5 \end{array}\right.$ $N=p\cdot(p_2-p_1)+\frac{p_1+p_2}{2}$
Задержка delay (τ)	$I\,(p1(t-\tau)\,/\,p2(t))$	τ – время задержки	$b(p_2)=F(n,g_\tau)[b(p_1)]$	g_τ – импульсная переходная функция звена задержки $g_\tau=\frac{1}{2\pi}\int\limits_0^\infty e^{-p\tau}e^{pt}dp$
Начало (конец) цикла { }	A(B), A(E)			

Преобразование алгоритмической модели формализовано в виде алгебраической системы

[3]. Эквивалентные преобразования алгоритмической модели осуществляются на основе свойств:

– *paste(B,n1,n2) cut(n1,n2) ≡ 1;*
– *cut(n1,n2) paste(B,n1,n2) ≡ 1;*
– *paste(B1,n1,n2) paste(B2,n3,n4) ≡ paste(B2,n3,n4) paste(B1,n1,n2), если (n1,n2) ∩ (n3,n4) = ∅;*
– *cut(n1,n2) cut(n3,n4) ≡ cut(n3,n4) cut(n1,n2), если (n1,n2) ∩ (n3,n4) = ∅;*
– *En1(op,X,Y) En2(op-1,Y,X) ≡ 1.*

В [9] определены также отношения сравнения неопределенных данных в системе *G*:

Определение 1. Неопределенные данные *x, y* считаются равными $X = Y$ если $\beta_X = \beta_Y$.

Определение 2. Для неопределенных данных $X \gtrsim Y$, если $Z = X - Y$ и

$$\int_{0}^{+\infty} \beta_Z dz > \int_{-\infty}^{0} \beta_Z dz .$$

Для оценивания степени эквивалентности алгоритмов в условиях неопределенности введем понятие степени равенства.

Определение 3. Степенью равенства неопределенных данных *x* и *y*, которые характеризуются функциями неопределенности β_x и β_y, соответственно будем называть величину

$$d = \int_{-\infty}^{+\infty} \beta_{xy}(x = \xi, y = \xi) d\xi = \int_{-\infty}^{+\infty} \beta_x(\xi) \cdot \beta_y(\xi) d\xi . \tag{4}$$

Очевидно, *d=1* при $x = y$ по *определению 1.*

Результат работы алгоритма можно определить на основе таких типов представления:

1) числовое значение, которое принадлежит некоторому непрерывному интервалу возможных значений;
2) числовое значение, которое принадлежит некоторому конечному дискретному множеству возможных значений;
3) действие, которое осуществляется другими техническими средствами системы;
4) изображение на экране.

Для применения *определения 3* для каждого вида результатов необходимо определить метрику с учетом неопределенности. Учитывая удобство применения при решении оптимизационных задач, используем эвклидовую метрику.

Метрику результатов 1-го типа определим соответственно выражению:

$$M_1 = \int_{-\infty}^{+\infty} z^2 \cdot \beta(z) \cdot dz , \tag{5}$$

где $z = x - y$.

Учитывая независимость результатов двух алгоритмов, получаем:

$$M_1 = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} (x - y)^2 \cdot \beta(x) \beta(y) \cdot dx \cdot dy .$$

Метрику результатов 2-го типа определим соответственно выражению:

$$M_2 = \sum_{i=1}^{n} \sum_{j=1}^{n} (x_i - y_i)^2 \cdot \mathrm{B}_{x_i} \cdot \mathrm{B}_{y_j} , \tag{6}$$

где *n* – мощность множества результатов; $\mathrm{B}_\xi : \beta_\xi = \sum_{i=1}^{n} \mathrm{B}_{\xi_i} \cdot \delta[\xi - \xi_i]$; $\delta[\xi]$ – дельта-функция.

Метрику результатов 3-го типа определим аналогично выражению (6), считая, что $\{y\}$ – множество возможных действий технических средств системы.

Метрику результатов 4-го типа вводится отдельно для двух случаев: когда изображения на экране выбирается из определенного множества стандартных изображений, тогда определение метрики аналогично выражению (6), и когда изображение имеет постоянную структуру со сменными параметрами (например, график функции), тогда определение метрики аналогично выражению (5), где x и y множества неопределенных параметров изображений двух алгоритмов.

Рассмотрим пример. Система обработки сигналов, изображенная на рис. 2, может быть выполненная в параллельном (2, а) и последовательном (2, б) вариантах.

Алгоритмическая модель параллельного выполнения будет иметь вид:

M1=A(B) || [Im_1(a(t) / $A\sin[\omega(t+\tau)]$, $\varepsilon 1$) I_2(f(a(t),ξ_1) / u(t)=a(t)+ξ_1) I_3(u(t) / u'(t+τ)) Im_4(d(t) / $D\sin[\omega(t+\tau)]$,$\varepsilon 2$) I_5 (f(d(t),ξ_2) / v(t)=d(t)+ξ_2) I_6(v(t) / v'(t+τ))] I_7(f(u'(t+τ),v'(t+τ)) / x(t)) A(E). (7)

Алгоритмическая модель последовательного выполнения в алгебраическом виде имеет вид:

M2=A(B) Im_1(a(t) / $A\sin[\omega(t+\tau)]$, $\varepsilon 1$) I_2(f(a(t),ξ_1)/u(t)=a(t)+ξ_1) I_3(u(t)/u'(t+τ)) Im_4(d(t+τ) / $D\sin[\omega(t+2\tau)]$,$\varepsilon 3$) I_5 (f(d(t+ τ),ξ_2) / v(t+ τ)=d(t +τ)+ξ_2) I_6(v(t+ τ) / v'(t+2τ)) I_7(f(u'(t+ τ),v'(t+2 τ)) / x(t)) A(E). (8)

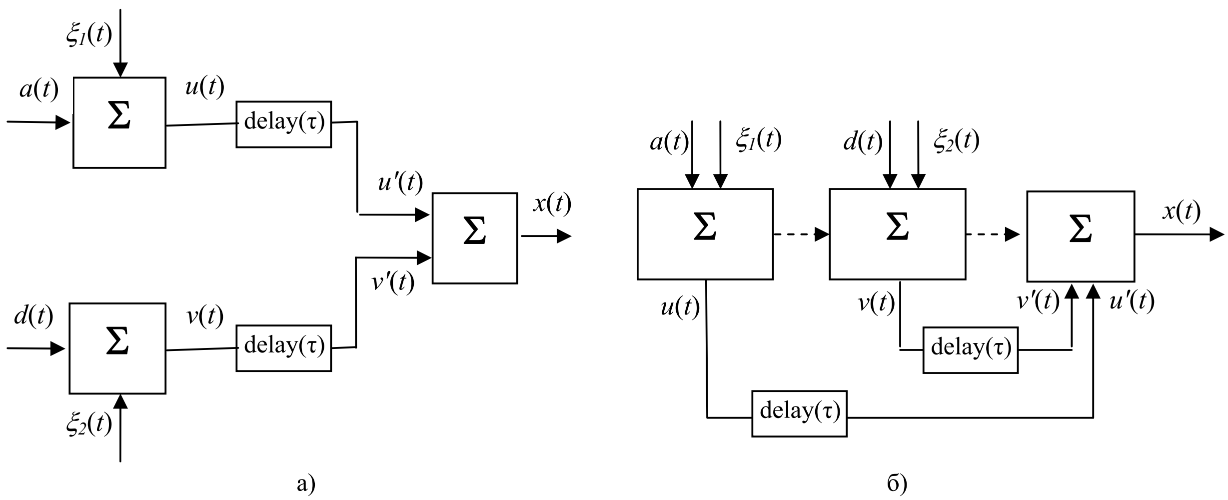

Рис. 2. Пример системы обработки сигналов

Алгоритмические модели (7) та (8) в операторной форме будут иметь вид:

$b_1(x) = F_7(2,+)[\,F_6(n,g_\tau)\,[F_5(2,+)[\beta(\xi_2)\,,\,F_4(2,+)[b(d),\,b(\varepsilon_2)]]]\,,\,F_3(n,\,g_\tau)[F_2(2,+)[b(\xi_1),\,F_1(2,+)[b(a),\,b(\varepsilon_1)]]$,

$b_2(x) = F_7(2,+)[\,F_6(n,g_{2\tau})\,[F_5(2,+)[\beta(\xi_2)\,,\,F_4(2,+)[b(d),\,b(\varepsilon_2)]]]\,,\,F_3(n,\,g_\tau)[F_2(2,+)[b(\xi_1),\,F_1(2,+)[b(a),\,b(\varepsilon_1)]]$.

Пусть сигналы, которые поступают на входы алгоритмов имеют вид:

$$a(t) = A\sin\omega t\,,\;\; d(t) = D\sin\omega t\,.$$

На входы также поступает белый нормальный шум ξ_1 и ξ_2.

В условиях определенности (при отсутствии шума) результат получается с помощью R-модели. Для схем рис. 2, а, б получаем:

а) $A\sin[\omega(t+\tau)] + D\sin[\omega(t+\tau)] = (A+D)\sin[\omega(t+\tau)]$,

откуда $Y_1 = (A+D)$.

б) $A\sin[\omega(t+\tau)] + B\sin[\omega(t+2\tau)] = \sqrt{A^2 + 2AB\cos(\omega\tau) + B^2}\,\sin[\omega t + \phi]$, откуда $Y_2 = \sqrt{A^2 + 2AD\cos(\omega\tau) + D^2}$.

Очевидно, в условиях определенности алгоритмические модели (7) и (8) не эквивалентны.

В условиях неопределенности (при наличии шума) результат преобразований получаем с помощью *G*-модели, которая разрешает определить функции неопределенности результатов $\beta(Y_1^*)$ и $\beta(Y_2^*)$:

а) $Y_1^* = (A + D) + \xi_1 + \xi_2$;

б) $Y_2^* = \sqrt{A^2 + 2AD\cos(\omega\tau) + D^2} + \xi_1 + \xi_2$;

При условии нормального белого шума функции неопределенности результатов будут гаусинами со средними значениями соответственно Y_1 и Y_2 и дисперсиями $D_{Y^*} = \sigma_{\xi_1}^2 + \sigma_{\xi_2}^2$.

Достоверность совпадения результатов в условиях неопределенности:

$$B(Y_1^* = Y_2^*) = \int\limits_{-(A+B)}^{+(A+B)} \left[\frac{1}{2\pi\sqrt{\sigma_{\xi_1}^2 + \sigma_{\xi_2}^2}} e^{-\frac{(Y-Y_1)^2}{2(\sigma_{\xi_1}^2+\sigma_{\xi_2}^2)}}\right]\left[\frac{1}{2\pi\sqrt{\sigma_{\xi_1}^2 + \sigma_{\xi_2}^2}} e^{-\frac{(Y-Y_2)^2}{2(\sigma_{\xi_1}^2+\sigma_{\xi_2}^2)}}\right] dY = \frac{1}{2\sqrt{2\pi}\cdot\sigma} e^{-\left(\frac{y_1-y_2}{2\sigma}\right)^2}.$$

Зависимость степени эквивалентности моделей (7) и (8) в условиях неопределенности от времени выполнения операций и дисперсии шума приведена на рис. 3. Из рис. 3 видно, что при взятых для примера параметрах сигналов и быстродействия блока $0.15 < \tau < 0.25$ достоверность совпадения результатов для систем рис.2, а и рис.2, б имеет максимум при суммарной дисперсии шума $\sigma = 0.1$.

Из анализа эквивалентности моделей вытекает, что при определенных условиях последовательная система, которая может быть реализована с меньшим количеством аппаратных средств, является эквивалентной параллельной системе, причем степень эквивалентности может быть увеличена искусственным введением неопределенности.

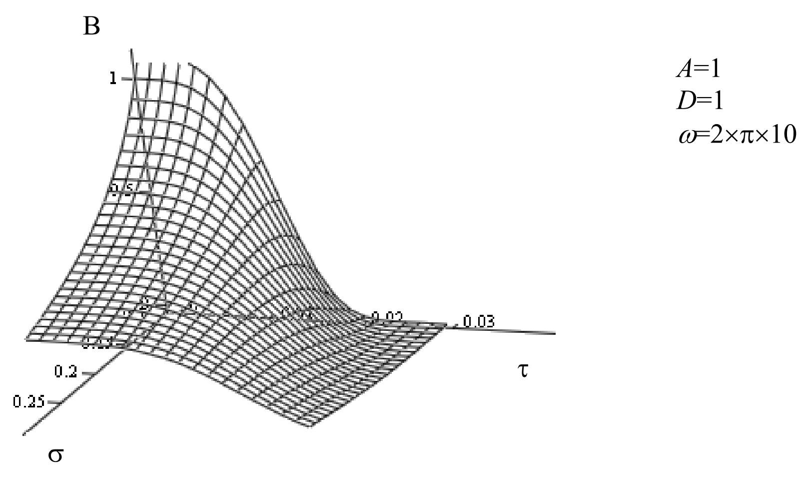

Рис. 3. Зависимость степени эквивалентности моделей в условиях неопределенности от дисперсии шума

Выводы

Для оценивания степени эквивалентности алгоритмов в условиях неопределенности введено понятие степени равенства. Функции неопределенности результатов работы двух алгоритмов, степень эквивалентности которых исследуется, получается с помощью операторного метода. На примере доказано, что алгоритмы, которые неэквивалентные в определенных условиях, могут быть эквивалентными в неопределенных условиях, причем степень эквивалентности может быть увеличена искусственным введением неопределенности.

СПИСОК ЛИТЕРАТУРЫ

1. Дубовой В. М. Визначення вимог до структури підсистеми керування вимірювально-обчислювальної системи/ Дубовой В. М., Никитенко О. Д. // Вісник Хмельницького національного університету. – 2005. – №4. Ч.1. – Т.1(68). С. 115 – 118.

2. Цапенко М. П. Измерительные информационные системы: Структуры и алгоритмы, системотехническое проектирование. Учебн. Пособие для вузов. – 2-е. изд. перераб. и доп. – М.: Энергоатомиздат, 1985. – 439 с.

3. Дубовой В. М. Застосування алгоритмічної моделі до оптимізації інформаційно-обчислювальних систем в умовах невизначеності / Дубовой В. М., Никитенко О. Д. // Вісник ВПІ. – №6. – 2005. – С. 9 – 13.

4. Ляпунов А. А. О логических схемах программ // Проблемы кибер нетики: Сб. статей. – Вып.1. – М.: Фнзматгнз, 1958. – С. 46 – 74.

5. Янов Ю. И. О логических схемах алгоритмов // Проблемы кибернетики: Сб. статей. – Вып. 1. – М.: Физматгиз, 1958. – С. 75 – 127.

6. Криницкий Н. А. Равносильные преобразования алгоритмов и программирование. – М.: Советское радио, 1970. – 304 с.

7. Калужнин Л. А. Об алгоритмизации математических задач // Проблемы кибернетики: Сб. статей. – Вып. 2. – М.: Физматгиз, 1959. – С. 51 – 67.

8. Котов В. Е., Сабельфельд В. К. Теория схем программ. – М.: Наука, 1991. – 248 с.

9. Глонь О. В., Дубовой В. М. Моделювання систем керування в умовах невизначеності: Монографія: – Вінниця: УНІВЕРСУМ-Вінниця, 2004. – 169 с.

Дубовой Владимир Михайлович – д. т. н., профессор, заведующий кафедры компьютерных систем управления, тел.: (0432) 598-157, E-Mail: dub@faksu.vstu.vinnica.ua.

Никитенко Елена Дмитриевна – аспирантка кафедры компьютерных систем управления, тел.: (0432) 27-20-44, E-Mail: lena_2607@mail.ru.

Глонь Ольга Витальевна – к. т. н., доцент кафедры компьютерных систем управления, тел.: (0432) 598-222

Винницкий национальный технический университет.