

УДК 519.71

А.А. Бессонов, В.А. Бобух, О.Г. Руденко

ПРИМЕНЕНИЕ НЕЙРОННОЙ СЕТИ СМАС ПРИ РЕШЕНИИ ЗАДАЧ ОБРАБОТКИ ИЗОБРАЖЕНИЙ И ЕЕ АППАРАТНАЯ РЕАЛИЗАЦИЯ

Введение

Искусственная нейронная сеть СМАС – *Cerebellar Model Articulation Controller* (церебральная модель артикуляционного контроллера), описывающая процессы управления движением, происходящие в мозжечке и первоначально предложенная для управления роботом манипулятором [1, 2], оказалась достаточно эффективной не только в задачах управления, но и в задачах аппроксимации, восстановления и интерполяции одно- и многомерных функций, цифровой обработки сигналов и т.д.

Отличительной особенностью СМАС является использование в ней специального кодирования, позволяющего резко уменьшить объем памяти, необходимый для хранения обрабатываемой информации. Кодирование информации заключается в том, что каждому N -мерному входному вектору $x(i)$ ставится в соответствие n -мерный ($n < N$) вектор ассоциаций $a(i)$, $\rho = const \ll n$ элементов которого могут принимать отличные от нуля значения из интервала $[0, 1]$ (в классической СМАС они принимаются равными 1), то есть только ρ элементов памяти являются активными. Дополнительное уменьшение памяти достигается применением хеширования информации.

В статье рассматривается применение сети СМАС для фильтрации и кодирования изображений и предлагается один из вариантов ее аппаратной реализации.

Нейронная сеть СМАС

Сеть СМАС, пример которой приведен на рис. 1, в общем случае осуществляет следующие преобразования:

$$S: X \Rightarrow A, \quad (1)$$

$$H: A \Rightarrow A', \quad (2)$$

$$P: A' \Rightarrow y, \quad (3)$$

где X – N -мерное пространство непрерывных входных сигналов; A – n -мерное пространство ассоциаций; A' – преобразованное алгоритмом хеширования пространство ассоциаций; y – вектор выходных сигналов.

Преобразование (1) соответствует кодированию информации

$$a = S(x), \quad (4)$$

(2) – хешированию

$$a' = H(a), \quad (5)$$

а (3) – вычислению выходного сигнала

$$\hat{y} = P(a') = (a')^T w = (H(a))^T w. \quad (6)$$

Важным вопросом при реализации сети является выбор базисных функций.

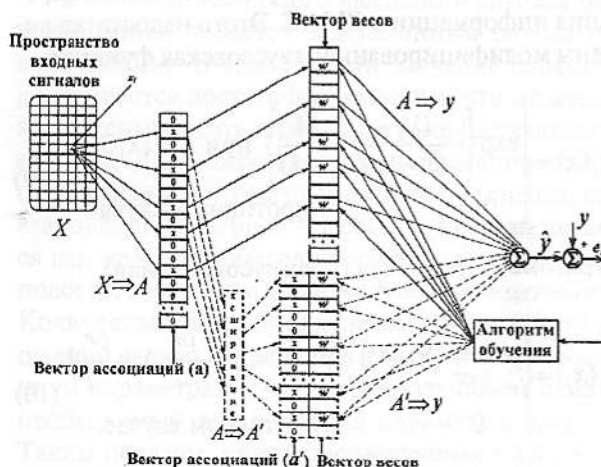


Рис. 1. Структура сети СМАС

Выбор базисных функций

Выражение (6) описывает преобразование, осуществляемое в традиционной СМАС с использованием хеширования информации при выборе прямоугольных базисных функций. Если же в сети используются нейроны с активационными функциями, отличными от прямоугольной, преобразование (6) принимает вид

$$\hat{y} = H(a^T \Phi(x)) w, \quad (7)$$

$$\text{где } \Phi(x) = \begin{bmatrix} \Phi_1(x) & 0 & \dots & 0 \\ 0 & \Phi_2(x) & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \Phi_n(x) \end{bmatrix},$$

где $\Phi_i(x) = \prod_{j=1}^N \phi_{ij}(x_j)$; $\phi_{ij}(x_j)$ – значение выбранной базисной функции в точке x_j .

В традиционной СМАС используются прямоугольные базисные функции, что позволяет осуществлять кусочно-постоянную аппроксимацию. В этом случае вычислительные затраты будут минимальными, вследствие чего значительно сокращается время реакции сети на поступивший вход-

ной сигнал. Компоненты вектора ассоциаций при этом могут принимать значения 0 или 1. Скорость обучения сети при выборе базисных функций прямоугольной формы будет максимальной [3].

В качестве базисных в сети СМАС часто используются гауссовские функции

$$\Phi_i(x_j) = \exp\left\{-\frac{(x_j - \mu_i)^2}{\sigma_i^2}\right\}. \quad (8)$$

Однако, хотя гауссовская функция и обладает свойством локального возбуждения, достаточно сложно четко выделить ее границы возбуждения, что является важным для осуществления кодирования информации в СМАС. Этому недостатка лишены модифицированная гауссовская функция

$$\Phi_i(x) = \begin{cases} \exp\left\{-\frac{(\lambda_2 - \lambda_1)^2 / 4}{(x - \lambda_1)(\lambda_2 - x)}\right\} & \text{при } x \in (\lambda_1, \lambda_2), \\ 0 & \text{в противном случае} \end{cases} \quad (9)$$

и тригонометрическая (косинусоидальная)

$$\Phi_i(x_j) = \begin{cases} \cos\left(\frac{\pi}{\rho r_j}(x_j - \lambda_i)\right) & \text{при } x_j \in (\lambda_i - \frac{\rho r_j}{2}, \lambda_i + \frac{\rho r_j}{2}), \\ 0 & \text{в противном случае,} \end{cases} \quad (10)$$

где λ_i — i -й центр области квантования; r_j — шаг квантования по j -ой компоненте входного сигнала.

Алгоритмы хеширования информации в СМАС

Хеширование заключается в отображении большой требуемой памяти в меньшую, физически реализуемую. Объем большей памяти определяется размерностью вектора ассоциаций, полученного на этапе кодирования. Если адреса ячеек большей памяти лежат в интервале $[1, M]$, а меньшей — в интервале $[1, m]$, то функция преобразования адреса (хеш-функция) должна удовлетворять требованию

$$1 \leq H(k) < m, \quad \forall 1 \leq k \leq M.$$

Выбор функции $H(k)$ является весьма трудной задачей, однако многочисленные исследования показали, что хорошие результаты обеспечивает применение двух основных типов хеш-функций, один из которых основан на делении, а другой — на умножении [4].

Хеш-функция, реализующая метод, основанный на делении, имеет вид

$$H(k) = 1 + k \bmod m, \quad (11)$$

где $k \bmod m$ означает деление по модулю m .

Адрес ячейки памяти с меньшим объемом при методе хеширования, основанном на умножении, вычисляется по следующей формуле:

$$H(k) = 1 + \left\lfloor m \left[\left(\frac{F}{w} k \right) \bmod 1 \right] \right\rfloor, \quad (12)$$

где w — размер машинного слова; F — некоторая целая константа, взаимно простая с w ; $\lfloor \bullet \rfloor$ — означает округление в сторону ближайшего целого числа.

В качестве F возможно использование чисел Фибоначчи.

Хеш-функция, основанная на делении, равномерно распределяет большую память в меньшую в соответствии с адресом.

Функция, описываемая уравнением (12), «тащит» элементы памяти.

Хеш-коллизии

Какой бы метод хеширования ни применялся, невозможно полностью устранить проблему хеш-коллизий. При хешировании информации в сети СМАС возможно возникновение двух типов коллизий. Коллизии первого типа возникают при активации одним входным сигналом одной ячейки памяти более одного раза. Вероятность того, что один входной сигнал обратится два и более раз к одной ячейке физической памяти, можно определить по формуле

$$P = \sum_{i=2}^{\rho-1} \frac{i}{m}, \quad (13)$$

где m — объем меньшей памяти.

Так, например, для случая $\rho = 20$, $m = 2000$ вероятность многократного обращения к одной ячейке будет равна примерно 0,1, то есть является достаточно малой. Возникновение такой коллизии означает то, что значение, хранящееся в многократно активизируемой ячейке, несколько раз участвует в формировании выходного сигнала.

Причиной возникновения коллизий второго типа является активация одной и той же ячейки памяти различными далеко расположенными друг от друга (в смысле расстояния Хэмминга) входными сигналами. Эти коллизии приводят к более острой проблеме — появлению нежелательных корреляций двумя далеко расположенными (в смысле расстояния Хэмминга) входными сигналами. Эти коллизии не могут существенно повлиять на работу сети, пока выполняется условие

$$a'_1 \wedge a'_2 \ll \rho,$$

где a'_1 и a'_2 — преобразованные хеш-функцией ассоциации двух различных входных сигналов.

Для любых двух входных сигналов, для которых не выполняется $a'_1 \wedge a'_2 = \phi$ (здесь ϕ — пустое множество), вероятности обращения к обеим ячейкам можно определить следующим образом

$$P_n(k) = C_n^k p^k q^{n-k}, k = \overline{1, n}, \quad (14)$$

где C_n^k — число сочетаний; $p=1-q$.

Для нашего примера $q = \frac{20}{2000}$. В этом случае вероятность того, что $a'_1 \wedge a'_2 = 0$, то есть $P_{20}(20)$, равна 0,818. Вероятности же того, что $a'_1 \wedge a'_2 = 1$, $a'_1 \wedge a'_2 = 2$, $a'_1 \wedge a'_2 \geq 3$ будут равны соответственно 0,165; 0,016; 0,001.

Обучение

Обучение сети СМАС состоит в настройке вектора ее весовых параметров w на основании предъявления ей обучающих пар $\{x(k), y(k)\}$ и сравнения реакции сети $\hat{y}(k)$ с требуемой $y(k)$. Соответствующий алгоритм обучения может быть записан следующим образом:

$$w(k+1) = w(k) + \frac{y(k) - \hat{y}(k)}{\|\Phi(x(k))a(k)\|^2} \Phi(x(k))a(k). \quad (15)$$

Однако в ряде случаев для улучшения аппроксимирующих свойств сети возникает необходимость настройки не только весовых параметров сети, но и параметров базисных функций. Наиболее эффективным оказывается применение такого подхода для настройки центров μ и дисперсий σ гауссовских функций активации (8).

Настройка данных параметров может быть осуществлена, например по алгоритму, аналогичному алгоритму Уидроу-Хоффа:

$$m(k+1) = m(k) + \frac{y(k) - \hat{y}(k)}{\left\| \frac{\partial \hat{y}(k)}{\partial m(k)} \right\|^2} \frac{\partial \hat{y}(k)}{\partial m(k)},$$

где m — вектор настраиваемых параметров.

В случае использования гауссовских функций в качестве m используются центры μ и дисперсии

σ , а векторы частных производных $\frac{\partial \hat{y}(k)}{\partial \mu(k)}$ и $\frac{\partial \hat{y}(k)}{\partial \sigma(k)}$

вычисляются по формулам

$$\frac{\partial \hat{y}(k)}{\partial \mu_{ji}(k)} = 2w_i(k) \prod_{s=1}^N e^{-\frac{(x_s - \mu_{sj}(k))^2}{\sigma_{sj}(k)^2}} \frac{(x_j - \mu_{ji}(k))}{\sigma_{ji}(k)^2},$$

$$\frac{\partial \hat{y}(k)}{\partial \sigma_{ji}(k)} = 2w_i(k) \prod_{s=1}^N e^{-\frac{(x_s - \mu_{sj}(k))^2}{\sigma_{sj}(k)^2}} \frac{(x_j - \mu_{ji}(k))^2}{\sigma_{ji}(k)^3},$$

где i — номер возбужденного нейрона; j — номер компонент векторов ассоциаций и весов, соответствующих данному возбужденному нейрону.

Задача фильтрации изображений

Для фильтрации изображений предлагается использовать иерархическую сеть СМАС, состоящую из нескольких подсетей, расположенных пос-

ледовательно и отличающихся количеством ступеней квантования (число ступеней квантования определяется параметром ρ), и представленную на рис. 2 [5, 6].

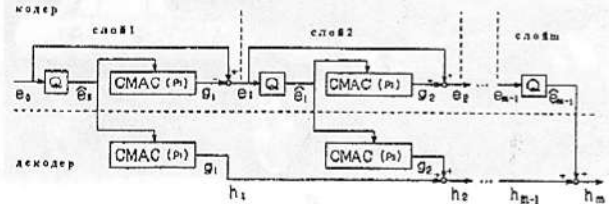


Рис. 2. Схема фильтрации изображений

В качестве желаемого выходного сигнала первой подсети подается зашумленный сигнал или изображение. В этой подсети значение параметра ρ выбирается достаточно большим, что позволяет аппроксимировать низкочастотные составляющие сигнала. Далее вычисляется ошибка аппроксимации (из исходного сигнала вычитается сигнал, восстановленный первой подсетью), которая подается как желаемый выходной сигнал для следующей подсети, с меньшим числом ступеней квантования. Количество подсетей определяется характером решаемой задачи. Следует отметить, что с уменьшением параметра ρ будет возрастать объем памяти, необходимый для хранения параметров подсети. Таким образом, каждая последующая подсеть будет содержать большее число настраиваемых параметров.

Пусть оригинальное изображение имеет размеры $N \times N$ и обозначается $e_0(x, y)$. Для СМАС 1-го слоя выбирается параметр $\rho_1 \approx 0,5N$ и обучается данная сеть на дискретизированном оригинальном изображении, причем сети в качестве обучающих пар предъявляются не все пиксели изображения, а лишь те, которые лежат на пересечении строк и столбцов, номера которых кратны ρ_1 (этот принцип обучения используется для всех подсетей). Информация о пропущенных пикселях частично восстанавливается текущей подсетью, а затем будет восстановлена последующими подсетями. Изображение, аппроксимированное первой подсетью, обозначим $g_1(x, y)$. Тогда ошибка аппроксимации после обучения первой подсети равна $e_1(x, y) = e_0(x, y) - g_1(x, y)$. Ошибка $e_1(x, y)$ является входным сигналом второй подсети. Для этой подсети рекомендуется выбирать параметр ρ_2 равный $0,5\rho_1$. Уточнение иерархической структуры прекращается при достижении желаемой точности восстановления изображения.

На рис. 3 представлены результаты фильтрации изображения размером 400×200 пикселей. На рис. 3а показано оригинальное изображение, а на рис. 3б — зашумленное. На рис. 3в, 3е приведены результаты восстановления изображения иерархической сетью СМАС, состоящей из четырех слоев.

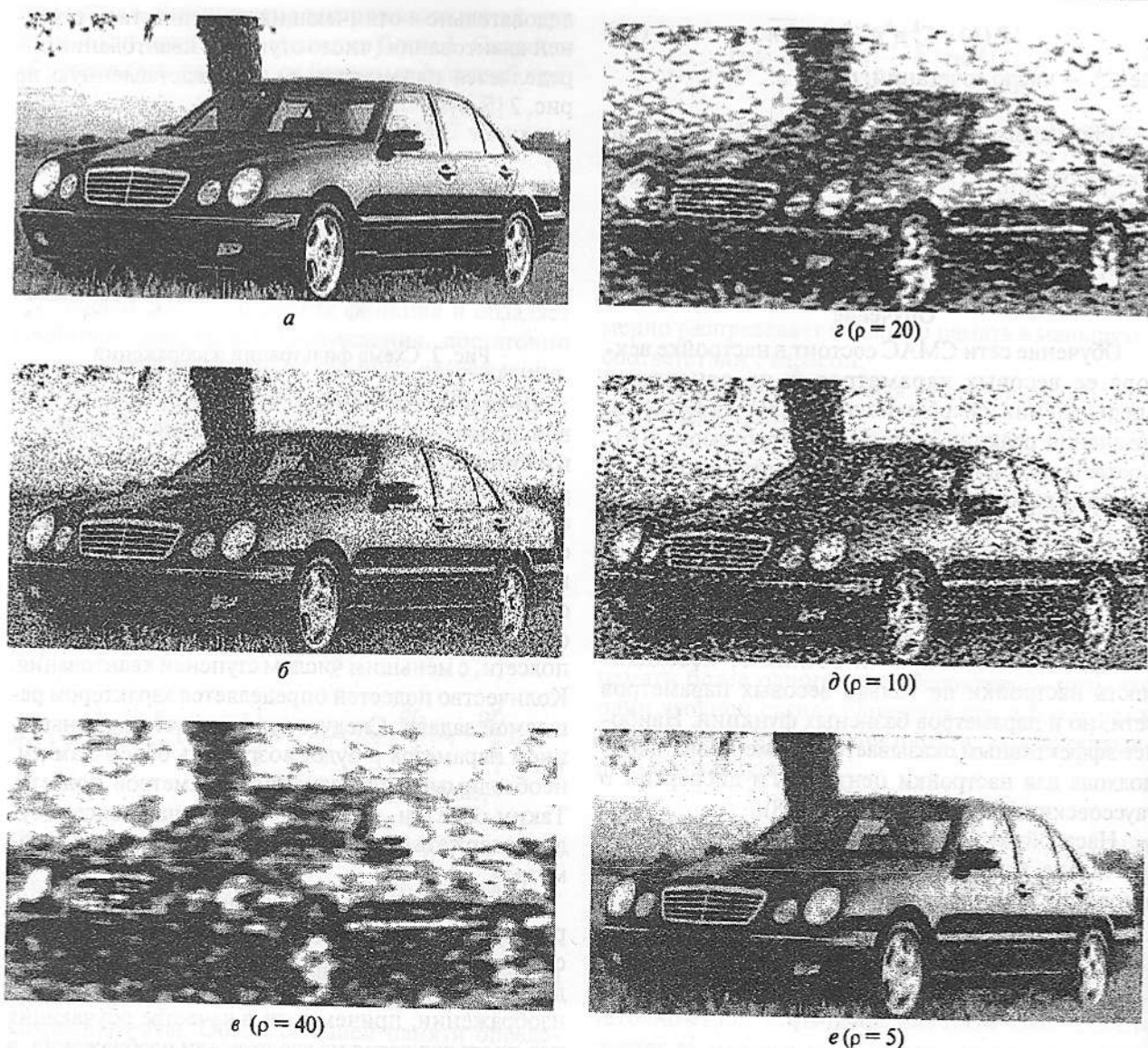


Рис. 3. Пример фильтрации изображения с помощью сети СМАС

В первом слое параметр s выбирался равным 40, во втором – 20, в третьем – 10, а в четвертом – 5. Во всех подсетях использовались косинусоидальные базисные функции. Общий объем требуемой памяти составил $n_{\text{общ}} = 32644$ ячеек.

Кодирование изображений

В работе [5] показано, что нейронная сеть СМАС может быть эффективно применена для решения задач кодирования и сжатия изображений. В данной работе исследовалось влияние выбора параметров сети СМАС (R и ρ), а также настройки параметров базисных функций на качество кодирования. На рис. 4а представлено оригинальное изображение. На рис. 4б – изображение, восстановленное с помощью нейронной сети СМАС с параметрами $R=256$, $\rho = 20$, в которой не осуществлялась настройка параметров базисных функций (то есть они были заданы априорно). Коэффициент сжатия при этом равен $k = 17,33$. На рис.

4в приведены результаты кодирования изображения с использованием настройки параметров. При этом количество хранимых параметров несколько увеличилось, и коэффициент сжатия составил $k = 13,37$.

Аппаратная реализация

Рассмотрим кодирование переменной x_i , представляющей собой n -разрядное двоичное число. Так как входной слой сети СМАС содержит ρ ступеней квантования, таких, что для любой ступени существует другая ступень, на которой области квантования сдвинуты относительно рассматриваемой ступени на один уровень квантования входной переменной x_i , являющейся двоичным числом, то такой сдвиг соответствует единице младшего разряда входной переменной. Присвоим ступени квантования индекс $j = \overline{0, \rho-1}$.

В связи с тем, что область квантования содержит ρ уровней квантования входной переменной,



ия изображе-
аметров. При
ов несколько
тия составил

нной x_i , пред-
ичное число.
держит ρ сту-
обой ступени
рой области
рассматрива-
ования вход-
ным числом,
ладшего раз-
ступени кван-

вания содер-
переменной,



a



б



в

Рис. 4. Пример кодирования изображения с помощью сети СМАС

активационная функция нейрона, соответствующего любой области квантования, определена в точках. Аргументом этой функции будет являться значение

$$u = (x_i + j) \bmod \rho, \quad (16)$$

где \bmod — функция вычисления остатка от целочисленного деления.

Номер активного нейрона на ступени j можно определить как

$$v = (x_i + j) \operatorname{div} \rho, \quad (17)$$

где div — функция целочисленного деления.

Как известно, аппаратная реализация операции деления на произвольное число является либо ресурсоёмкой либо медленной. Поэтому предлагается выбирать значения ρ , равные 2^m . В этом случае операция деления может быть заменена операцией сдвига на m разрядов вправо, а остаток от деления, записанный в m младших разрядах, исчезает.

В классическом варианте сети СМАС активационные функции нейронов областей квантования имеют прямоугольную форму. Однако, как показывают исследования, лучший результат достигается при использовании функций высших порядков (линейных, квадратичных, кубических), а также тригонометрических и степенных функций.

Аппаратная реализация широкого спектра активационных функций, отличных от прямоугольных, сложна и ресурсоёмка. Поэтому целесообразно функцию активации задавать таблично в ОЗУ, содержащем 2^m ячеек. Разрядность ячеек ОЗУ h определяется областью допустимых значений функции активации, а также необходимой точностью задания значений функции активации. Если, например, $m = 8$, то необходимо ОЗУ ёмкостью 256 ячеек. Блоки ОЗУ такого объёма имеются в большинстве современных ПЛИС (например семейство ACEX1K фирмы ALTERA). Подобная реализация функций активации является простой и гибкой, так как позволяет задать функцию любого вида.

Структура блока кодирования переменной x_i представлена на рис. 5. Код переменной x_i защелкивается в регистре RgX , откуда поступает на сумматор $\Sigma 1$. На второй вход этого сумматора поступает m — разрядное значение номера ступени j , на которой в данный момент происходит кодирование. Это значение формируется устройством случайной выборки ступени, служащим для устранения диагональной структуры ассоциативного поля, и, соответственно, улучшения аппроксимирующих свойств сети. С выхода сумматора n -разрядный код поступает на логический сдвигатель (LSh — Logical Shifter), позволяющий изменять значение m . Значение u определяют m младших разрядов выходного кода LSh, а значение v — соответственно старшие $k = n - m$ разрядов. Значение u через мульти-

плексор MX1 поступает на адресные входы ОЗУ хранения активационной функции. Мультиплексор MX1, а также буфер В1 служат для инициализации ОЗУ значениями активационной функции от устройства управления. С выхода ОЗУ h — разрядное значение функции активации поступает на регистр Rg VFS, а с регистра — на выход VFS_i (Value of Function on Stair i) блока кодирования (CB — Coding Block). Значение v поступает на регистр Rg NNS, а оттуда — на выход NNS_i (Number of Neuron on Stair i) блока. Тактирующие импульсы на регистры, а также сигналы управления ОЗУ формируются устройством управления, общим для всей структуры.

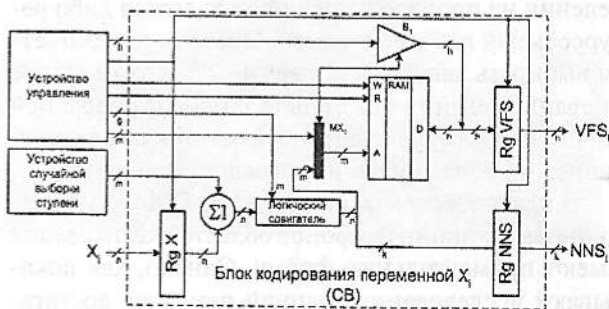


Рис. 5. Блок кодирования переменной x_i

Таким образом, для кодирования переменной x_i с помощью m ступеней квантования необходимо 2^m тактов. При $m < 5$ такое кодирование может осуществляться с частотой 10–20 МГц, то есть в реальном времени для обработки телевизионных изображений. В течение этих 2^m тактов значение x_i хранится во входном регистре RgX и обновляется по их истечении.

Устройство случайной выборки ступени может представлять собой m -разрядный линейный рекуррентный регистр с обратными связями, определяемыми простым полиномом. Такой регистр генерирует $2^m - 1$ различных кодовых комбинаций, обладающих статистическими свойствами равномерно распределённой случайной последовательности. Одна комбинация является запрещённой, так как на очередном такте работы регистра приводит к генерации самой себя. Так как работа блока кодирования предусматривает использование 2^m комбинаций, необходимо предусмотреть её аппаратную генерацию и подстановку вместо кода регистра на любом такте (при этом не производится очередной сдвиг регистра). Вид последовательности определяется начальной установкой регистра. Эта операция производится однократно перед первым обучением сети. Регистры, отвечающие за различные входные переменные, должны быть проинициализированы различными значениями. Эти значения, равно как и начальные значения весов в сети, могут быть сформированы аппаратным датчиком

случайных чисел [7]. Рассмотрим возможную структуру аппаратной реализации сети СМАС, представленную на рис. 6. Сеть обрабатывает две входные переменные, каждая из которых кодируется блоками СВ, содержащими равное число ступеней квантования. Функционирование такой структуры состоит в следующем. На j -м такте блоки СВ формируют значения номеров нейронов в текущих ступенях (NNS₁ и NNS₂) и значения их функций активации (VFS₁ и VFS₂). Значения VFS поступают на умножитель M1, формирующий произведение функций активации текущих ступеней. Значение этого произведения поступает на умножитель M2, вычисляющий произведение веса и произведения функций активации. Адрес веса в ОЗУ весов определяется алгоритмом хеширования, входным параметром которого является комбинация значений NNS. Накапливающий сумматор Σ2 вычисляет сумму произведений весов и функций активации и после выполнения m тактов в регистре RgY формирует значение функции, соответствующее входным переменным. С помощью сумматора Σ3 формируется код ошибки e_k , определяемый как разность полученного u и требуемого y_k значений функции. Значение ошибки поступает на блок реализации алгоритма обучения, который в течение m тактов вычисления накапливал адреса выборки весов, значения весов и произведений функций активации. Получив значение ошибки, алгоритм обучения модифицирует участвовавшие в вычислении веса и записывает их в ОЗУ весов по соответствующим адресам. Координацию работы узлов сети осуществляет устройство управления.

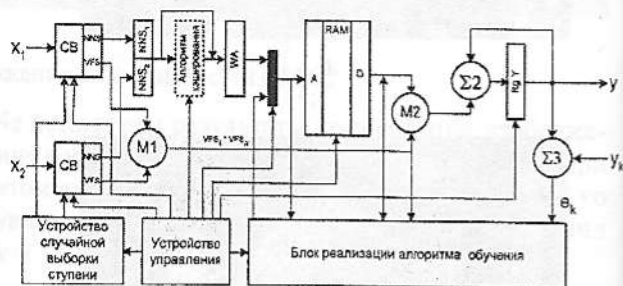


Рис. 6. Вариант аппаратной реализации сети СМАС

Основные результаты и выводы

Как показывают результаты моделирования, нейронная сеть СМАС является достаточно эффективной и удобной при решении задач фильтрации и кодирования изображений. Существенная экономия требуемой памяти в сочетании с простыми алгоритмами обучения делают ее особенно привлекательной для реализации в микроконтроллерных системах. Следует отметить, что приведенная в статье аппаратная реализация сети СМАС является универсальной и может использоваться не только

в задачах обработки изображений, но и для аппроксимации функций, идентификации нелинейных объектов и управления ими. Это связано с тем, что, во-первых, при этом возможно использование любых базисных функций, а во-вторых – в схеме предусмотрен блок хеширования информации. Дело в том, что вопрос выбора базисных функций нейронной сети СМАС остается открытым и определяется характером решаемой прикладной задачи. Хеширование же необходимо при обработке больших объемов информации (изображения высокого качества, объекты большой размерности).

Необходимо также отметить, что применение сети СМАС для фильтрации изображений является менее эффективным, однако в ряде случаев возможным. Сложность реализации иерархической СМАС не позволяет использовать ее, если фильтрация изображений должна происходить в реальном времени.

Список литературы: 1. *Albus J.S.* A new approach to manipulator control: the cerebellar model articulation controller (CMAC) // ASME Trans., J. Dynamic Systems, Measurement and Control, 1975. Vol. 97. №3. P. 220–227. 2. *Albus J.S.* Data storage in cerebellar model articulation controller (CMAC) // ASME Trans. J. Dynamic Systems, Measurement and Control, 1975. Vol. 97. №3. P. 228–233. 3. *Руденко О.Г., Бессонов А.А.* О выборе базисных функций в нейронной сети СМАС // Проблемы управления и информатики. 2004. №2. С. 143–154. 4. *Кнут Д.Э.* Искусство программирования, том 3. Сортировка и поиск, 2-е изд., М.: Вильямс, 2000. С. 549–597. 5. *Руденко О.Г., Бессонов А.А., Бобух В.А.* Фильтрация изображений с помощью нейронной сети СМАС // Вестник Херсонского гос. техн. уни-та. 2004. №1(19). С. 378–384. 6. *Liguni Youji.* Hierarchical image coding via Cerebellar Model Arithmetic Computers (CMAC) // IEEE Trans. on Image Processing, 1996. Vol.5. №10. P. 1393–1401. 7. *Торба А.А., Елаков С.Г., Степченко А.З.* Генерация равновероятных случайных последовательностей на основе физических датчиков // «Радиотехника»: Всеукр. межвед. науч.-техн. сб. Харьков. ХТУРЭ. 2001. Вып. 119. С. 108–113.

Поступила в редколлегию 22.11.2004