

ИНТЕГРИРОВАННЫЙ КОНТУР ОБУЧЕНИЯ, КАК СРЕДСТВО ПОВЫШЕНИЯ ПРОИЗВОДИТЕЛЬНОСТИ И ФУНКЦИОНАЛЬНОСТИ ЦИФРОВОГО НЕЙРОСЕТОВОГО УСТРОЙСТВА

Казанцев П.А.

Научный центр нейрокомпьютеров при РАСУ
105066, Москва, Новая Басманная ул, 20, тел. 263-94-30

Общее описание. Обучение является одним из наиболее важных свойств нейросистем. Алгоритм обучения более сложен и менее регулярен, чем алгоритм самих вычислений. За время эксплуатации устройства обучение, зачастую, проводится лишь один раз. Внедрение контура обучения на кристалл является довольно спорным моментом, поэтому необходимо выделить те области задач, в которых оно было бы оптимально с точки зрения отношения производительности к стоимости.

Для реализации обучения требуются дополнительные аппаратные или программные модули. Иногда, некая часть аппаратного модуля используется, как для самих вычислений, так и для обновления весовых коэффициентов (например, один умножитель может использоваться для умножения веса на входной сигнал, ошибки, и приращения веса). В этом случае следует увеличить фактор виртуализации ζ_S (обычно в два или три раза), так как для симуляции дополнительных операций используется один и тот же обрабатывающий элемент [1]. В полностью параллельных системах аппаратные части обычно используются только для одной цели (для самих вычислений или обучения), так как фактор $\zeta_S = 1$ в них не изменяется.

Можно выделить три основных типа реализации обучения :

•**Автономное обучение.** Обучение производится на внешнем компьютере (например, с помощью самостоятельных или встроенных пакетов моделирования) без использования специализированного аппаратного модуля. Для этого в симуляторе необходимо построить точную модель аппаратного модуля и/или создать алгоритм хорошо совместимый с аппаратной реализацией. В этом случае, аппаратный контур обучения не нужен.

Весы можно хранить в ПЗУ (в СИС для реализации жесткого обучения) или в памяти с однократной записью (в ПЛИС для более гибкого предварительного обучения), или в перезаписываемом ОЗУ (для обучения после изготовления, которое является более гибким, но требует наличия энергонезависимой памяти).

•**Обучение посредством внутрикристалльной прогонки.** Алгоритм обучения выполняется на внешнем компьютере, который связан с обучаемым (настраиваемым) аппаратным модулем посредством интерфейсных устройств. Последний производит сами вычисления, а внешний компьютер обновляет веса (шаг обучения). Фактическая модель аппаратного модуля в этом случае не требуется. Для *непрерывного обучения*, внешний компьютер должен быть всегда подключен к устройству. В этом случае, аппаратный контур обучения не нужен.

В гибридных программно-аппаратных реализациях используется несколько измененный метод обучения посредством внутрикристалльной прогонки. Алгоритм обучения реализуется в виде программы, выполняемой процессором, а сами вычисления осуществляются на цифровом аппаратном модуле. Такая комбинация обеспечивает очень высокую скорость самих вычислений и высокую гибкость обучения (которое зачастую приходится проводить на пониженной скорости).

Весы необходимо хранить в перезаписываемой и энергонезависимой памяти. Исключение составляет непрерывное обучение.

•**Накристалльное обучение .** Обучение реализуется путем введения в кристалл дополнительного контура. Хотя внешний компьютер в этом случае не нужен, этот способ используется редко, так как устройство аппаратного модуля сильно усложняется (цена возрастает в два-три раза). Не имеет смысла встраивать дополнительные контуры, если использоваться они будут только один раз. Накристалльное обучение приобретает смысл, когда требуется реализация непрерывного обучения (например, в адаптивных нейросистемах).

Весы необходимо хранить в перезаписываемой и энергонезависимой памяти. Исключение составляет непрерывное обучение.

Рассмотрев вышеизложенные реализации обучения нейросистемы, можно для каждой из них определить класс (классы) задач, который решался бы нейросистемой с этим типом обучения наиболее эффективно. Разумеется, задачи в данном случае следует классифицировать с точки зрения аппаратной реализации их решения и области применения системы. Приведем классификацию задач по следующим признакам:

а) Скорость изменения весов в режиме работы.

При обучении первого типа веса поступают в кристалл с хост-машины, что исключает возможность применения этого способа в задачах, где требуется *быстрое* изменение весов, так как последовательное вычисление весов и их подача через интерфейсные устройства обуславливает неприемлемую задержку в работе нейромодуля. Во втором способе сами веса обновятся быстро, однако, нейромодуль придется переключить в режим обучения, останавливая прямые вычисления. Третий способ (накристалльное обучение) позволяет избежать ограничений предыдущих двух способов – веса вычисляются параллельно, их запись происходит непосредственно во внутрикристалльную память, прямые вычисления не останавливаются.

б) Выбор быстроменяющихся данных: веса или входные признаки?

Нейросетевые постановки задач можно условно разбить на два подкласса:

- 1) Входные признаки меняются быстро, веса - медленно (например, задача классификации)
- 2) Входные признаки меняются медленно, веса – быстро (например, решение системы алгебраических уравнений)[2].

При создании нейромодуля целесообразно не делать логического разбиения веса/входные признаки. Целесообразно классифицировать входы умножителей, как быстрые и медленные. Таким образом, нейромодуль будет универсален относительно указанных двух подклассов задач. Путем всего лишь одного управляющего однобитового сигнала можно менять назначение всего нейромодуля. Если реализация аппаратной поддержки для первого подкласса очевидна (веса записаны во внутрикристалльное ОЗУ, признаки подаются на кристалл через ножки), то реализация второго подкласса затруднительна, по той причине, что веса могут быть различны для каждого нейрона. Для примера возьмем вертикальный фрагмент нейросети [3] с числом входов и нейронов равным 8 (веса и признаки имеют одинаковую разрядность). Тогда количество весов равняется 64. Т.е. число ножек корпуса кристалла необходимых для одновременной подачи весов больше числа ножек необходимых для подачи признаков в количество раз равное количеству нейронов. Соответственно, из-за нехватки ножек корпуса кристалла, веса придется подавать последовательно – проблема сводится к проблеме, описанной в п. а). Следовательно, для эффективной реализации возможности выбора быстрых/медленных входов умножителей требуется накристалльное обучение.

в) Автономность нейромодуля

Очевидно, что развязывание нейросистемы и учителя-человека обуславливает ее автономность. Накристалльное обучение с этой точки зрения является единственным способом обеспечения автономности нейросистемы.

г) Требование к масштабированию(расширяемости)

Предположим, что некий класс задач требует расширяемости системы за счет физического каскадирования базового фрагмента (например, вертикальных фрагментов нейросети). Примером может служить увеличение числа алгебраических уравнений в системе. Увеличение количества нейросетевых фрагментов неизбежно влечет за собой расширение контура обучения, если мы хотим сохранить массовый параллелизм системы. Реализуя обучение по первому способу, мы вынуждены выносить контур обучения одного фрагмента на отдельный универсальный процессор. То есть вся система разрастается по двум веткам: с одной стороны это фрагменты нейросети, с другой – контуры обучения. Причем, если реализация каскадируемости нейросетевого фрагмента была физически предусмотрена, то реализацию физического каскадирования контуров обучения на универсальных процессорах предстоит переделывать на каждом этапе наращивания системы. Второй способ обучения не имеет подобной проблемы, так как для рассылки команд вполне хватит небольшого числа универсального процессора, однако остается проблема простоя прямых вычислений. При этом, если система выросла до размеров стойки, то факт дороговизны системы, а следовательно и машинного времени, очевиден. Накристалльное обучение не имеет обоих недостатков, которые присутствуют у первых двух способов. Следует отметить, что накристалльный контур обучения также должен быть каскадируемым, ибо по сути своей представляет собой ничто иное, как нейронную сеть.

д) Время изготовления продукта

Очевидно, что введение дополнительного аппаратного алгоритмического контура в систему влечет за собой увеличение времени разработки. На первый взгляд, первые два способа имеют преимущество над третьим с точки зрения выигрыша во времени разработки. Однако, развитые системы автоматизированного проектирования и технология программируемых логических интегральных схем позволяют сильно сократить время разработки.

г) Стоимость нейромодуля

Разумеется, себестоимость цифровой нейросистемы на кристалле в основном определяется стоимостью самого кристалла и стоимостью разработки, которая в свою очередь определяется временем необходимым для ее завершения. Себестоимость реализации накристалльного обучения безусловно выше, чем себестоимость реализаций по первым двум способам. Сегодня, однако, стоимость разработки можно сократить (см. п.д)). Более того, как следует из вышеизложенных пунктов, нейромодуль с накристалльным обучением более универсален и автономен, поэтому позволяет не вовлекать дополнительные аппаратные средства в процессе эксплуатации самого модуля, а также наращиваемой системы нейромодулей.

Выводы

Учитывая вышеприведенное рассмотрение, можно сделать вывод, что наличие накристалльного контура обучения цифровой нейросети крайне желательно при ориентации на увеличение производительности многофункциональной, и как следствие нацеленной на решение сложных задач реального времени, нейросетевой системы.

Литература.

1. L.M. Reyneri, "Implementation Issues of Neuro-Fuzzy Hardware: Going Towards HW/SW Design", IEEE Transactions on Neural Networks, Vol. 14, №1, January 2003, pp. 176-194.
2. Под ред. Галушкина А.И., Нейроматематика, ИПРЖР, Москва 2002

3. 3. Галушкин А.И., Кирсанов Д.В., Цифровые нейрочипы (специализированные цифровые БИС для нейрокомпьютеров)// Зарубежная радиоэлектроника 1999 №1, стр. 17-37.