В результате работы была разработана полноценная программа для повышения производительности игрового движка Unity, на примере повышения быстродействия игры Prime World.

Разработанное программное обеспечение успешно внедрено в реализацию, за время внедрения было установлено на компьютеры, с различной аппаратной конфигурацией, более 700 пользователей, которые оценили повышение производительности.

В дальнейшем планируется учитывать больший объем исходных файлов проектов на Unity, с целью выявления возможных оптимизаций конфигурационных файлов для еще большего повышения производительности проектов.

1. Подробная информация о возможностях игрового движка и его технические характеристики [Электронный ресурс]. 2019. URL: <https://unity3d.com/unity> (дата обращения 17.11.2019).
2. Оптимизация производительности игры параметрами импорта звука Unity [Электронный ресурс]. 2019. URL: <https://habr.com/ru/post/437474/> (дата обращения: 22.11.2019).
3. Особенности, преимущества и недостатки движка Unity [Электронный ресурс]. 2019. URL: <https://cubiq.ru/dvizhok-unity/> (дата обращения: 27.11.2019).
4. Джефф Кент, С++ Основы программирования - М.: NT Press Москва, 2013. - 365 с.
5. Макс Шлее, Qt 5.10 Профессиональное программирование на С++ - М.: БХВ-Петербург, 2018. - 1051 с.

Синицын С.В., Семикина Е.В., Долбилов А.Д.

Повышение эффективности автоматических систем управления за счет упрощения программно-алгоритмического обеспечения

*Филиал ФГКВУ ВО «Военная академия ракетных войск стратегического назначения имени Петра Великого» Министерства обороны РФ
(Россия, Серпухов)*

doi:10.18411/sciencepublic-12-12-2019-08

idsp: sciencepublic-12-12-2019-08

Введение и обоснование актуальности упрощения программно-алгоритмического обеспечения автоматических систем управления.

Традиционно выделяют два основных направления обеспечения эффективности автоматических систем управления (АСУ): аппаратное – за счет применения современной элементной базы и программно-алгоритмическое – за счет реализации эффективного программного обеспечения (ПО). Как правило, в настоящее время, первое направление реализуется посредством разработки и изготовления финальных изделий российского производства из импортной элементной базы и на оборудовании ведущих западных и китайских фирм. Приоритет этого направления обоснован недостаточным развитием российского промышленного производства изделий отечественной микроэлектронной промышленности. Попытки производителей изготовить электронное устройство автоматического управления на основе российской элементной базы ведет к увеличению его стоимости в несколько раз, что априори коммерчески неэффективно. Например [1], стоимость отечественного 32-х разрядного микроконтроллера K1921BK01T в разы превышает стоимость его аналога LM4F132, а если учесть стоимость средств отладки, необходимых для разработки изделий, то разница в цене становится существенней.

Недостатком применения импортной микроэлектроники является невозможность изготовления на ее основе АСУ образцов вооружения и военной техники, а так же продукции двойного назначения. Поэтому изготовление такой продукции на основе отечественной микроэлектроники остается актуальным направлением, которое в настоящее время подчеркивается программой «импортозамещения».

Анализ современной продукции российской микроэлектроники показывает, что отечественные предприятия, приобретая лицензию на изготовление процессора какого либо типа, ориентируют производство отечественных микропроцессоров и микроконтроллеров не на развитие периферии, а на повышение их спецстойкости.

В таких условиях коммерческий успех отечественной продукции на основе российской микроэлектроники, может быть обеспечен не только аппаратными средствами, но и разработкой эффективного ПО, реализующего оптимальные алгоритмы. При этом эффективность ПО должна рассматриваться не только с позиции математической оптимальности алгоритмов, но и предусматривать оптимальность их аппаратной реализации по показателям быстродействия, точности или объема требуемой памяти.

Общие подходы к постановке задач упрощения алгоритмов и ожидаемый эффект от их решения.

Как правило, разработка или выбор разработанного алгоритма цифровой АСУ проводится на основе математической модели объекта управления. Такие алгоритмы записаны в принципиально простой форме, математически формализованы и являются оптимальными по выбранному критерию качества (например, минимум целевой функции или обеспечение сходимости при заданных исходных данных).

Оптимальные, с математической точки зрения алгоритмы, не всегда разрабатываются с учетом их реализации на цифровых ЭВМ. Поэтому, программное обеспечение, реализующее математически оптимальный алгоритм, может иметь необоснованно низкое быстродействие АСУ, большой расход памяти, низкую точность вычислений. Обычно повышенные значения показателей компенсируются применением процессора большей производительности, объема памяти и разрядности, что ведет к повышению стоимости цифровой АСУ и снижению её конкурентоспособности.

Анализ алгоритмов, реализуемых в цифровых системах управления летательными аппаратами, показал потенциальные возможности их оптимизации по рассматриваемым показателям. В основе такой оптимизации лежит упрощение математически оптимального алгоритма или упрощение математической модели объекта управления по критерию числа арифметических операций, выполняемых процессором. Сокращение числа арифметических операций, выполняемых процессором, ведет: во-первых, к сокращению времени выполнения алгоритма и повышению быстродействия, во-вторых, к уменьшению объемов требуемой памяти и в третьих, к повышению точности вычислений. Обратной стороной такого упрощения является возможная потеря логической связи формальной записи математической модели объекта управления с физическими принципами его функционирования.

Упрощение математической модели, реализуемой алгоритмом, должно проводиться на основе принципов валидации и верификации [2], сущность которых заключается в проведении процедур подтверждения адекватности упрощенного алгоритма объекту моделирования и его соответствия исходному с требуемой точностью. Основой обеспечения таких процедур являются следующие способы упрощения программно-алгоритмического обеспечения АСУ: применение адекватных математических методов к ММ, реализуемой алгоритмом; замена алгоритмических структур аналогичными; замена сложных вычислительных процедур в широком диапазоне условий совокупностью простых арифметических операций с сокращением диапазона условий в рамках допустимых значений.

Пример упрощения алгоритма и оценка его эффективности.

В цифровых системах управления при вычислении среднего значения N_{cr}^m некоторого параметра по результатам измерений $N_j, j = \overline{1, \dots, m}$, для ослабления (сглаживания) влияния случайных помех измерений на искомый результат, может применяться дискретный фильтр по методу наименьших квадратов (МНК) [3] вида

$$N_{cp}^m = \frac{12}{m(m^2 - 1)} \cdot \sum_{i=1}^m \left[\left(i - \frac{m+1}{2} \right) \cdot \sum_{j=1}^i N_j \right], \quad (1)$$

где m – общее число измерений параметра N .

Алгоритмическая реализация математической модели дискретного фильтра по МНК (1) предполагает использование алгоритмической структуры вложенного цикла, что ведет к существенным вычислительным затратам. Таким образом, математическая модель (1) оптимальна по МНК, но не адаптирована к реализации на ЭВМ.

В работе [4] показано, что ММ (1) посредством адекватных математических преобразований может быть преобразована к виду

$$N_{cp}^m = -\frac{6}{m(m^2 - 1)} \sum_{i=0}^{m-1} i \cdot (i - m) \cdot N_{i+1} \quad (2)$$

Анализ выражения (2) показывает, что вычисление среднего значения по МНК может быть реализовано на ЭВМ с применением простой циклической алгоритмической структуры.

Сравнительный анализ выражений (1) и (2) показывает, что для реализации вычислений параметра N_{cp}^m по исходному алгоритму требуется массив ячеек памяти для хранения результатов измерения параметра N_i , $i = 1, 2, \dots, m$, содержащий m элементов (обозначим его $MN[m]$), и две ячейки памяти: S_1 для хранения суммы внешнего цикла и S_2 для хранения суммы внутреннего цикла. Всего – $m+2$ ячейки памяти. Для реализации вычислений по упрощенному алгоритму требуется две ячейки памяти: S_1 для хранения суммы и S_2 – ячейка для промежуточных вычислений. Таким образом, применение упрощенного алгоритма позволяет сократить объем используемой памяти не менее чем в $2m$ раз.

Оценку повышения быстродействия реализации исходного и упрощенного алгоритмов проведем по числу тактов работы ЭВМ необходимых для расчета среднего периода следования импульсов. Пусть операции загрузки, сложения, вычитания выполняются за один такт работы ЭВМ, а операция умножения выполняется за 10 тактов. Для обозначенных условий программная реализация ММ (1) требует $(m^2+47)/2$ тактов работы ЭВМ, а по упрощенному алгоритму $23m$ тактов. Отношение этих выражений показывает, что реализация расчета среднего значения параметра, по упрощенному алгоритму требует в $(m/46+1)$ раз меньше тактов работы ЭВМ.

Заключение.

Упрощение алгоритмов на основе принципов валидации и верификации является эффективным способом повышения качества ПО АСУ. Упрощенные алгоритмы АСУ могут быть реализованы на микропроцессорах меньшей производительности, что ведет к снижению стоимости конечного изделия и, как следствие, к повышению его конкурентоспособности. При равных аппаратных условиях, реализация упрощенных алгоритмов ведет к повышению быстродействия, сокращению объема памяти и к повышению точности вычислений за счет сокращения числа арифметических операций. В свою очередь, освободившиеся вычислительные ресурсы могут быть использованы для расширения функциональных возможностей АСУ программными методами, например для реализации диагностических алгоритмов.

1. Каталог продукции 2018 г. НИИЭТ. Росэлектроника. г. Воронеж.
2. ГОСТ Р57412 – 2017. Компьютерные модели в процессах разработки, производства и эксплуатации изделий. Общие положения. – М.: Стандартинформ, 2017. – 20 с.
3. Аксенов В.Е. Надежность систем управления ракет. Монография. ч.2. / В.Е.Аксенов. – Серпухов: ФВА РВСН. 2015. – 315 с.
4. Синицын С.В., Пискулин Е.В. Упрощенный алгоритм масштабирования акселерометра. Известия ИИФ. 2019. №3. С.48-52.