

ИНСТРУМЕНТАЛЬНАЯ ПЛАТФОРМА ОЦЕНКИ ВЫЧИСЛИТЕЛЬНОЙ УСТОЙЧИВОСТИ АЛГОРИТМОВ РАБОТЫ БОРТОВЫХ КОМПЛЕКСОВ ЛА

Светлов Р.В.

ООО «Экспериментальная Мастерская НаукаСофт»,
Россия, г. Москва, ул. Годовикова, 9, стр.1

svetlovroman@gmail.com

Аннотация: Раскрыты особенности построения и использования инструментальных средств оценки вычислительной устойчивости алгоритмов работы роботизированных бортовых комплексов ЛА, в которых источником вычислительной неустойчивости служат операции традиционной машинной арифметики, не удовлетворяющие в общем случае требованиям ассоциативности, коммутативности и дистрибутивности.

Ключевые слова: вычислительная (не)устойчивость, бортовые комплексы, ассоциативные машины Тьюринга, масштабируемая разрядная сетка.

Введение

Использование в современных бортовых комплексах алгоритмов цифровой обработки сигналов и изображений сопряжено с потерей *вычислительной устойчивости* [1,2], а с ней и устойчивости контуров управления ЛА. Этот эффект обусловлен ограниченной разрядной сеткой арифметико-логических устройств в силу чего выполняемые ими арифметические операции в общем случае не ассоциативны, не коммутативны и не дистрибутивны. Тем не менее этим обстоятельством пренебрегают как при переходе от аналитической формы записи алгоритмов к вычислительной, так и при динамическом управлении вычислительными ресурсами в многопроцессорных и многоядерных бортовых вычислительных системах, что чревато *нарушением отношения эквивалентности* между исходным аналитическим и исполняемым вычислительным алгоритмом.

Подобного рода нарушения способны непрогнозируемым образом привести к потере управления ЛА после выполнения им маневров, предусмотренных регламентом эксплуатации, последствия которых, обычно, не обосновано списывают на действия экипажа.

Полностью устранить подобного рода эффекты работы бортовых комплексов управления ЛА практически невозможно, так как для этого требуются «вековые» летные испытания, которые включают не только отдельные режимы работы бортового оборудования, но и всевозможные переходы из одного режима в другой.

В таких условиях можно с помощью модельных экспериментов исследовать и оценить вычислительную устойчивость алгоритмов и на этой основе выработать рекомендации по предотвращению ее нарушений. Но для проведения модельных экспериментов требуется либо «эталонный» вычислитель, эквивалентный бортовому по архитектуре, либо его симулятор, который преодолевает ограничения конечной разрядной сетки имеющейся аппаратуры и восстанавливает ассоциативные, коммутативные и дистрибутивные свойства арифметических операций.

Цель работы: раскрыть особенности построения и использования программных симуляторов алгоритмически ориентированных «эталонных» вычислителей, которые позволяют оценить вычислительную устойчивость алгоритмов работы бортовых комплексов на этапе создания и ввода в эксплуатацию ЛА.

1 Ассоциативная машина Тьюринга как формальная основа поддержки масштабируемой разрядности

Для восстановления базовых свойств арифметических операций, а с ними и точности их выполнения в работе используется *символьный формат* вещественного числа (*Symbol*), который неограничен по разрядности мантиссы и порядку и имеет структуру {«знак»; «порядок»; «мантисса»; «разрядность»}, где поле:

- «знак» определяет знак числа и с целью экономии памяти представлен как булево (*bool*) число (*true* – «плюс», *false* – «минус»);
- «порядок» определяет порядок вещественного числа в нормализованном виде, который представлен как целое число (*int*);
- «мантисса» имеет вид последовательности символов в форме строки (*string*), длина которой ограничена только объемом памяти используемого компьютера.
- «разрядность» указывает «плавающую» длину строки мантисс и представлена целым числом (*int*).

Этот формат позволяет:

- произвольно масштабировать разрядность преобразуемых операндов, а сами арифметические операции выполнять в «нечисленном» операционном базисе подстановки выходного символа паре входных символов, как это делает ассоциативная машина Тьюринга (рисунок 1);
- каждой выполняемой процессором инструкции (I) ставить в соответствие свойственную только ей таблицу подстановки выходных символов, каждая из которых поддерживается универсальным механизмом ассоциативного доступа по содержимому входных символов (a_i, b_i) ($A_n = (a_{n-1}, \dots, a_i, \dots, a_0)$, $B_n = (b_{n-1}, \dots, b_i, \dots, b_0)$).

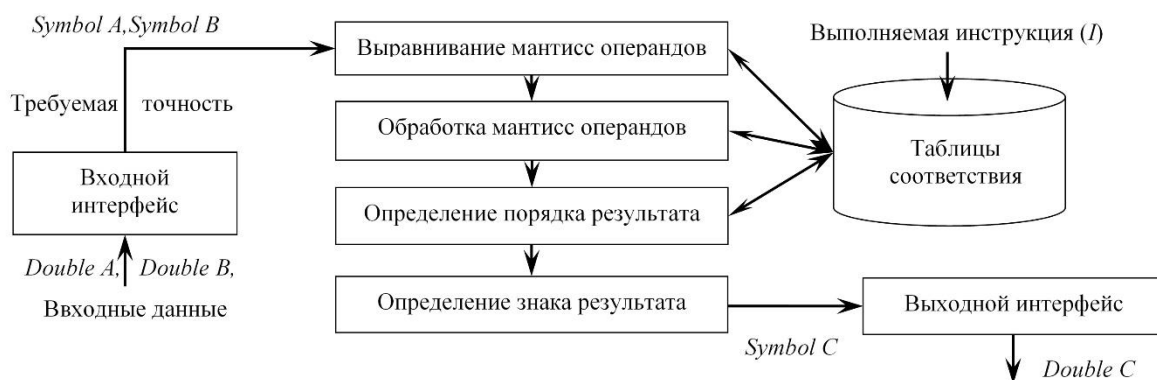


Рис. 1. Общая схема функционирования ассоциативной машины Тьюринга

Главные особенности вычислительных технологий на основе ассоциативных машин Тьюринга:

1. Если классические машины Тьюринга абстрактны и в основном используются в теории вычислений, то машины Тьюринга, работающие в ассоциативном операционном базисе, реализуемы на практике в виде машин символьной обработки данных.

2. Практическая ценность ассоциативных машин Тьюринга состоит в том, что в них эффект «бесконечной» ленты достигается за счет представления всех операндов в виде *списков*, что повышает точность вычислений до произвольно задаваемой величины. Это позволяет, в частности, использовать такие машины в качестве *эталонных ЭВМ*, контролирующих вычислительную устойчивость алгоритмов, предназначенных для использования в классических ЭВМ с ограниченной разрядной сеткой.

3. Гибридные вычислительные технологии с ассоциативными субпроцессорными трактами [3] позволяют достичь компромисса между используемыми на микропрограммном уровне ассоциативными методами и средствами «нечисленного» преобразования информации и вычислительными технологиями, отработанными в классических ЭВМ, начиная с ассемблерного уровня организации вычислений.

4. Любая задача, решаемая классическими ЭВМ, представима в операционном базисе ассоциативной выборки и подстановки символов из некоторого конечного алфавита. Другими словами, если абстрагироваться от громоздкости описания, то ассоциативного операционного базиса достаточно для формального описания алгоритма решения любой вычислительной задачи, если такое решение существует.

5. Для аппаратной реализации ассоциативной машины Тьюринга требуется только ассоциативная память, поддерживающая работу с таблицами подстановки, а также механизмы хранения и обработки списков данных на основе двух базисных операций: «сравнение» и «подстановка», что можно осуществить на физико-техническом уровне организации вычислений, как это имеет место в быстрой одноквантовой логике [4].

2 Инженерная методика оценки алгоритмов цифровой обработки

Необходимость разработки методики оценки вычислительной устойчивости обусловлена следующими обстоятельствами.

1. В теории вычислений [1,2] *вычислительная устойчивость* оценивается неравенством $\Delta y < \Delta x$, где Δx – абсолютная *погрешность представления* входных переменных, Δy – абсолютная *погрешность вычисления* зависимости $y=f(x)$. В идеале это неравенство выполняется, если вычислитель работает с нулевой погрешностью. При этом специфика механизмов поддержки этого

неравенства не учитывается, в то время как она играет решающую роль при выборе *вычислительного алгоритма*, по максимуму использующего операционный и коммутационный ресурс параллельных вычислительных систем, встроенных в бортовые комплексы.

2. Аппаратно-временные затраты на поддержку масштабируемой разрядной сетки вынуждают оценивать вычислительную устойчивость не всего алгоритма, а только «критического пути» в его граф-потоке [1], который, в основном, определяется кратностью (v) последовательно выполненных операций умножения, которая приводит к пропорциональному росту разрядности результирующих операндов $m = v^*n$.

3. Задача оценки вычислительной устойчивости эквивалентных по своему функциональному назначению алгоритмов усложняется двумя взаимно обусловленными обстоятельствами – из-за наличия операторов условных переходов последовательность команд *упорядочена только частично* и не стабильна на разных циклах решения задачи, а исполняемые процессором арифметические команды не ассоциативны, не дистрибутивны и не коммутативны.

Совокупность приведенных обстоятельств вынуждает разрабатывать и утверждать *индивидуальную методику* проведения государственных испытаний и оценки точности, вычислительной устойчивости для *каждого типа* бортового комплекса с встроенными процессорами. В «метрологической плоскости проекта» это автоматически вынуждает рассматривать все процессоры как алгоритмически ориентированные.

Разработанная методика нацелена на избирательную оценку вычислительной устойчивости операторов условных переходов, как главных индикаторов потери управления бортовыми роботизированными комплексами. Она регламентирует такой порядок ее проведения, который позволяет найти баланс между точностью вычислений и темпом реального времени, который диктуется регламентированными «сценариями» производства полетов ЛА.

Выбрать *вычислительный алгоритм*, удовлетворяющий требованиям темпа реального времени с минимальным числом «вложений» операций умножения и деления, критичных для точности выполнения арифметических преобразований.

На основе выбранного вычислительного алгоритма дифференцированно определить вычислительную (операционную и коммутационную) нагрузку, которая диктуется темпом реального времени решения задачи.

Выбрать параллельную архитектуру и параметры вычислителя, способного реализовать вычислительную нагрузку в реальном времени.

Оценить влияние различных источников погрешностей на конечный результат, то есть на вычислительную устойчивость всего программно-аппаратного и алгоритмического комплекса, которая определяется *стабильностью реакции операторов условных переходов* на одни и те же входные данные.

Эта методика не лишена двух внутренних «метрологических» противоречий, вытекающих из требований пункта 1, который предполагает:

а) преобразование формальных моделей предметной области с использованием «скобочных форм», эквивалентных только при выполнении требований ассоциативности, коммутативности и дистрибутивности составляющих операций, что выполнимо не в каждом формате машинной арифметики;

б) при наличии в вычислительном алгоритме циклов по условию число «вложенных» операций умножения и деления зависит от содержимого преобразуемых данных и трудно предсказуемо.

Последнее иллюстрирует таблица 1, из данных которой следует, что в формате *double* за устойчивость вычислительного алгоритма можно бороться при наличии в нем не более 3–5 последовательно выполненных операций умножения, так как при 7 таких операциях он теряет вычислительную устойчивость даже по теоретическому критерию [1,2].

Одна из центральных «метрологических» проблем оценки вычислительной устойчивости алгоритмов, использующих операции векторно-матричной алгебры, состоит в том, что при этом необходимо распространить результаты оценки погрешностей составляющих элементов на весь результирующий вектор или матрицу, которые могут быть как одной, так и разной модальности.

В таких условиях вычислительную устойчивость матричного умножения можно характеризовать (рисунок 2), либо по элементу результирующей матрицы с максимальной абсолютной погрешностью, либо по средневзвешенной абсолютной погрешности в зависимости от степени ее физической «однородности».

Из данных рисунка 2 следует:

- средневзвешенная абсолютная погрешность является более оптимистичной, так как при 16-битных входных данных по теоретическому критерию она не выходит за пределы погрешности представления в $1,05E-6 < 7,63E-6$ даже при размерности результирующей матрицы $50*50$;
- оценка по элементу с максимальной абсолютной погрешностью удовлетворяет условию вычислительной устойчивости $\Delta y < \Delta x$ только до размерности результирующей матрицы меньше $20*20$.

Таблица 1. Абсолютная погрешность вычисления последовательности произведений (порядок не более 5)

Длина цепочки умножений	Значение в double	Значение в symbol	Абсолютная Погрешность
1	7,11851934305098	+7,11851934305098508564228205591	5,086E-15
2	1332484,9374608	+1332484,93746080090895440923298783328472261452	9,090E-10
3	114146994627,465	+114146994627,465192986758588062103385024078813563550176029024	1,930E-04
5	1273881183,36193	+1273881183,361937503351346096269562971781802293031623924947106747991336262664993149464768	7,503E-06
6	2851077617863,45	+2851077617863,47438249633663353163572488491262860666097347769590067097126197808484669600971811173779608	2,438E-02
7	8,29389102152345E+19	+82938910215234128918,763652290153022379031465455079935732949030526878830431692126561120933773010440983949721825116922	-3,711E+05

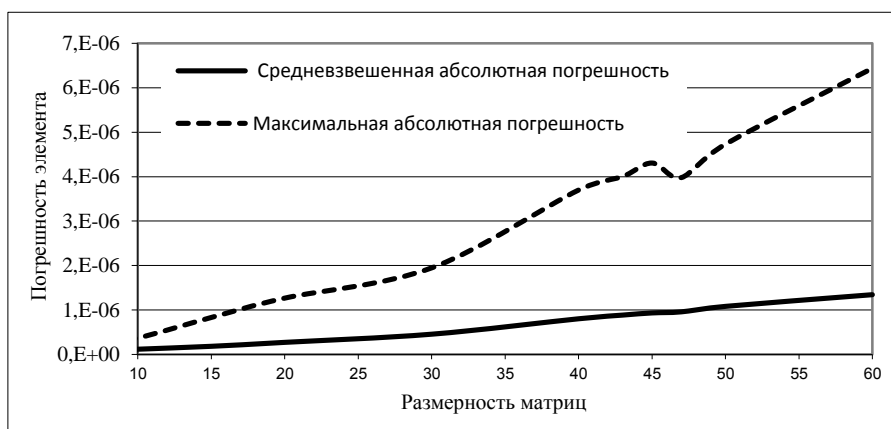


Рис. 2. Средневзвешенная и максимальная абсолютная погрешности вычисления элементов произведения матриц (float)

3 Оценка влияния основных факторов, определяющих интенсивность проявления источников погрешностей вычислений

Изложенная методика оценки вычислительной устойчивости состоятельна, если в эмуляторах, работающих в символьном формате, имеется инструментарий восстановления ассоциативного, коммутативного и дистрибутивного свойств арифметических команд формата *symbol*.

При этом основным источником погрешности считается округление вычисленных значений промежуточных операндов, которое сопровождает процедуры денормализации и пересылки операндов из аккумуляторов арифметико-логических устройств во внутренние или буферные регистры обмена данными с памятью или другими процессорами.

3.1 Фактор упорядоченности потока входных данных

Эффект потери коммутативного свойства суммирования данных в формате *double* и его восстановление в формате *symbol* иллюстрирует таблица 2 и график рисунка 2, где приведены

результаты сравнительной оценки точности однопроцессорного суммирования при *разном упорядочении* последовательности операндов, значения которых ограничены порядком не более 10.

Здесь под дельта-разницей понимается отклонение вычисленного значения не от истинного, а от наиболее точного, которое принимается за эталонное.

Приведенные эмпирические данные позволяют заключить:

1. Значения сумм в формате *symbol* не зависят от порядка суммирования операндов, что говорит о восстановлении *коммутативного свойства* операции ADD_s .

2. Отклонение накопленной суммы от результата, полученного при упорядочении суммируемых операндов по возрастанию, при достаточно широком диапазоне изменения значений суммируемых операндов (порядок равен 10) может отличаться даже в первом разряде до запятой от условно «истинного» значения, представленного дельта-разницей с результатом формата *symbol*.

Таблица 2. Сравнительная точность однопроцессорного суммирования в форматах *double* и *symbol* при разном порядке поступления операндов порядка не более 10.

Кол-во эл-тов	Тип упорядочения	Значение суммы (double)	Значение суммы (symbol)	Дельта-разница от значений double по возрастанию	Отклонение от symbol
1000	Хронология	55614748574,5503	+55614748574,55028854052 927918	0,00000E+00	-1,14595E-05
	Возрастание	55614748574,5503	+55614748574,55028854052 927918	0,00000E+00	-1,14595E-05
	Убывание	55614748574,5502	+55614748574,55028854052 927918	3,81470E-04	8,85405E-05
10000	Хронология	583655220920,491	+583655220920,4932458167 54658368	2,56348E-03	2,24582E-03
	Возрастание	583655220920,494	+583655220920,4932458167 54658368	0,00000E+00	-7,54183E-04
	Убывание	583655220920,496	+583655220920,4932458167 54658368	2,19727E-03	-2,75418E-03
50000	Хронология	2812925078375,14	+2812925078375,122247048 442802934	1,36719E-02	-1,77530E-02
	Возрастание	2812925078375,12	+2812925078375,122247048 442802934	0,00000E+00	2,24705E-03
	Убывание	2812925078375,09	+2812925078375,122247048 442802934	3,56445E-02	3,22470E-02

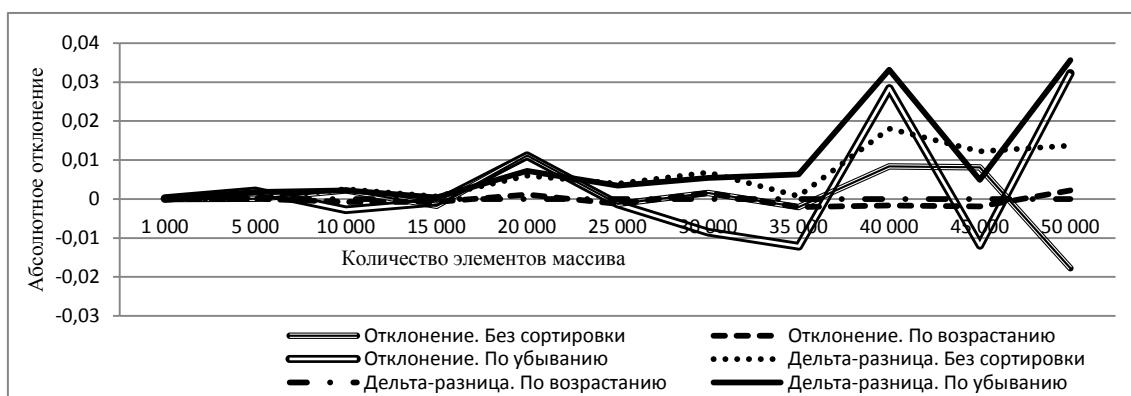


Рис. 3. Точность однопроцессорного суммирования массива чисел формата *double* при разном упорядочении операндов порядка не более 10

3. Отклонения накопленной суммы, полученные в формате *double*, хорошо коррелируют по минимумам и максимумам зависимости от количества суммируемых операндов с дельта-разницей, полученной в формате *symbol*.

Проецируя эти эффекты на предметную область, можно сказать:

а) бортовые вычислители в общем случае работают с разной погрешностью при маневрах, сопряженных с ростом и уменьшением высотно-скоростных и угловых параметров их движения,

причем вычислительная устойчивость бортовых комплексов существенно зависит от выполняемого ЛА маневра, когда угловые и/или высотно-скоростные параметры либо плавно нарастают, либо плавно убывают;

б) если алгоритм оказался вычислительно устойчивым по отношению к формату *double*, то с определенной численной поправкой он объективно вычислительно устойчив по отношению к формату *symbol*, но не наоборот.

3.2 Фактор роста числа пересылок данных при распараллеливании вычислений

Потерю *дистрибутивного свойства* суммирования данных в формате *double* и его восстановление в формате *symbol* иллюстрирует график рисунка 4, где приведена зависимость «точности» параллельного вычисления суммы, понимаемой как дельта-разница между *double* и *symbol* (порядок не более 10) при разных коэффициентах распараллеливания и с разным упорядочением суммируемых операндов.

Эти эмпирические данные позволяют заключить:

1. Оценка вычислительной устойчивости параллельных алгоритмов над данными формата *double* и порядком не более 10 при хронологическом порядке суммирования дает (за редким исключением) *систематическое смещение* по отношению к оценке, полученной в формате *symbol*.

2. Колебательный характер оценок вычислительной устойчивости параллельных алгоритмов и связанные с ним *артефакты* в форматах *double* и *symbol* обусловлены *содержимым* суммируемых данных и накопленных сумм, которое делает «индифферентным» практически любой сдвиг вправо младшего операнда во время его денормализации. Например, в формате плавающей запятой любой сдвиг вправо операнда с десятичным значением 0,5 на 22 бита в формате *float* или на 51 бит в формате *double* не изменяет его содержимое, так как отбрасываются «нулевые» разряды.

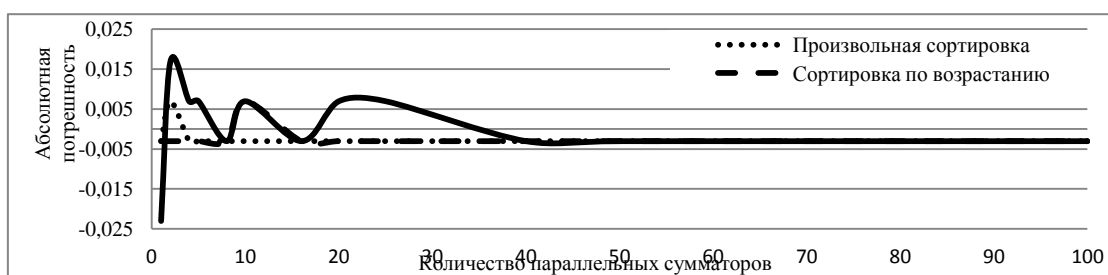


Рис. 4. Точность параллельного суммирования массива из 50000 чисел формата *double* порядка не более 10

3. Если по каким-то причинам суммируемые операнды оказались упорядочены по убыванию, то оценка вычислительной устойчивости параллельного алгоритма в формате *double* становится не состоятельной.

4. Снижение динамического диапазона изменения численных значений слагаемых до порядка, равного 5 (рисунок 5) делает состоятельной оценку вычислительной устойчивости, полученной в формате *double*, только для входных данных, представленных с абсолютной погрешностью не более 10^{-7} , что соответствует 23-битным входным аналого-цифровым преобразователям, удовлетворительно решающим задачи картографирования местности.

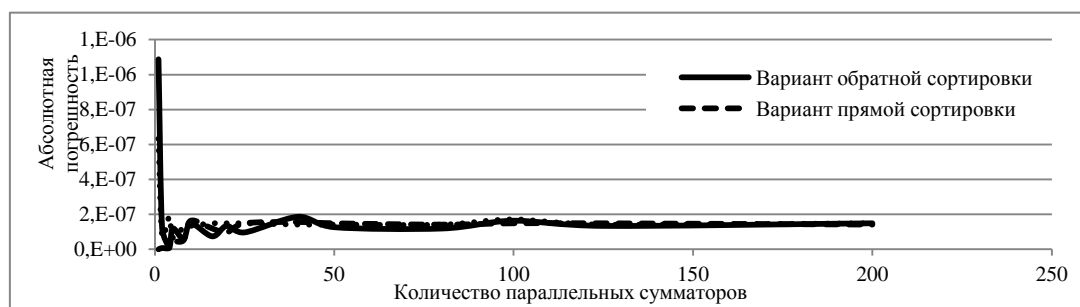


Рис. 5. Точность параллельного суммирования массива из 50000 чисел формата *double* с порядком не более 5

5. Если же входные данные системы получены 32-битным аналого-цифровым преобразователем, а для поддержания темпа реального времени требуется распараллелить вычисление в пределах 5÷50 (см. рисунок 5), то оценку ее вычислительной устойчивости, надо проводить с помощью программных эмуляторов, работающих в формате *symbol*.

3.3 Взаимосвязь факторов снижения вычислительной устойчивости интеллектуальных бортовых вычислительных систем и маневренностью ЛА

Точность определения собственного положения ЛА на местности в изменяющихся суточных, погодных и климатических условиях является решающим фактором, от которого зависит эффективность, а в ряде случаев и целесообразность использования авиации для экомониторинга земной и водной поверхности, оценки природных ресурсов, разведки полезных ископаемых, проведения поисково-спасательных работ и т.п.

Одним из путей решения этой проблемы является использование стереоскопических навигационных систем [5], в работе которых одновременно используются методы и алгоритмы *оценивания, идентификации и проверки гипотез* при обработке в реальном времени стереопар изображений достаточно больших участков местности. Поэтому успех практического использования таких систем во многом определяется производительностью и вычислительной устойчивостью бортовых вычислительных систем.

Проведенные с помощью симулятора ассоциативной машины Тьюринга модельные исследования производительности и вычислительной устойчивости таких систем показали [6]:

1. Требуемая производительность вычислений достижима как в цифровых процессорах обработки сигналов с плохо масштабируемой разрядностью, так и в хорошо масштабируемых архитектурах на основе *FPGA*, которыми комплектуются самолеты 5-го поколения США, создаваемые в рамках программы *Joint Strike Fighter*. Но для высокоточного определения собственных координат и эти самолеты должны стабилизировать угловые и высотно-скоростные параметры полета на период определения собственного положения на местности, а их бортовые вычислительные системы должны работать с повышенной точностью не ниже формата *double*.

2. Из-за плохой масштабируемости разрядной сетки повышение коэффициента распараллеливания вычислений свыше 4 может привести к потере вычислительной устойчивости бортовых вычислителей стереоскопических навигационных систем на цифровых процессорах обработки сигналов.

Заключение

Полученные эмпирическим путем данные позволяют заключить:

1. Вычислительная устойчивость алгоритмов работы роботизированных бортовых комплексов является не абсолютной, а *относительной* и она зависит от разрядности мантисс и динамического диапазона изменения порядка входных операндов, порядка их поступления на обработку, состава и порядка выполнения арифметических преобразований и коэффициента распараллеливания вычислений. Все эти факторы хорошо коррелируют с высотно-скоростными параметрами и динамикой изменения траекторных и угловых параметров полета ЛА, что позволяет на основе полученных оценок вычислительной устойчивости алгоритмов управления обоснованно выработать предложения по ограничению не только отдельных маневров, но и их последовательностей.

2. Стратегия симуляции алгоритмически-ориентированных ассоциативных машин Тьюринга с произвольно масштабируемой разрядностью операндов кардинально снижает размерность задачи оценки вычислительной устойчивости алгоритмов работы бортовых комплексов ЛА за счет:

- анализа не всех, а только критических путей в граф-потоках алгоритмов, выбор которых определяется количеством операций умножения в цепочке;
- использованием в качестве ключевого индикатора операторов условных переходов, в которых численные нарушения пороговых условий приводят к инверсии «истины» в «ложь» и наоборот.

3. Рост интеллектуальности задач, решаемых бортовыми комплексами, неизбежно приводит к росту коэффициента распараллеливания вычислений, что автоматически увеличивает количество операций межпроцессорной пересылки данных. В результате возрастает интенсивность округлений значений промежуточных операндов, которая снижает вычислительную устойчивость реализуемых алгоритмов.

Литература

1. *Фейлмейер М.* Параллельные численные алгоритмы. Системы параллельной обработки: Пер. с англ. /Под ред. Д. Ивенса – М.: Мир, 1985, с. 285–337.
2. *Панкратьев Е.В.* Элементы компьютерной алгебры. Изд. ООО «ИНТУИТ.ру», ООО «БИНОМ. Лаборатория знаний». Москва. 2007.
3. *Алакоз Г.М., Котов А.В., Курак М.В., Попов А.А., Сериков А.П.* Вычислительные наноструктуры /Под редакцией д.т.н., профессора кафедры Г.М. Алакоза, т.1. Задачи, модели, структуры, т.2. Программно-аппаратные платформы. – М.: «ИНТУИТ–БИНОМ», 2009.
4. Быстрая одноквантовая логика. <http://www.intuit.ru/studies/courses/12176/1169/lecture/24917>
5. *Белоглазов И.Н.* Синтез алгоритмов стереоскопической системы навигации, наведения и дистанционного зондирования местности //Теория систем управления: Изв. РАН, №5, 2009, с.21–33.
6. *Алакоз Г.М., Белоглазов И.Н., Коллеганов М.М., Светлов Р.В.* Эмпирическая оценка вычислительной устойчивости и отказоустойчивости бортовых вычислителей стереоскопических навигационных систем //Информационно-измерительные и управляющие системы, №8, 2010, с.26–38.