

Параллельный алгоритм моделирования управляемого движения космической тросовой системы

В.В. Жидченко¹, А.Н. Коварцев¹

¹Самарский национальный исследовательский университет им. академика С.П. Королева, Московское шоссе 34А, Самара, Россия, 443086

Аннотация. Моделирование управляемого движения космической тросовой системы существенно отличается от решения аналогичной задачи для свободного движения. Процесс развертывания троса увеличивает его длину и сопровождается добавлением новых точек в дискретную модель троса. Одновременно в модель добавляются уравнения, описывающие движение точек троса. При этом расширение модели должно производиться с условием сохранения основных законов механики. Выпуск троса реализуется под контролем системы регулирования, управляющей развертыванием троса на заданную длину. Это приводит к необходимости решения задачи интегрирования системы дифференциальных уравнений с переменным и возрастающим количеством уравнений. В настоящей работе предложена модель параллельного алгоритма моделирования управляемого движения троса, которая с точки зрения ускорения незначительно уступает модели свободного движения. Разработанный параллельный алгоритм позволяет разбить трос на большое количество точек, приближающее данную модель к модели с распределенными параметрами. В результате становится возможным моделирование таких режимов движения троса, вычисление которых затруднительно при применении других методов. Произведена оценка ускорения параллельного алгоритма. Эффективность предложенного алгоритма измерена при проведении вычислительных экспериментов.

1. Введение

Исследование динамики космических тросовых систем (КТС) имеет большую практическую значимость в связи с широким спектром возможных применений КТС: исследование верхних слоев атмосферы, создание искусственной гравитации на космических аппаратах, транспортные операции в космосе и возвращение грузов с орбиты, изменение параметров орбит космических аппаратов, создание орбитальных электростанций, зондирование геомагнитного поля Земли, удаление космического мусора и т.п. [1,2,3].

Различают две разновидности движения КТС. При свободном движении два тела связаны тросом фиксированной длины. В задаче управляемого движения рассматривается изменение длины троса в соответствии с заданной функцией (программой) управления. Моделирование управляемого движения очевидно сложнее, поскольку программа управления влияет на длину и силу натяжения троса. Существуют различные модели управляемого движения КТС. В модели эластичного стержня трос рассматривается как прямой, негибкий, но растяжимый стержень [4]. В модели троса, представленного в виде жестких стержней, соединенных шарнирами (hinged-rod model), описывается гибкость троса, а также учитывается влияние

аэродинамических сил при движении троса в атмосфере [5, 6]. В модели бисера (bead model) трос представляется набором материальных точек с конечной массой, соединенных пружинами нулевой массы [2, 7, 8, 9]. Модель бисера в вычислительном плане проще модели жестких стержней, соединенных шарнирами, однако для достижения высокой точности необходимо большое количество точек, что увеличивает трудоемкость вычислений. По этой причине в большинстве работ, использующих модель бисера, ограничиваются несколькими десятками точек. С другой стороны, при большом количестве точек модель бисера превосходит другие модели в таких аспектах, как точное моделирование механизма разворачивания, учет неравномерной силы натяжения или электропроводности троса. Настоящая работа преследует цель устранения ограничений на количество точек при моделировании управляемого движения КТС с использованием модели бисера.

При моделировании протяженных тросовых систем, характеризующихся неравномерным распределением массовых характеристик системы и параметров окружающей среды в пространстве, используется аппарат дифференциальных уравнений в частных производных со сложными краевыми условиями. Сложность краевых условий вызвана наличием концевых тел, совершающих пространственные колебания, и переменностью длин тросов. Число уравнений в системе обыкновенных дифференциальных уравнений (ОДУ), описывающей динамическую модель КТС, может измеряться десятками тысяч. Это приводит к большим временным затратам при математическом моделировании движения тросовой системы на ЭВМ. Численное решение таких систем напрямую представляется затруднительным даже для современных вычислительных систем.

В настоящей работе представлены результаты разработки модели параллельного алгоритма численного интегрирования управляемого движения космической тросовой системы. Данная модель использует способ распараллеливания численного метода Рунге-Кутты четвертого порядка, сходный с предложенным в работе [10], однако рассматриваемая модель существенно отличается от модели свободного движения троса. Дело в том, что первоначально базовый космический аппарат (БКА) и малый космический аппарат (МКА) или спускаемая капсула (СК) связаны очень маленьким участком троса. По мере его разворачивания добавляются новые точки, а следовательно, и соответствующие уравнения, описывающие движение точек троса. При этом возникает проблема, связанная с организацией добавления новой точки при условии сохранения основных законов механики. Кроме того, выпуск троса реализуется под контролем системы регулирования, управляющей разворачиванием троса с БКА на заданную длину. Все это приводит к необходимости решения задачи интегрирования ОДУ с переменным и возрастающим количеством уравнений. Существенно изменились концепции модели параллельного алгоритма для реализации управляемого движения КТС. В результате была создана новая оригинальная модель параллельного алгоритма интегрирования управляемого и свободного движения троса, которая с точки зрения ускорения незначительно уступает модели параллельного алгоритма свободного движения троса.

2. Математическая модель управляемого движения космической тросовой системы

Рассмотрим задачу разворачивания троса. При построении математической модели КТС рассматривается как механическая система, состоящая из совокупности n материальных точек, соединенных упругими невесомыми связями (рисунок 1).

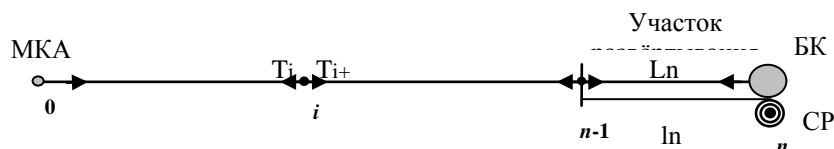


Рисунок 1. Схема разворачивания троса.

Разворачивание троса происходит с базового космического аппарата. Управление разворачиванием троса реализуется с помощью системы регулирования (СР) по заданной программе выпуска троса с использованием механизма торможения троса, обеспечивающего

практическую реализацию развертывания троса в соответствии с заданной программой. На начальном участке развертывания существуют две материальные точки: БКА и малый космический аппарат (МКА). После развертывания троса на заданную длину участка троса происходит фиксация нового участка троса, т.е. появляется новая материальная точка. Затем на участке троса, прилегающем к БКА, с минимальной длиной производится следующий цикл развертывания троса. Процесс повторяется, пока трос не развернется на заданную длину.

2.1. Математическая модель движения внутренних точек КТС

Математическая модель движения внутренних точек КТС описывается системой ОДУ:

$$\begin{cases} \frac{dR_i}{dt} = V_i \\ \frac{dV_i}{dt} = -\frac{\mu_i}{r_i^3} R_i - \frac{F_i}{m_i L_{i-1}} (R_i - R_{i-1}) + \frac{F_{i+1}}{m_i L_i} (R_{i+1} - R_i) \end{cases}$$

$$R_i = [x_i, y_i, z_i]^T, V_i = [V_i^x, V_i^y, V_i^z]^T, r_i = |R_i|, L_i = |R_{i+1} - R_i|$$

$$F_i = \begin{cases} c \frac{L_{i-1} - l_{i-1}}{l_{i-1}}, & L_{i-1} > l_{i-1} \\ 0, & L_{i-1} \leq l_{i-1} \end{cases} \quad (1)$$

Здесь F_i – сила натяжения троса, определяемая по закону Гука для односторонней механической связи, μ – гравитационная постоянная Земли, R_i и V_i – координаты и скорость i -ой внутренней точки в геоцентрической неподвижной системе координат.

2.2. Математическая модель движения МКА

Малый космический аппарат взаимодействует с БКА только на первом этапе выпуска троса, пока его длина не достигла момента формирования новой точки. Математическая модель движения МКА описывается системой ОДУ:

$$\begin{cases} \frac{dR_0}{dt} = V_0 \\ \frac{dV_0}{dt} = -\frac{\mu}{r_0^3} R_0 - \frac{F_0}{m_0 L_0} (R_1 - R_0) \end{cases}$$

$$r_0 = |R_0|, L_0 = |R_1 - R_0| \quad (2)$$

2.3. Математическая модель движения БКА

Математическая модель движения БКА на этапе выпуска троса описывается следующей системой ОДУ:

$$\begin{cases} \frac{dR_n}{dt} = V_n \\ \frac{dV_n}{dt} = -\frac{\mu}{r_n^3} R_n + \frac{F_n}{m_{КА} L_n} (R_{n-1} - R_n) \end{cases}$$

$$r_n = |R_n|, L_n = |R_{n-1} - R_n|, F_n = c \frac{L_n - l_n}{l_n} \quad (3)$$

Величина нерастянутой длины троса (выпущенной) l_n определяется отдельно в соответствии с выпущенным участком, определяемым СР и описываемым отдельной системой ОДУ, описывающей работу механизма выпуска троса.

2.4. Модель работы механизма выпуска троса

Процесс развертывания троса описывается одномерной моделью выпуска троса, описываемой системой ОДУ:

$$\begin{cases} \frac{dl}{dt} = V_l \\ \frac{dV_l}{dt} = \frac{1}{m_u}(F_n - F_c) \\ F_c = p_l(\sum_{i=0}^n l_i - l_p) + p_v(V_l - V_{lp}) \end{cases} \quad (4)$$

Здесь l, V_l - длина первого участка троса, считая от БКА (метрическая, нерастянутая) и скорость выпуска троса соответственно, m_u - коэффициент, учитывающий инерционность механизма управления, F_c - сила сопротивления выпуску троса в механизме управления, F_n - сила упругости троса, действующая на БКА, l_p и V_{lp} - программные (номинальные) зависимости длины и скорости троса от времени, определяющие силу торможения, p_l, p_v - коэффициенты обратной связи; l_i - нерастянутые длины всех участков.

Номинальная программа развертывания КТС вычисляется из решения системы ОДУ:

$$\begin{cases} \frac{d\theta}{dt} = \omega_\theta \\ \frac{d\omega_\theta}{dt} = -2\frac{V_{lp}}{l_p}(\omega_\theta + \Omega) - \frac{3}{2}\Omega^2 \sin(2\theta) \\ \frac{dl_p}{dt} = V_{lp} \\ \frac{dV_{lp}}{dt} = l_p[(\omega_\theta + \Omega)^2 - \Omega^2(1 - 3\cos^2 \theta)] - \frac{T_0(l_p, V_{lp})}{m_0} \end{cases} \quad (5)$$

где Ω - угловая скорость орбитального движения базового КА по круговой орбите, m_0 - масса спутника; θ - угол отклонения троса от направления местной геовертикали КА; l_p и V_{lp} - номинальная длина и скорость развертывания троса, T_0 - сила упругости троса (программная); ω_θ - угловая скорость вращения троса относительно базового КА.

Номинальная (программная) сила упругости задается соответствующими динамическими законами. Например, можно рассмотреть закон, состоящий из двух этапов. Первый этап развертывания длится около 6000с. Номинальная программа развертывания в этом случае задается динамическим законом:

$$T_0(l_p, V_{lp}) = m_0\Omega^2(a l_p + b \frac{V_{lp}}{\Omega} - c L_k) \quad (6)$$

где $a=4.6, b=3.5, c=1.6$ - параметры закона управления; $L_k=3000$ м - конечная длина троса для первого этапа, $\Omega = 0.0011587c^{-1}$. Второй этап развертывания длится около 2165с. На этом этапе используется релейный закон управления:

$$\begin{cases} T_0 = T_{min}, t \leq t_n \\ T_0 = T_{max}, t > t_n \end{cases} \quad (7)$$

где T_{min}, T_{max} - параметры закона; t_n - время переключения; t - текущее время развёртывания. Выбором значений параметров закона можно обеспечить заданные конечные значения характеристик движения МКА. В данной работе использовались следующие значения: $T_{min}=0.02$ Н, $T_{max}=2.1$ Н, $t_n=7900$ с.

2.5. Алгоритм формирования новой точки

В процессе развертывания троса происходит процесс последовательной дискретизации задачи моделирования движения КТС. Предполагается, что в конечном итоге тросовая система должна состоять из N точек. Уровень дискретизации определяет точность расчетов и выбирается

заранее. Расстояние между точками при выбранном уровне дискретизации равно $l_0=L_T/(N-1)$. Здесь L_T - длина троса. Первоначально БКА и МКА связаны небольшим (стремящимся к нулю) участком троса. По мере выпуска троса, до достижения приблизительно заданной длины формируемого участка (l_0), возникает необходимость порождения новой точки. Введение нового элемента необходимо производить так, чтобы не нарушить основные законы сохранения механической системы до и после выделения новой точки. На рисунке 2 представлена схема добавления новой точки.

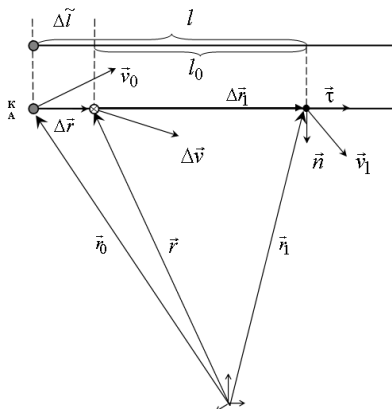


Рисунок 2. Схема добавления новой точки троса.

На рисунке 2 r_0 и v_0 - радиус-вектор и скорость КА, r_1 и v_1 - радиус-вектор и скорость формируемой точки троса в геоцентрической системе координат. Положение этой точки относительно БКА и ее относительная скорость находятся следующим образом:

$$\begin{cases} \Delta \vec{r}_1 = \vec{r}_1 - \vec{r}_0 \\ \Delta \vec{V}_1 = \vec{V}_1 - \vec{V}_0 \end{cases}$$

Положение новой точки относительно БКА определяется следующим образом:

$$\Delta \vec{r} = \frac{|\Delta \vec{r}_1| - l_0}{|\Delta \vec{r}_1|} \Delta \vec{r}_1, \quad \vec{r} = \vec{r}_0 + \Delta \vec{r}.$$

Пусть $\vec{n}, \vec{\tau}$ - единичные нормальный и тангенциальный векторы по длине разворачиваемого троса, локально в окрестностях БКА. Найдем скорость нового элемента.

Обозначим через α угол между векторами $\Delta \vec{r}_1$ и $\Delta \vec{V}_1$. Найдем составляющую относительной скорости новой точки, направленную вдоль троса. Разложим скорость $\Delta \vec{V}_1$ по тангенциальной и нормальной составляющим:

$$\Delta \vec{V}_1 = \Delta \vec{V}_{1\tau} + \Delta \vec{V}_{1n}.$$

Тангенциальную составляющую несложно подсчитать по формуле:

$$\Delta \vec{V}_{1\tau} = |\Delta \vec{V}_1| \frac{|\Delta \vec{r}_1|}{|\Delta \vec{r}_1|} \cos \alpha, \quad \cos \alpha = \frac{(\Delta \vec{r}_1, \Delta \vec{V}_1)}{|\Delta \vec{r}_1| \cdot |\Delta \vec{V}_1|}$$

Теперь можно вычислить $\Delta \vec{V}_{1n} = \Delta \vec{V}_1 - \Delta \vec{V}_{1\tau}$. Учитывая, что тангенциальная составляющая скорости в новой точке и крайней по тросу совпадают, можно определить вектор скорости в формируемой точке:

$$\vec{V}_n = \frac{|\Delta \vec{r}|}{|\Delta \vec{r}_1|} \Delta \vec{V}_{1\tau}, \quad \Delta \vec{v} = \vec{V}_n + \Delta \vec{V}_{1\tau}, \quad \vec{V} = \vec{V}_0 + \Delta \vec{V}$$

Полученный результат следует поправить с учетом того факта, что масса КА уменьшилась на величину массы выделенного участка троса m_i :

$$\vec{V} = \frac{m_{КА}\vec{V}_0 - m_i\vec{V}}{m_{КА} - m_i}$$

Естественно, что массу БКА следует уменьшить на величину m_i . Длину отделяемого участка можно посчитать по формуле:

$$l_1 = l \frac{|\Delta\vec{r}_1| - |\Delta\vec{r}|}{|\Delta\vec{r}_1|},$$

при этом на остаток троса, прилегающего к БКА, полагается:

$$l_{00} = l \frac{|\Delta\vec{r}|}{|\Delta\vec{r}_1|}$$

В качестве условия формирования новой точки можно рассмотреть выражение:

$$l > l_0 + 0.25 \cdot l_0$$

3. Модель параллельного алгоритма управляемого движения КТС

3.1. Описание схемы сегментации троса для случая управляемого движения КТС

Особенностью рассматриваемого случая управляемого движения КТС является неравномерность распределения вычислительной нагрузки по мере развертывания троса. В начальный момент расчетов имеется два объекта движения: БКА и МКА. Очевидно, что такая задача не требует значительных вычислительных затрат. Однако по мере развертывания троса, когда количество точек троса увеличивается до нескольких сотен или тысяч, сложность вычислительной задачи многократно возрастает.

В концепцию сегментации троса, принятую на первом этапе исследований и описанную в [10], были внесены некоторые изменения, связанные с необходимостью учета динамики роста сложности решаемой задачи. В предлагаемом варианте вершины нумеруются от МКА к базовому космическому аппарату. Дело в том, что в процессе развертывания троса через определенные промежутки времени добавляются новые точки. Чтобы не перезаписывать элементы матрицы координат объектов КТС (точек, МКА и БКА) при добавлении новой точки, её координаты записываются на место БКА, а вектор координат БКА записывается в следующую строку.

Процесс развертывания троса происходит в «нулевом» процессе, который кроме расчетов КТС занимается организацией параллельных вычислений, то есть выполняет функции «менеджера» вычислительного процесса. По мере заполнения матрицы фазовых координат КТС в процессе 0, он организует перезапись части точек в свободный процессор и запуск на нем нового процесса-вычислителя. При этом в процессе 0 остаётся одна точка и БКА.

На рисунке 3 представлена схема сегментации троса для случая управляемого движения КТС. Предлагаемый параллельный алгоритм управляемого движения троса начинает работать как последовательная программа в одном процессе (рисунок 3(а)). В момент заполнения матрицы фазовых координат в процессе 0 практически вся матрица, кроме двух последних точек, перезаписывается в процесс 1 (рисунок 3(б)). После этого алгоритм развертывания троса по-прежнему реализуется в процессе 0, а в процессе 1 производятся вычисления неуправляемого движения троса [10]. На последнем этапе развертывания троса «запускаются» все процессы (рисунок 3(в)), а по завершении развертывания троса в процессе 0 предложенная параллельная вычислительная схема моделирует неуправляемое движение КТС.

На рисунке 3 $D_0 - D_3$ - информационные матрицы, обеспечивающие каждый из сегментов троса, приписанный к соответствующему процессору, информацией о координатах точек соседних сегментов, смежных с крайними точками матриц фазовых координат каждого сегмента. На рисунке этот факт схематично показан пунктирными стрелками.

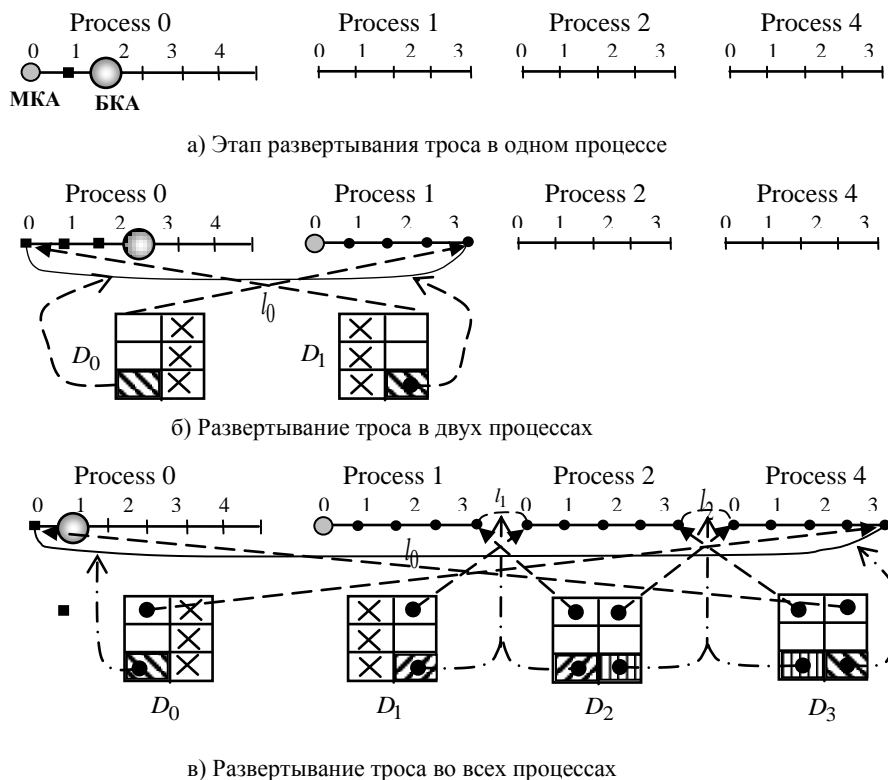


Рисунок 3. Схема сегментации КТС для случая управляемого движения

3.2. Графическая модель параллельного алгоритма

Изобразим параллельный алгоритм графически. Для этого воспользуемся нотацией технологии графосимволического программирования (ГСП) [11]. Технология ГСП позволяет описать алгоритм в виде набора диаграмм потоков управления, а затем автоматически скомпилировать и запустить программу, реализующую этот алгоритм. Диаграмма потоков управления представляет собой ориентированный граф, в котором вершины обозначают действия, выполняемые над данными некоторой предметной области, а дуги обозначают последовательность выполнения этих действий. Если из вершины исходят несколько дуг, то в зависимости от типа дуг может выполняться либо одна, либо одновременно несколько смежных вершин, в которые эти дуги ведут. Дуги различного типа обозначаются графически по-разному. Преимущество диаграмм потоков управления заключается в наглядном представлении последовательности развития вычислений в программе. Недостатком является отсутствие отображения зависимости по данным между вершинами. На рисунке 4 представлен фрагмент параллельного алгоритма, предназначенного для решения приведенной выше системы дифференциальных уравнений методом Рунге-Кутты четвертого порядка. Коэффициенты для метода Рунге-Кутты четвертого порядка вычисляются в четыре фазы (четыре стадии).

Дуги, начало которых помечено кружком, обозначают передачу управления в новый параллельный процесс. Дуги, начало которых помечено наклонной чертой, означают возврат управления из параллельного процесса. На каждой фазе вычислений используются четыре процесса для расчета различных сегментов троса.

Первый сегмент троса, прилегающий к КА, рассчитывается в модулях развёртывания (вершины 1.1, 2.1, 3.1, 4.1, 1.1.0, 2.1.0, 3.1.0, 4.1.0, 5.1.0, 6.1.0, на рисунке 4) с использованием математической модели расчета управляемого движения троса, описанной выше. Кроме того, указанные модули контролируют процесс запуска вычислений на других сегментах троса.

Процессы, описываемые вершинами 1.2-4.2, 1.3-4.3, 1.4-4.4, запускаются по мере накопления нужного количества точек на сегменте развертывания. При этом накопленный массив точек переносится на свободный процесс и изымается с сегмента развертывания. Модуль развертывания, представленный вершинами 1.1, 2.1, 3.1, 4.1, вычисляет коэффициенты Рунге-Кутты для соответствующей фазы алгоритма. В модуле 4.1, кроме уравнений движения КТС, учитывается работа программного аппарата развертывания тросовой системы и алгоритма добавления новых точек. Модуль развертывания также следит за распределением данных по процессам и отвечает за запуск остальных процессов. Фактически он исполняет роль менеджера, определяющего загрузку процессов параллельной программы. В конце расчетов на каждом временном шаге в модуле развертывания формируется сообщение для каждого из остальных процессов, которое сигнализирует о количестве сформированных сегментов троса. Это количество хранится в переменной F_{upr} . Сообщения отправляются из вершины 5.1.0 в вершины 5.X.0 ($X=2..4$). В зависимости от содержимого полученного сообщения каждый процесс либо продолжает ожидать своей очереди на запуск, либо запускается и получает от модуля развертывания свой сегмент троса, либо изменяет номера своих соседей по коммуникации в соответствии со схемой, представленной на рисунке 3. При достижении последнего временного шага моделирования модуль развертывания формирует сигнал о завершении вычислений. Остальные модули в процессах 2,3,4 работают в соответствии с моделями, описывающими свободное движение КТС для соответствующих сегментов.

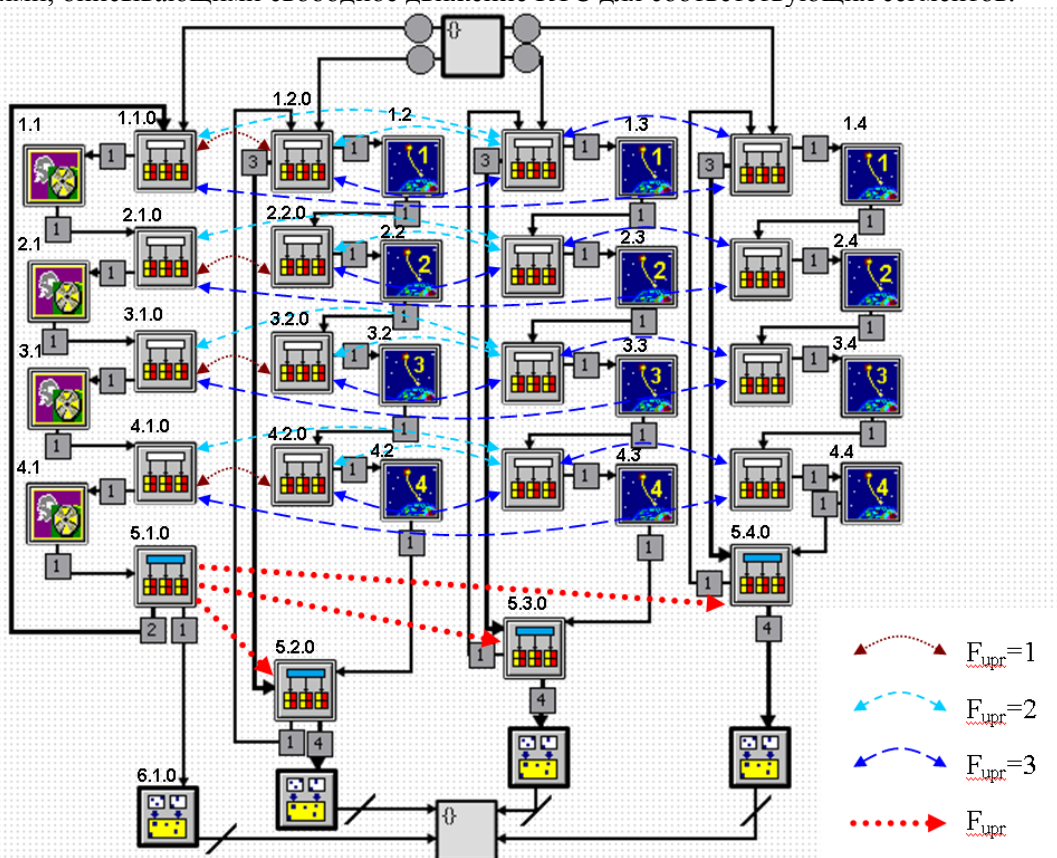


Рисунок 4. Параллельный алгоритм вычисления коэффициентов метода Рунге-Кутты.

На рисунке 4 некоторые вершины для наглядности помечены пиктограммами с упрощенным изображением космической тросовой системы и номером фазы вычислений. Эти вершины представляют собой последовательные подпрограммы вычисления коэффициентов формулы Рунге-Кутты для отдельных сегментов троса. Таким образом, граф алгоритма на рисунке 4 является иерархическим. Иерархическое построение графов алгоритмов в технологии ГСП

позволяет скрыть несущественные подробности реализации различных частей алгоритма, оставляя на графе только важные элементы с точки зрения понимания структуры алгоритма.

На рисунке 4 пунктирными дугами изображены сообщения между процессами. Сообщения пересылают данные и формируют механизм синхронизации между процессами. Операция приема сообщения является блокирующей, то есть принимающий процесс приостанавливается до получения сообщения, представленного дугой на графе программы. Модули $X.Y.0$ ($X=1..4$, $Y=1..4$) на каждой фазе вычислений осуществляют обмен граничными точками между процессами, выполняющими расчет соседних сегментов троса. Номера процессов-адресатов сообщений меняются в зависимости от количества сформированных сегментов троса. На рисунке 4 сообщения, соответствующие каждому количеству сегментов троса, обозначены дугами уникального вида.

3.3. Оценка ускорения параллельного алгоритма численного моделирования управляемого движения КТС

Оценим ускорение параллельной программы, реализующей предложенный алгоритм. Вершины одной фазы вычислений, выполняющиеся одновременно в различных параллельных процессах и обозначенные номерами $X.1$, $X.2$, $X.3$, $X.4$ на рисунке 4 (где $X=1..4$ — номер фазы), обрабатывают сегменты троса одинаковой длины. Продолжительность работы этих вершин приблизительно одинакова во всех процессах в пределах одной фазы. Вершины процесса 0, в которых моделируется работа системы регулирования и движение точек троса, непосредственно примыкающих к БКА, обрабатывают переменное количество точек (от 2 до K). На этапе развертывания троса среднее количество обрабатываемых точек в процессе 0 меньше, чем в остальных процессах, поэтому вершины процесса 0 работают быстрее вершин остальных процессов в каждой фазе. С приближением количества точек в разворачивающемся первом сегменте троса к числу точек в остальных сегментах длительность вычислений во всех процессах выравнивается. Перед отделением очередной секции троса процесс 0 работает дольше остальных, поскольку в нем моделируется процесс развертывания и движение большого количества точек, близкого к числу точек в остальных процессах. Тем не менее, трассировка программы показывает, что указанные флуктуации незначительно влияют на среднюю продолжительность выполнения вершин процесса 0. Обозначим эту величину через T_0 . Длительность выполнения вершин остальных процессов изменяется меньше и составляет $T_i \approx 1.132T_0$.

Моделирование управляемого движения КТС в рассматриваемой постановке задачи включает три этапа:

- 1) начальный этап развертывания, на котором выдвигается первый сегмент троса, и вычисления производятся последовательно в процессе 0;
- 2) этап развертывания, на котором продолжается моделирование системы управления и выполняется генерация новых точек троса с переносом сегментов троса на отдельные процессы;
- 3) этап свободного движения, на котором развертывание прекращается и моделируется свободное движение полной длины троса параллельно во всех процессах.

В каждой фазе между процессами передаются значения фазовых координат для граничных точек сегментов троса, обрабатываемых в соответствующих процессах. Длина этих сообщений фиксирована, поэтому длительность обмена сообщениями между двумя процессами на каждой фазе не зависит от номера фазы и длины сегментов троса в процессах. Если считать, что сообщения между процессами, образующими различные пары, могут передаваться одновременно, то суммарная длительность обмена сообщениями на каждой фазе не будет зависеть от количества процессов. Трассировка программы подтверждает это предположение.

Длительность работы последовательной программы можно оценить следующим выражением:

$$T_s = 4T_0N_0^d + (4T_0 + 4T_i)N_1^d + (4T_0 + 2(4T_i))N_2^d + \dots + (4T_0 + (p-1)4T_i)N_{p-1}^d + 4T_i p N^f, \quad (8)$$

где N_0^d — количество шагов по времени, в течение которых происходит развертывание первого сегмента троса, N_i^d — количество шагов по времени, в течение которых происходит

развертывание i -ого сегмента троса, N^f – количество шагов по времени, в течение которых моделируется свободное движение полностью развернутого троса, p – количество сегментов троса, равное количеству процессов в параллельной программе. При условии равномерного развертывания троса, используя формулу арифметической прогрессии, выражение (8) можно привести к следующему виду:

$$T_s = 4T_0N_0^d + 4T_0 \sum_{i=1}^{p-1} N_i^d + 2p(p-1)T_iN_i^d + 4T_i pN^f = 4T_0N^d + 2pT_i(N^d - N_0^d) + 4T_i pN^f, \quad (9)$$

где N^d – суммарное количество шагов по времени, в течение которых происходит развертывание троса.

Оценку длительности работы параллельной программы можно представить следующим выражением:

$$T_{||} = (4T_0 + \beta)N_0^d + (p-1) \frac{\beta K}{2} + (4T_i + 5\beta)(N^T - N_0^d) \quad (10)$$

где β – длительность обмена сообщениями на каждой фазе, K – количество точек в каждом сегменте, N^T – суммарное количество шагов по времени, в течение которого моделируется движение КТС. Второе слагаемое в выражении (10) оценивает длительность передачи сегмента троса процессу, отвечающему за его моделирование. Всего в программе $(p-1)$ таких передач. Длительность каждой передачи не учитывает длину пакетов коммуникационной среды, а оценивается в предельном случае как сумма длительностей передачи каждой точки. При этом значение β принимается за длительность передачи двух сообщений, содержащих по одной точке, так как процессы на каждой фазе попарно обмениваются своими граничными точками. Ускорение и эффективность параллельной программы оцениваются следующими выражениями:

$$S = \frac{T_s}{T_{||}} = \frac{4T_0N^d + 2pT_i(N^d - N_0^d) + 4T_i pN^f}{(4T_0 + \beta)N_0^d + (p-1) \frac{\beta K}{2} + (4T_i + 5\beta)(N^T - N_0^d)} \quad (11)$$

$$E = \frac{S}{p} = \frac{4T_0N^d p^{-1} + 2T_i(N^d - N_0^d) + 4T_i N^f}{(4T_0 + \beta)N_0^d + (p-1) \frac{\beta K}{2} + (4T_i + 5\beta)(N^T - N_0^d)} \quad (12)$$

С увеличением количества точек в каждом сегменте троса ускорение будет определяться отношением времени, в течение которого выполняется развертывание троса, к общему интервалу моделирования движения КТС:

$$S \xrightarrow{T_0 \rightarrow T_i \rightarrow \infty} \frac{4N^d + 2p(N^d - N_0^d) + 4pN^f}{4N_0^d + 4(N^T - N_0^d)} \quad (13)$$

Чем большую часть интервала моделирования занимает свободное движение троса, тем ближе ускорение к количеству процессов параллельной программы.

4. Вычислительные эксперименты

С использованием технологии графосимволического программирования предложенный параллельный алгоритм был реализован в виде программы для вычислительного кластера. В качестве средства передачи сообщений использовалась технология MPI. Программа была скомпилирована и запущена на высокопроизводительном вычислительном кластере «Сергей Королев» Самарского университета. Моделирование управляемого движения КТС проводилось для случая с отношением времени развертывания к общему интервалу моделирования, равным 0.425. На рисунке 5 приведены зависимости ускорения и эффективности параллельной программы, состоящей из четырех процессов, от количества точек в сегменте троса, моделируемом на каждом процессе.

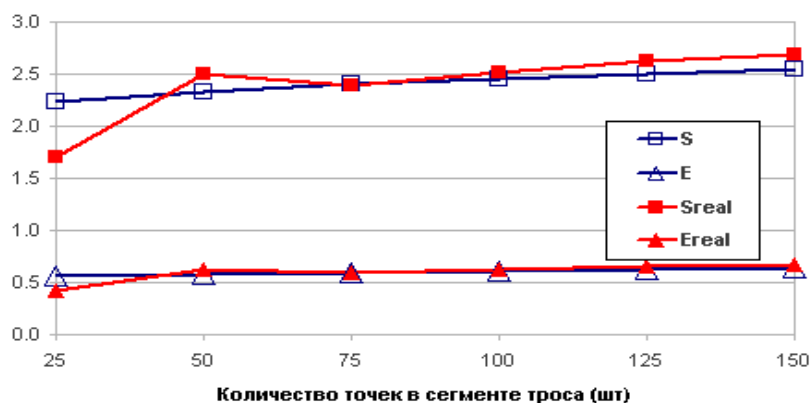


Рисунок 5. Зависимость ускорения (S) и эффективности (E) параллельной программы от количества точек в сегменте троса.

Линии S и E соответствуют оценкам, вычисленным по формулам (11) и (12), линии Sreal и Ereal соответствуют экспериментальным данным. По причине завышения оценки длительности передачи сообщений, указанного выше, значение теоретической оценки ускорения оказалось ниже реального в некоторых точках. В целом вычислительный эксперимент показал соответствие предложенных оценок ускорения и эффективности реальным измерениям.

5. Заключение

В работе представлен параллельный алгоритм моделирования управляемого движения КТС, позволяющий использовать произвольно большое количество точек в модели бисера. Увеличение количества точек позволяет приблизить рассматриваемую модель к модели системы с распределенными параметрами и выполнять моделирование различных процессов, возникающих в КТС, с недостижимой для других методов точностью. Приведены оценки ускорения и эффективности параллельных программ, реализующих описанный алгоритм. Близость теоретических оценок реально наблюдаемым зависимостям проверена экспериментально. Принцип организации параллельных вычислений при решении системы с переменным числом дифференциальных уравнений, предложенный в работе, может быть применен не только в модели бисера, но и в других моделях, что позволяет использовать преимущества современных вычислительных систем с параллельной архитектурой для повышения точности моделирования управляемого движения КТС.

6. Литература

- [1] Белецкий, В.В. Динамика космических тросовых систем / В.В. Белецкий, Е.М. Левин. – М.: Наука, 1990. – 336 с.
- [2] Cosmo, M.L. Tethers in Space Handbook. Third edition / M.L. Cosmo, E.C. Lorenzini. – Smithsonian Astrophysical Observatory, 1997. – 234 p.
- [3] Johnson, L. Overview of future NASA tether applications / L. Johnson, B. Gilchrist, R.D. Estes, E.C. Lorenzini // *Advances in Space Research*. – 1999. – Vol. 24(8). – P. 1055-1063.
- [4] Williams, P. In-plane payload capture with an elastic tether // *J. of Guidance, Control, and Dynamics*. – 2006. – Vol. 29(4). – P. 810-821.
- [5] Williams, P. Deployment/retrieval optimization for flexible tethered satellite systems // *Nonlinear Dynamics*. – 2008. – Vol. 52(1-2). – P. 159-179.
- [6] Zhong, R. Dynamic analysis of deployment and retrieval of tethered satellites using a hybrid hinged-rod tether model / R. Zhong, Z.H. Zhu // *Int. J. of Aerospace and Lightweight Structures*. – 2011. – Vol. 1(2). – P. 239-259.
- [7] Zabolotnov, Y.M. Control of the deployment of a tethered orbital system with a small load into a vertical position // *J. of Applied Mathematics and Mechanics*. – 2015. – Vol. 79(1). – P. 28-34.

- [8] Ван, Ч. Управление при разворачивании тросовой системы на эллиптической орбите / Ч. Ван, Ю.М. Заболотнов // Известия Самарского научного центра Российской академии наук. Авиационная и ракетно-космическая техника. – 2017. – Т. 19, № 1. – С. 91-97.
- [9] Dong, Z. Modeling and analysis of deployment dynamics of a distributed orbital tether system with an atmospheric sounder / Z. Dong, Y.M. Zabolotnov, C. Wang // Advances in the Astronautical Sciences. – 2017. – Vol. 161. – P. 1053-1063.
- [10] Kovartsev, A.N. Application of Graph Models to the Parallel Algorithms Design for the Motion Simulation of Tethered Satellite Systems / A.N. Kovartsev, V.V. Zhidchenko // PaCT, LNCS. – 2017. – Vol. 10421. – P. 114-123. DOI: 10.1007/978-3-319-62932-2_10.
- [11] Egorova, D. Visual Parallel Programming as PaaS cloud service with Graph-Symbolic Programming Technology / D. Egorova, V. Zhidchenko // Proc. Inst. Syst. Program. – 2015. – Vol. 27(3). – P. 47-56.

Parallel algorithm for controlled motion simulation of space tether systems

V. Zhidchenko¹, A. Kovartsev

¹Samara National Research University, Moskovskoe Shosse 34A, Samara, Russia, 443086

Abstract. Controlled motion simulation of space tether systems differs significantly from the free motion case. Deployment of tether increases its length and adds new points into discrete model of tether. A set of dynamic equations is expanded with equations that describe a motion of new points. The necessary condition at each step is consistency of the system. Tether deployment is supervised by control system which limits the tether length. The problem arises of solving a system of differential equations with variable and increasing number of equations. In this paper a parallel algorithm for controlled motion simulation of space tether systems is designed. The parallel algorithm provides an ability to split the tether to such number of points that the model becomes similar to the model with distributed parameters. As a result the simulation of several modes of tether motion, unavailable for other methods, becomes possible. The estimation of speedup provided by the algorithm is implemented. It shows that the speedup is comparable with the more simple case of free motion. Computational experiments have been carried out for algorithm performance evaluation.