Полет на Луну: SMath vs SimInTech

Валерий Очков, Дарья Абрамова, Александр Бобин
НИУ «МЭИ»
Вячеслав Петухов
ООО «3В Сервис»

Аннотация. В статье рассказано, как с помощью двух отечественных компьютерных программ SMath Studio и SimInTech можно решать интересные задачи, возвращаясь при этом к истокам численных методов.

«Мир невозможен без времени,
но и время невозможно без мира»
Артур Шопенгауэр

«Полет на Луну» – был такой советский мультфильм 1953 года, два кадра которого показаны на рис. 1. Его можно посмотреть в интернете – https://yandex.ru/video/preview/10739278098391238699. Фильм наивный, но довольно забавный, если принять во внимание, что через четыре года был запущен первый искусственный спутник Земли, в 1961 году человек полетел в космос, а в 1969 году люди высадились на Луне.

В те времена почти все мальчишки, включая и одного из авторов этой статьи, мечтали стать космонавтами и так рисовали старт космических аппаратов: ракета разгоняется по длинной параболической эстакаде и по диагонали взмывает ввысь «навстречу звездам».

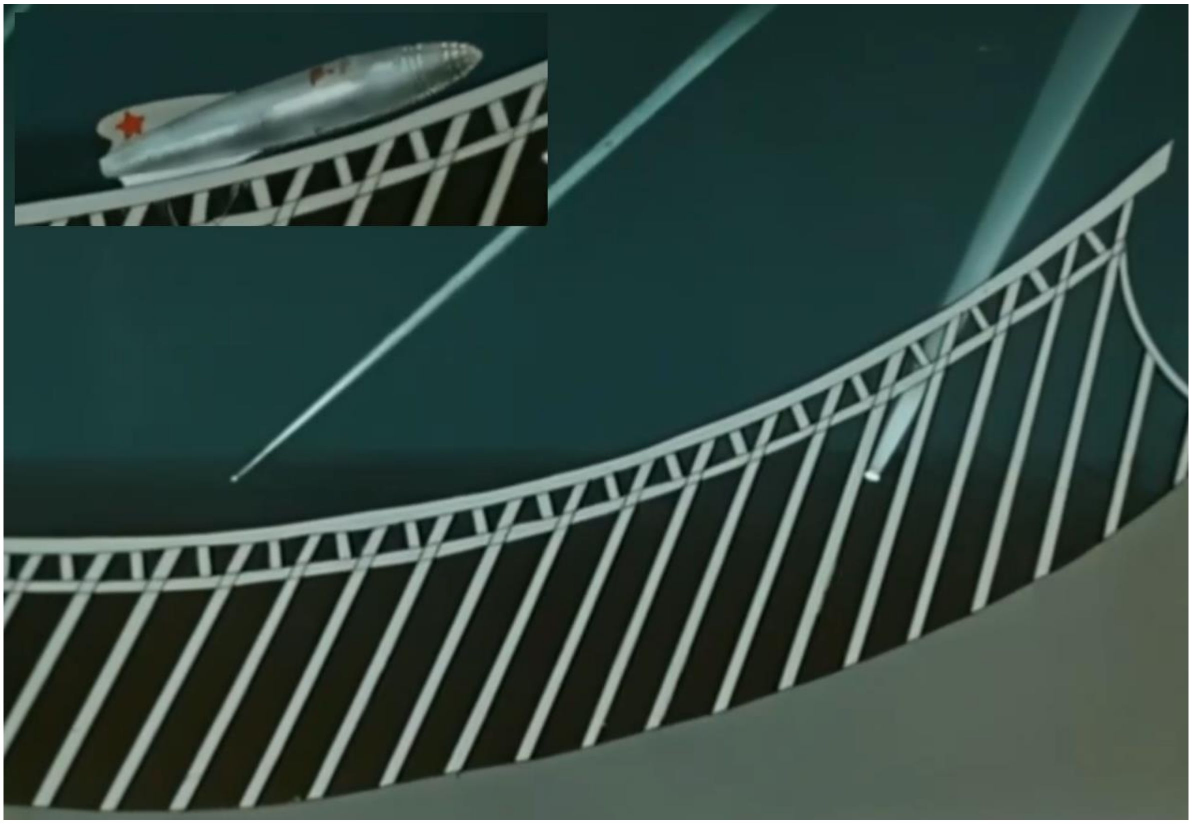


Рис. 1. Кадры из мультфильма «Полет на луну»

Но когда этот автор, будучи уже не ребенком, а юношей впервые увидел кадры кинохроники пуска реальной ракеты, то он очень удивился и разочаровался. Никакой романтики! Ракета, привязанная к какому-то столбу, отвязывается от него и как бы нехотя отрывается от поверхности Земли и медленно в клубах дыма вертикально поднимается вверх. Дело в том, что в СССР до конца 60-х годов прошлого века ни в кинотеатрах, ни по телевизору пуск советских ракет не показывали по соображениям секретности, а американских по идеологическим причинам. Поэтому-то ребята и рисовали фантастические огромные эстакады из знаменитого мультфильма.

А давайте рассчитаем пуск ракеты – изменение во времени её высоты и скорости и, главное, покажем, как это делается с помощью современных отечественных программных средств SMath и SimInTech, довольно успешно «импортнозаменивших» такие программы, как Mathcad и Simulink. Правильнее сказать так – программы SMath и SimInTech довольно успешно применялись у нас уже давно. Санкции дали новый импульс их развитию.

Ещё немного про кино времен детства и юношества одного из авторов. Есть такой советский культовый фильм «Девять дней одного года» (1961 г.[1]). В нем нет погонь и драк, на которые «подсели» современные юные и уже не совсем юные зрители, а есть прекрасные диалоги в исполнении прекрасных актеров: Татьяны Лавровой, Иннокентия Смоктуновского, Алексея Баталова и др.

Рис. 2. Кадр из фильма «Девять дней одного года»

Вот один диалог из фильма (рис. 2): актеры Евгений Евстигнеев (физик-теоретик Николай Иванович — слева) и Михаил Казаков (физик-романтик Валерий Иванович — справа):

— Скажите, Валерий Иванович, как глубоко вы собираетесь забраться в глубины нашей галактики?

— На 500 световых лет.

— С какой скоростью?

— Близкой к скорости света.

— Вес корабля?

— Сто тысяч тонн.

— Горючее?

— Самое современное.

— Сейчас мы подсчитаем, сколько вам понадобится горючего. Будьте добры, салфеточку.

И далее:

— Николай Иванович, вы закончили свои подсчеты?

— Да, пожалуйста. При весе космического корабля сто тысяч тонн при скорости, близкой к скорости света, для облета части галактики в разумный для человеческой жизни срок вам потребуется десять в двадцать второй степени тонн самого современного экстра-горючего. Справка.

---

[1] Его можно посмотреть на сайте: http://kino-ussr.ru/main/217-devyat-dney-odnogo-goda-1961.html.

Наша планета весит несколько меньше, счастливого пути!

Диалог, конечно, наивный, но представим себе, что у Николая Ивановича в руках не салфетка, а… «таблетка» – планшетный компьютер с программой SMath, и он на ней делает расчёты (рис. 3).

$$\mathrm{maple}\left(\mathrm{solve}\left(v_{end} = u \cdot \ln\left(\frac{M_1}{M_2}\right), M_1\right)\right) = \exp\left(\frac{v_{end}}{u}\right) \cdot M_2$$

$$M_2 := 100000\ \text{т} \qquad v_{end} := 0.995\ \text{c} = 298293\ \frac{\text{км}}{\text{с}}$$

$$u := 777 \cdot 1000\ \frac{\text{тс}}{\frac{\text{т}}{\text{с}}} = 7620\ \frac{\text{км}}{\text{с}}$$

$$M_1 := M_2 \cdot \mathbf{e}^{\frac{v_{end}}{u}} = 1.0034 \cdot 10^{22}\ \text{т}$$

Рис. 3. Реконструкция расчёта на салфетке

На рисунке 3 показана *формула Циолковского* (кстати, сам Циолковский упоминается в фильме[2]): конечная скорость ракеты v_{end} пропорциональна натуральному логарифму отношения стартовой массы ракеты M_1 к её конечной массе M_2. Разность же $M_1 - M_2$ — это масса топлива и окислителя (далее — просто топлива, как в фильме). Коэффициент пропорциональности в этом уравнении (величина u) — это отношение тяги двигателя f к массовому расходу топлива. Величины f и μ мы будем использовать в дальнейших расчетах.

Мы приняли значение переменной u равным 777 тысяч тонн силы (тс), деленных на тонны в секунду (т/с). Но в самом фильме эта цифра не озвучена, а 777 тысяч — это наше предположение (три семерки[3] на счастье!). Величина u — это скорость истечения газов из сопла ракеты (км/с). Теоретическое значение этой скорости для ядерных ракетных двигателей может превышать 70 км/с. Скорость истечения для электрического двигателя может достигать 140 км/с. Поэтому вполне можно помечтать и о «трех семерках».

Уравнение Циолковского на рисунке 2 решается относительно переменной M_1. В полученную

[2] Персонаж Казакова (Валерий Иванович) в конце диалога бросает своему оппоненту (персонажу Евгения Евстигнеева) такую фразу: «Когда Циолковский проектировал свою ракету, то в ресторане Яр сидели ученые-скептики вроде вас и на салфеточках доказывали, что он сумасшедший».
[3] На столе, показанном на рис. 1, мог стоять популярный в те времена портвейн «Три семерки», подсказавший физику-теоретику эту цифру.

формулу с экспонентой подставляются числовые значения и рассчитывается ответ. Если конечная скорость ракеты будет близка к скорости света (мы приняли, что эта скорость равна 0.995 скорости света c), то масса топлива (примерно 10^{22} тонн) и вправду превысит массу Земли – примерно $5.8 \cdot 10^{21}$ тонн. Но расчет наш, конечно, довольно грубый. Кроме того, он не учитывает то важное обстоятельство, что при скоростях, близких к скорости света, формула Циолковского не работает. Тут нужно будет отходить от законов классической механики и прибегать к теории относительности Эйнштейна.

Но давайте спустимся с галактических высот к околоземному пространству и подсчитаем, какие параметры могут быть у реальных, а не у полуфантастических космических аппаратов.

На рисунке 4 показан вывод формулы Циолковского через аналитическое решение дифференциального уравнения движения материальной точки с уменьшающейся массой. На ракету действуют две силы – сила тяги f и сила притяжения Земли, которая также уменьшается со временем. Сначала допускается, что значение ускорения свободного падения g – это постоянная величина, но затем это допущение будет снято – см. рис. 9.

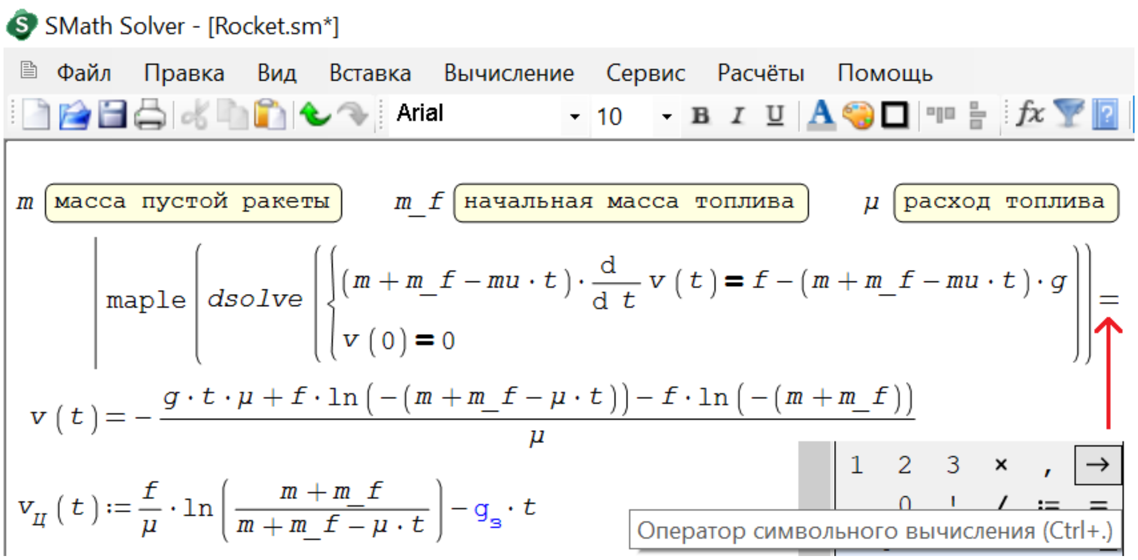


Рис. 4. Вывод формулы Циолковского

На рисунке 6 сделана попытка расчёта скорости ракеты, имеющей стартовую массу 250 тонн, из которых 50 тонн — это масса пустой ракеты (m), а 200 тонн — масса топлива (m_f). Тяга двигателя ракеты (f) составляет 2549 тонн силы (тс) при расходе топлива 5 тонн в секунду (т/с — μ). При таких исходных данных двигатель будет работать 40 секунд (t_c). Учитывается сопротивление воздуха, сила которого пропорциональна (коэффициент c) плотности воздуха, зависящей от высоты (рис. 5), площади лобового сечения s и квадрату скорости v.

$$\rho(h) := \left| \begin{array}{l} h := \min([h\ \ 40\ \text{км}]) \\ k := 0.0065\ \dfrac{\text{К}}{\text{м}} \\ T_0 := 15\ {}^\circ\text{C} \\ T := T_0 - k \cdot h \\ M_{air} := 28.96442\ \dfrac{\text{г}}{\text{моль}} \\ p := 1\ \text{атм} \cdot \left(1 - \dfrac{k \cdot h}{T_0}\right)^{\dfrac{g_з \cdot M_{air}}{R_m \cdot k}} \\ \dfrac{p \cdot M_{air}}{R_m \cdot T} \end{array} \right.$$

Рис. 5

$m := 50\ \text{т}$ масса пустой ракеты $\quad m\_f := 200\ \text{т}$ начальная масса топлива

$\mu := 5\ \dfrac{\text{т}}{\text{с}}$ расход топлива $\quad u := 5000\ \dfrac{\text{м}}{\text{с}}$ скорость уходящих газов

$f := \mu \cdot u = 2549\ \text{тс}$ тяга двигателя ракеты

$t_e := \dfrac{m\_f}{\mu} = 40\ \text{с}$ время работы двигателя

$c := 0.41$ коэффициент сопротивления $\quad d := 4\ \text{м}$ диаметр ракеты

$s := \pi \cdot \left(\dfrac{d}{2}\right)^2 = 12.57\ \text{м}^2$ площадь лобового сопротивления

$n := 1000 \qquad t_{end} := 40\ \text{с} \qquad v_{Ц}(t_e) = 7.6549\ \dfrac{\text{км}}{\text{с}}$

$$\begin{cases} h(0\ \text{с}) = 0\ \text{м} \qquad h'(0\ \text{с}) = 0\ \dfrac{\text{м}}{\text{с}} \\ (m + m\_f - \mu \cdot t) \cdot h''(t) = f - (m + m\_f - \mu \cdot t)\, g_з - c \cdot \rho(h(t)) \cdot s \cdot \dfrac{h'(t)^2}{2} \end{cases}$$

Вставка

Mathcad Блок

$$M := \text{rkfixed}\left(h(t),\ t_{end},\ \left| \begin{array}{l} [\ \text{кг} := 1\ \ \text{м} := 1\ \ \text{с} := 1\ ] \\ n \end{array} \right. \right)$$

$t := \text{col}(M, 1)\ \text{с} \qquad h := \text{col}(M, 2)\ \text{м} \qquad v := \text{col}(M, 3)\ \dfrac{\text{м}}{\text{с}}$

Рис. 6. Численное решение дифференциального уравнения полёта ракеты

Решение, полученное операторами рис. 6 (три вектора с именами t, h и v), отображается графически – см. рис. 7.

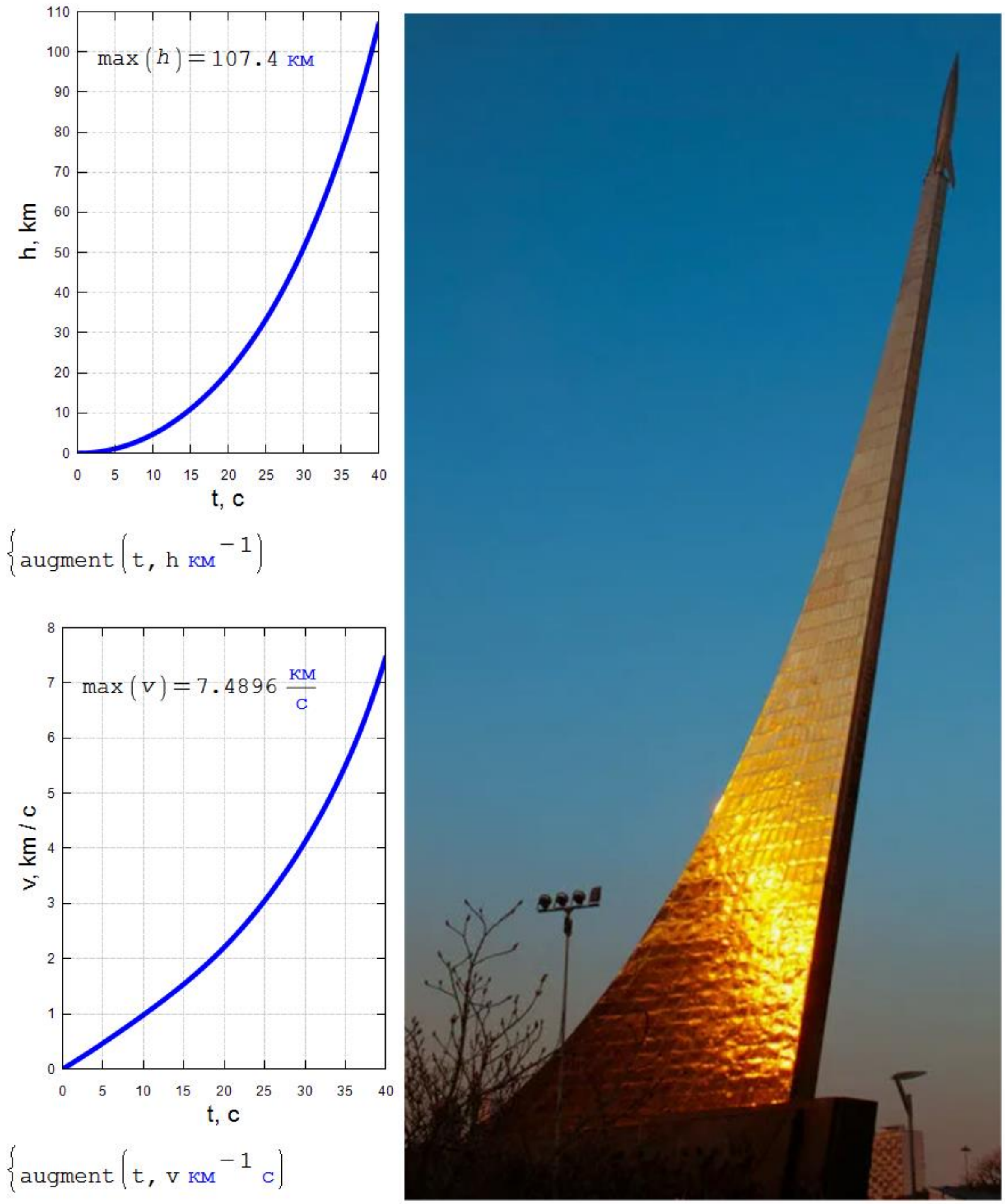


Рис. 7. Графики изменения высоты полета и скорости ракеты в зависимости от времени

Даже беглый взгляд на графики рисунка 7 вызывает в памяти разгонную ракетную эстакаду из мультфильма «Полёт на луну» (рис. 1) и… монумент покорителям космоса, установленный в Москве у ВДНХ. Можно предположить, что создатели этого монумента бесспорно были под впечатлением от этого мультфильма.

Но, откинем лирику в сторону и вернемся к физике – решим задачу о старте ракеты в среде программы SMath!

Сразу скажем, что это сделать не так-то просто. Тут будет иметь место некий психологический барьер. Опять даст себя знать некий шок, подобный тому, какой случился с одним из авторов, когда он увидел кинохронику старта реальной ракеты без разгонной эстакады. Или другое потрясение, какое пережил этот же автор, выросший на соцреализме в изобразительном искусстве, впервые увидев картину Сальвадора Дали, показанную на рис. 8.

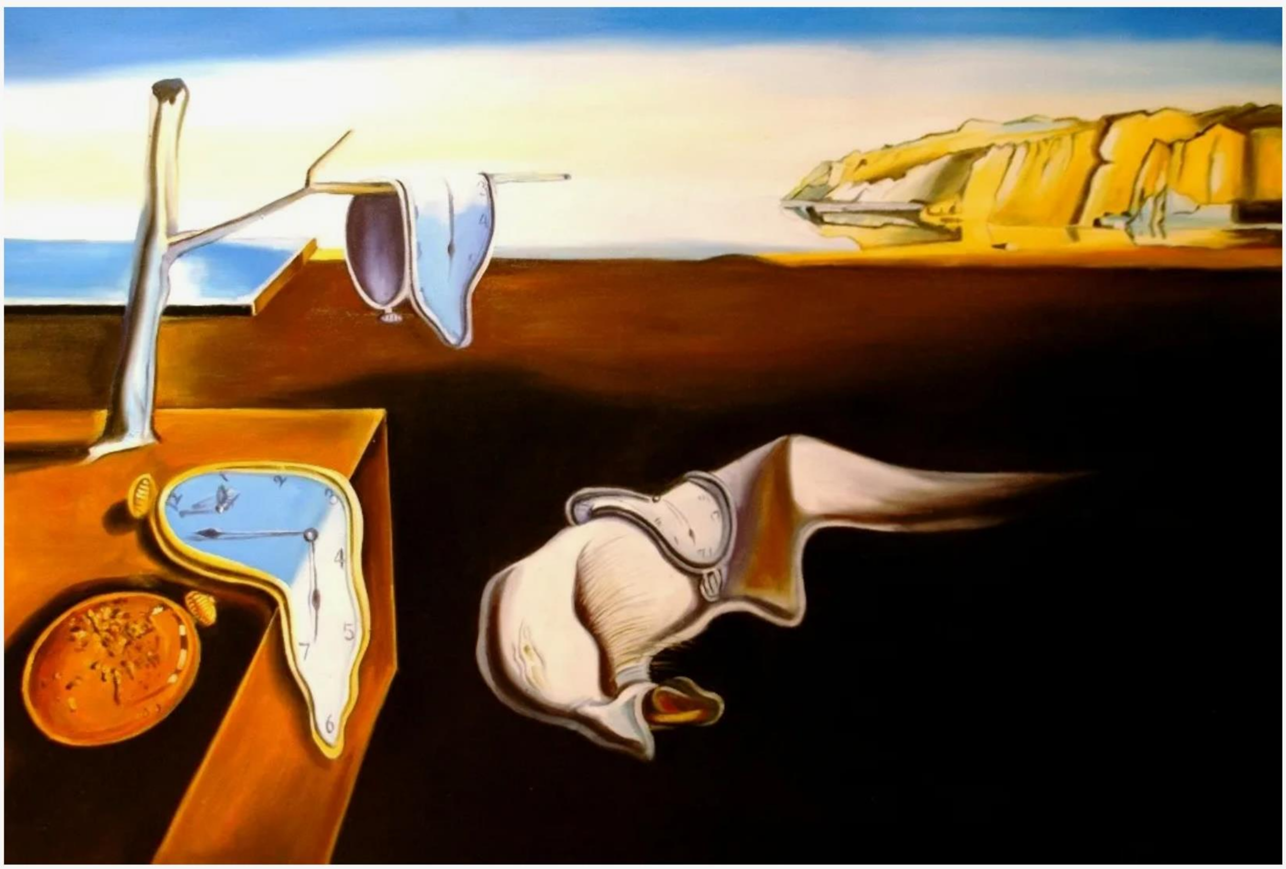


Рис. 8. Картина «Постоянство памяти» (фр. La persistance de la mémoire, 1931) художника Сальвадора Дали. Она широко известна и часто упоминается в популярной культуре. Чаще её называют более описательными названиями, такими как «Мягкие часы», «Тающие часы» или просто «Время».

Да, физическая величина время – это некая базовая субстанция программы SimInTech. В программе SMath мы пространство и скорость помещали в векторы h и v, элементы которых фиксировали дискретные значения в зависимости даже не самого времени, а некой цепочки дискретных значений времени. В среде SimInTech дискретность времени как бы пропадает, оставляя у человека полное ощущение непрерывности (текучести – см. рис. 8) единого времени, движущегося по связям между цифровыми объектами. Пространство, масса, скорость и другие физические величины становятся некими сопутствующими время субстанциями – см. эпиграф статьи.

На рисунке 9 показана блок-схема расчёта старта ракеты. В школе на уроках информатики детей учат рисовать схемы, отображающие в алгоритмах условные операторы и циклы. В программе SimInTech блок-схема наполнена другим содержанием – и техническим и сущностным – экзистенциональным, если так можно выразится.

После запуска программы SimInTech появляется лента кнопок, показанная в верхней части рис. 9. Далее необходимо нажать на крайнюю слева кнопку с изображением пустого листа с загнутым верхним уголком и выбрать из появившегося списка типовых проектов схему модели общего вида. После этого появится лист, разлинованный в клеточку. Клеточки мы убрали, воспользовавшись инструментами, расположенными в конце листа (сетка вкл./сетка откл.) и разместили на листе блок «Часы», ткнув курсором мыши сначала на кнопку с буквой t в ленте Источники, а затем в лист нашего проекта. После этого появится изображение часов с одним выходным портом, из которого будет вытекать время – Хронос. Вот, что можно прочесть в интернете, если запустить поиск по этому слову.

Хаос породил самое древнее, что было в нашей начинающейся Вселенной — Время. Эллины звали его Хронос. И теперь уже всё происходило во времени, так как пространство ещё только зарождалось.

В нашем Хаосе (пустой лист проекта) появилось Время – блок с изображением часов…

Затем в проект нужно будет вставить другие блоки, которые «породят пространство» – цифрового двойника полёта нашей ракеты: блок программирования (PL – рис. 10, где рассчитывается скорость ракеты по формуле Циолковского), блок интегрирования, который по скорости определяет пройденный путь. Скорость и время через блок Временной график отображаются на графике – см. рис. 11. Так как ускорение свободного падения зависит от высоты, а в арсенале пакета SimInTech есть соответствующий инструмент такого определения, то в расчёт вставлен ещё один блок с изображением Земного шара, возвращающий значение ускорение свободного падения по высоте. В принципе этой зависимостью можно пренебречь, так как на рассматриваемых высотах эта величина меняется незначительно: от 9.8 м/с^2 на уровне моря до 9.47 м/с^2 на высоте примерно 110 км. Этот блок вставлен в расчёт скорее для, чтобы показать, что такое *обратная связь* (важный элемент теории управления): выходной параметр влияет на входной. А именно для этой научной дисциплины пакет SimInTech в первую очередь и предназначен. Управление – это навигация по океану Времени.

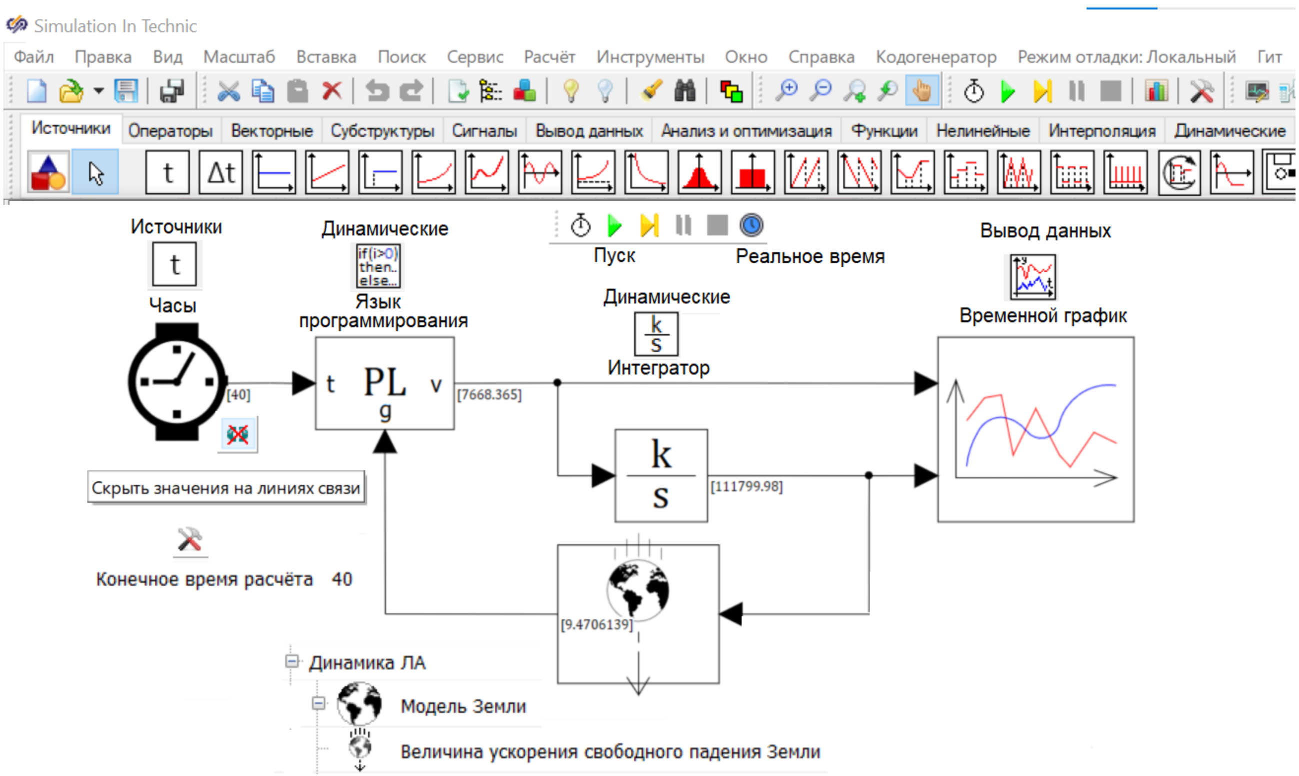


Рис. 9. Блок-схема расчета старта ракеты в среде SimInTech

Вставленные в расчет блоки необходимо протяжкой мыши соединить линиями, связывающими выходные порталы одних блоков с входными порталами других блоков. Далее следует сделать следующее:

1. В блок программирования PL через клавиатур ввести программу, показанную на рис. 10. Блок для такой записи раскрывается после двойного щелчка мышкой по его изображению. В программе четыре ключевых слова: input, const, end и output, смысл которых вполне понятен. Значение скорости полета ракеты ведётся по формуле, которую мы вывели ранее – см. рис. 4. Следует отметить, что язык программирования пакета SimInTech качественно отличается от языка программирования пакета SMath такими моментами. Во-первых, нет естественной записи формул с разводом по горизонтали числителя и знаменателя и, во-вторых, но не в последних, отсутствием механизма единиц измерения. В среде SimInTech все переменные безразмерны и имеют значения в базовых единицах СИ, которые отмечены в комментариях зелёного цвета. Вот бы внедрить язык SMath в среду SimInTech! Но это не так просто сделать.
2. Заменить конечное время расчёта с 10 (умолчание) до 40 секунд. Это делается через диалоговое окно, вызываемое нажатием кнопки Параметры расчета… – кнопки с изображением молотка и отвертки в правом верхнем углу рис. 9.
3. Отформатировать должным образом блоки, показанные на рис. 10. В частности, переместить порт ввода значения ускорения свободного падения g с левой части блока PL в его нижнюю часть, переставить местами входной и выходной порталы в блоке расчёта значения ускорения свободного падения так, чтобы вход был справа, а выход слева. У нас время и связанные со временем параметры текут слева направо, а тут мы как бы время поворачиваем вспять, реализуя обратную связь. Все эти действия

делаются через вызов соответствующих диалоговых окон, связанных с правой кнопкой мышки.

4. Открыть показ в квадратных скобках значений изменяющихся во времени параметров ракеты, нажав на кнопку с изображением скобок. Эта кнопка продублирована около часов на рис. 10.

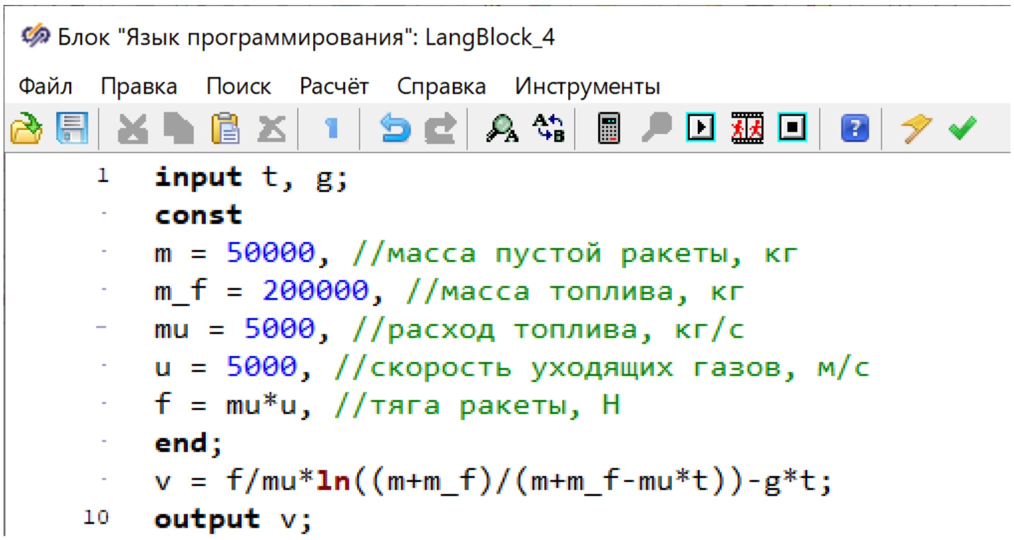


Рис. 10. Программа расчёта скорости ракеты

Теперь, если нажать кнопку Пуск (вспомним знаменитое гагаринское «Ну, поехали!»), показанную зелёным треугольником на рис. 9, то цифры в квадратных скобках моментально промелькнут с нулевых значений до значений, показанных под линиями связи на рис. 9 (значение g «промелькнет» в диапазоне с 9.80665 до 9.4706139). Двойной щелчок по блоку Временной график выведет на экран дисплея графики, показанные на рис. 11. Будут показаны два графика на одном, которые можно и нужно разделить на два отдельных через вызов диалогового окна Свойства нажатием правой кнопки мыши. Через это окно меняются и другие свойства графиков, пара которых отображена на рис. 11.

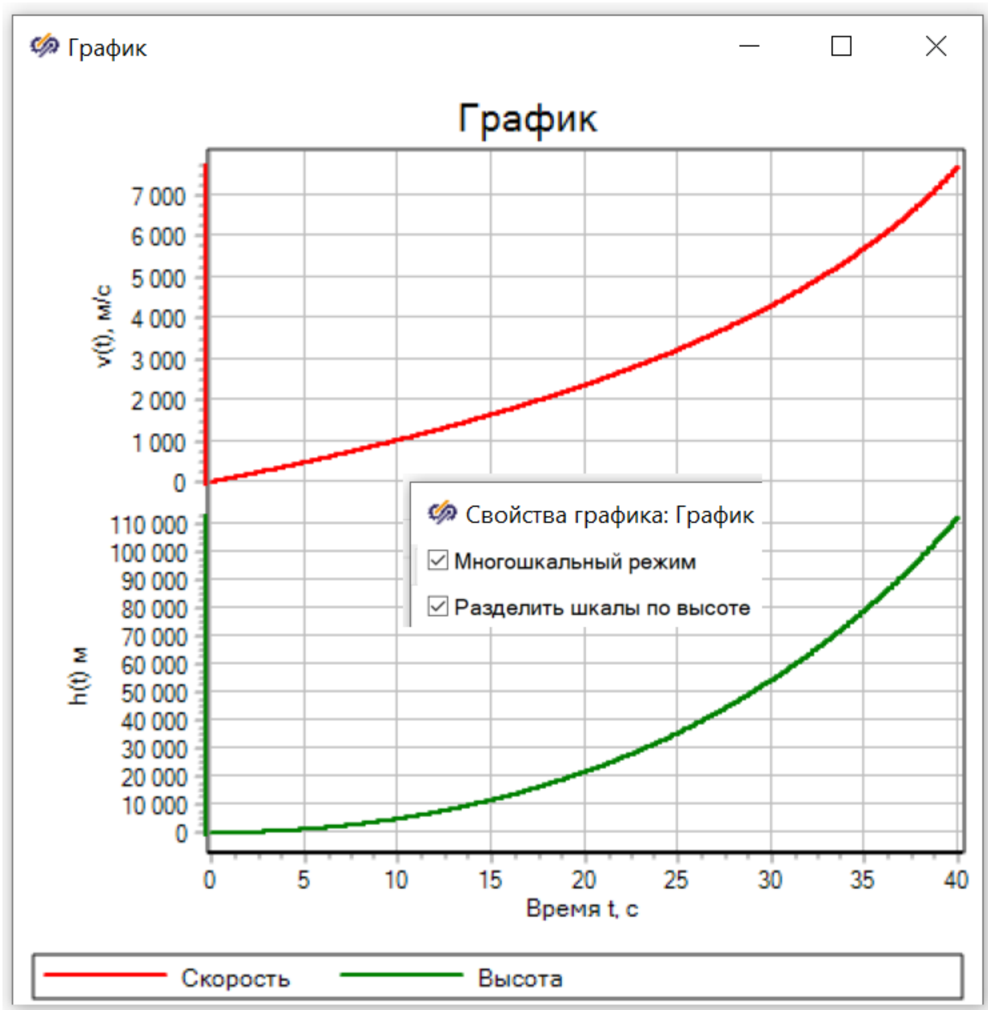


Рис. 11. График изменения скорости и высоты полёта ракеты

Мы не зря начали эту статью с упоминанием мультфильма «Полёт на Луну». Нам важно не только его содержание – старт ракеты, но и форма этого вида киноискусства – анимация. Если притормозить бег времени, нажав на кнопку Реальное время, расположенную правее кнопки Пуск то при отказе от автомасштабирования осей, можно получить эту самую анимацию – зримое изменение во времени графиков полёта ракеты (рис. 12).

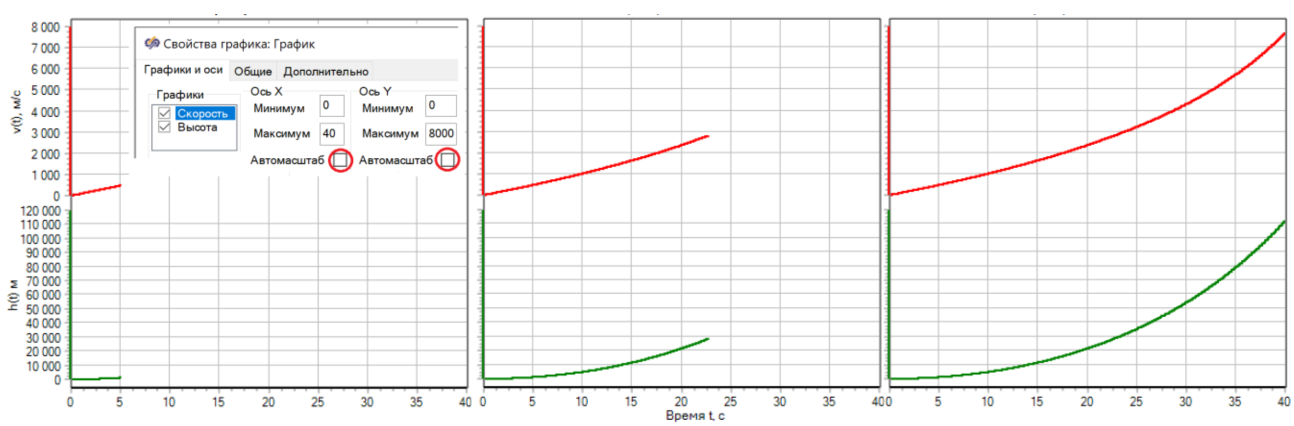


Рис. 12. Изменение во времени графиков полёта ракеты

Можно представить себе, как инженеры и учёные в центре управления полетами с замиранием сердца смотрели на эту «анимацию», вскакивали, кричали и обнимались, поздравляя друг друга с очередным успешным запуском ракеты-носителя. Но наша ракета, увы, не достигла первой космической скорости. Достигнет её она в своем двухступенчатом варианте – см. ниже. Помешало этому и сопротивление воздуха, которое нужно учесть – см. рис. 13.

Но перед этим давайте немного поговорим об анимации, об её кадрах. Современные молодые люди выросли на компьютерной анимации. «Полёт на Луну» – это рисованная анимация, требующая от её создателей – художников-мультипликаторов помимо таланта еще и огромного труда, связанного с рисованием отдельных кадров. Их количество, делённое на время показа, зависело от динамики отображаемого сюжета – персонаж мультфильма мог бежать, а мог стоять на месте. Что-то подобное можно наблюдать и в среде SimInTech: справа от кнопки t (Часы) на рис. 9 расположена кнопка Δt, задающая интервалы времени, за которые создаются отдельные «кадры анимации» расчёта.

Но время может и не присутствовать в явном виде в расчёте. Пример тому – расчёт старта ракеты с учетом сопротивления воздуха (рис. 12) где используется блок расчёта параметров атмосферы Земли в зависимости от высоты над уровнем моря.

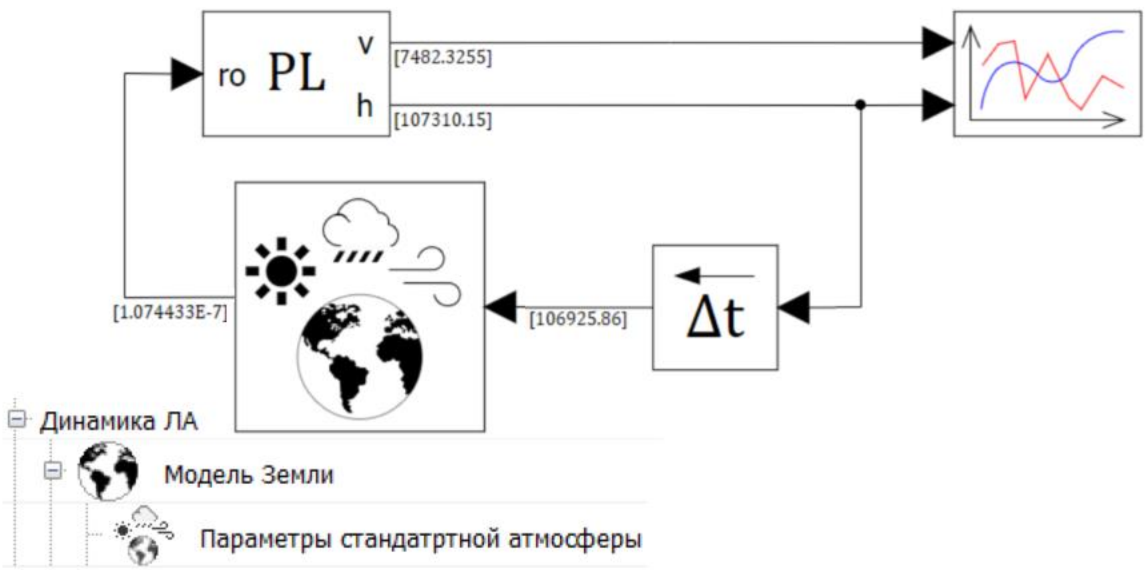
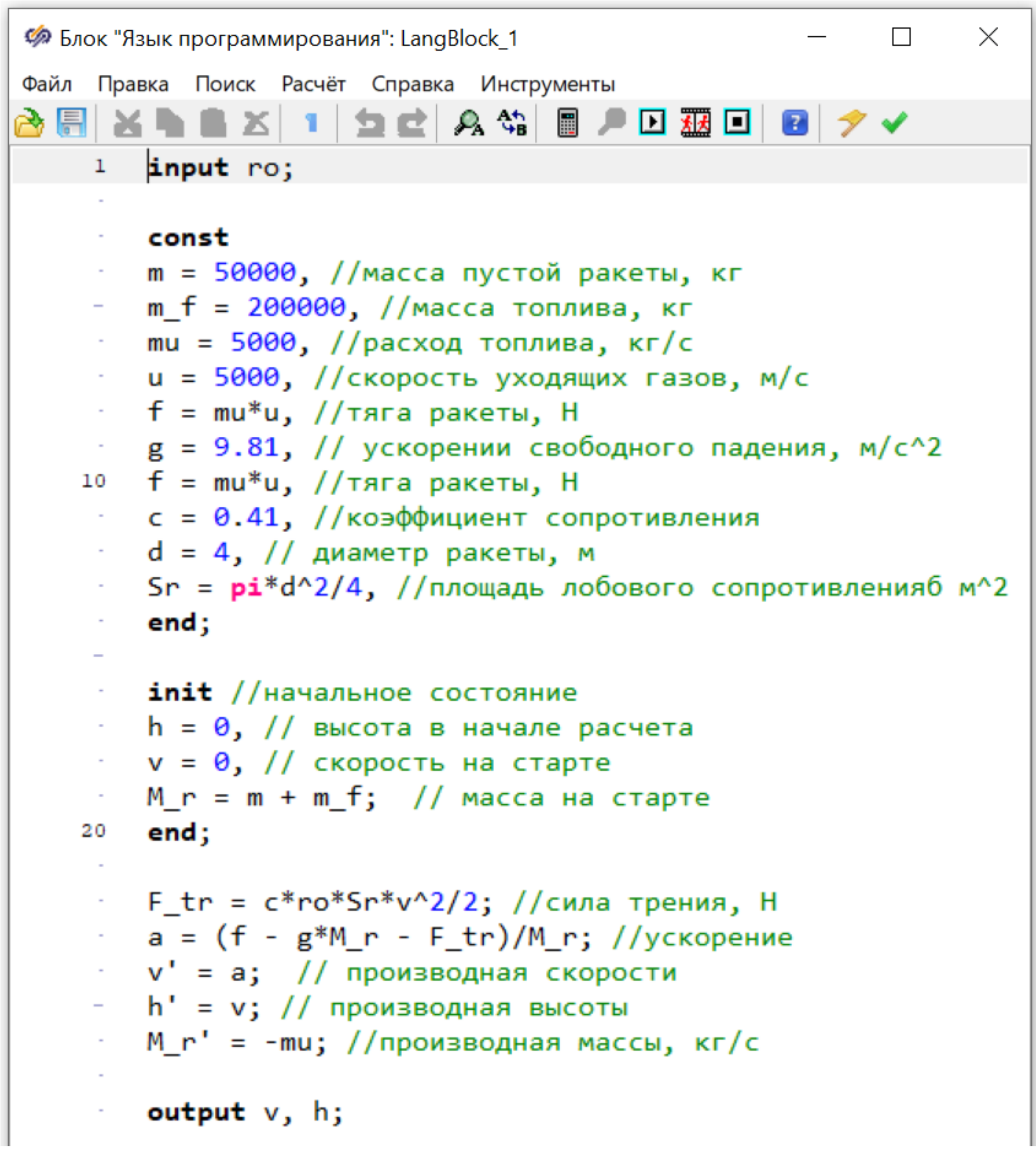


Рис. 12. Учёт сопротивления воздуха при старте ракеты

Программный блок в расчёте на рис. 12 содержит программу численного решения обыкновенного дифференциального уравнения, показанного на рис. 6. Время в расчёте на рис. 13 присутствует в производных: ускорение – это первая производная скорости от времени, а скорость – это производная пути от времени. Но лучше сказать так: путь – это интеграл (первообразная – антипроизводная) скорости по времени, а скорость – это интеграл ускорения по времени. Вот так через незаметный штрих у имён переменных в расчёт вклинилось время. Блок с изображением отрезка времени со стрелочкой наверху позволяет использовать в расчете не текущее значение высоты полета, а то, какое было в предыдущем шаге.




```
input ro;

const
m = 50000, //масса пустой ракеты, кг
m_f = 200000, //масса топлива, кг
mu = 5000, //расход топлива, кг/с
u = 5000, //скорость уходящих газов, м/с
f = mu*u, //тяга ракеты, Н
g = 9.81, // ускорении свободного падения, м/с^2
f = mu*u, //тяга ракеты, Н
c = 0.41, //коэффициент сопротивления
d = 4, // диаметр ракеты, м
Sr = pi*d^2/4, //площадь лобового сопротивленияб м^2
end;

init //начальное состояние
h = 0, // высота в начале расчета
v = 0, // скорость на старте
M_r = m + m_f;  // масса на старте
end;

F_tr = c*ro*Sr*v^2/2; //сила трения, Н
a = (f - g*M_r - F_tr)/M_r; //ускорение
v' = a;  // производная скорости
h' = v; // производная высоты
M_r' = -mu; //производная массы, кг/с

output v, h;
```

Рис. 13. Программа численного решения

Блок-схема полёта двухступенчатой ракеты с возвратом первой ступени для повторного использования показана на рис. 14. Рисунок 15 – это изменение по времени высоты подъёма и спуска первой ступени ракеты.

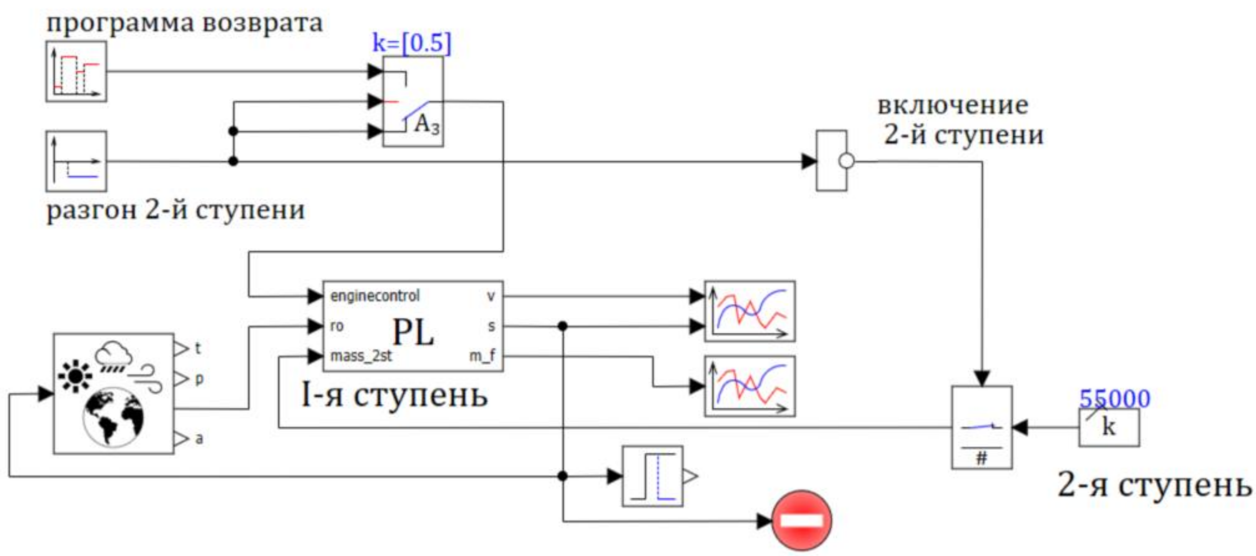


Рис. 14.

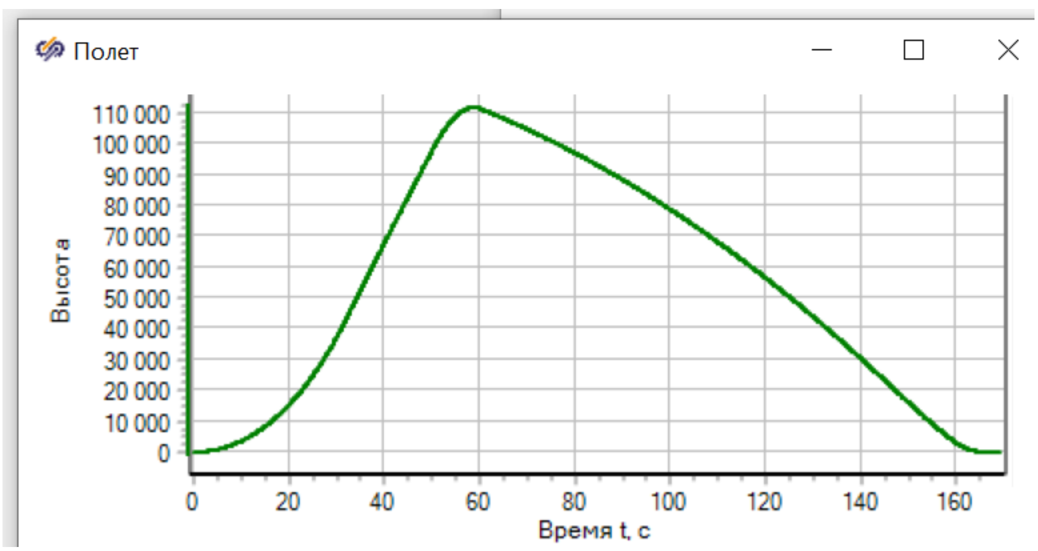


Рис. 15. График полёта и возврата на Землю первой ступени ракеты