

Глава 16. RARS, простой гоночный симулятор

Если вы читали предыдущие главы, Вы знаете что цель моей работы – создать гоночный симулятор. Я несколько раз обдумывал написать новый симулятор либо начать со стороннего уже работающего симулятора. Десять лет назад, когда я начал писать этот сборник, выбор был прост, потому что готовых симуляторов не было. С тех пор ситуация поменялась, сейчас есть по крайней мере одна отлично сделанная программа в свободном доступе.

Вывод результатов имеет неоспоримые преимущества, но противоречит одной из далеко идущих целей этого сборника. То есть создание больше полностью оригинальной работы, чем повторение информации, которую Вы можете почерпнуть в других источниках. Кроме того, было бы нелепо открыть законы Ньютона заново, поэтому я взял их как данность. Похожим образом я заключил, что было бы нелепо изобретать устройство симулятора. Это было бы слишком большим отступлением от Физики Гонок, и для этого надо было бы выполнить следующие работы:

- Управление памятью, окнами, графикой, выводом данных и т.д.;
- Языки программирования, создание скриптов, объектные технологии;
- Технологии симуляции, временная дискретизация, событийность, динамические решения;
- Структуры данных для описания трасс и машин
- Выбор системы координат.

Всё это, хотя и интересно, но не относится к физике. Более того, сейчас это всё уже более или менее обыденно. Для нас не слишком важно, что выбрать по этим пунктам, если есть уже базовые платформы, в которых выбор уже обоснованно сделан. Поэтому, лишь с небольшими колебаниями я сделал выбор начать с уже существующей программой. Мой выбор это RARS – Robot Auto Racing Simulator. Это отличная, и неожиданно простая платформа для программистов для экспериментов с робототехникой. Его назначение это поддержка распределенных виртуальных гоночных соревнований, в которой участники создают роботов-пилотов, с которыми и участвуют в соревнованиях. Последнее соревнование, которое я смог найти было проведено в 1999 году. Это не до предела реалистичный симулятор, и он никогда не предполагался таковым. Его физика упрощена таким образом, чтобы основным занятием соревнующихся было созданием роботов, а не борьба с детальной и реалистичной физикой. В этот симулятор включена работающая инфраструктура и достаточно информации описывающей знаменитые трассы. В конце концов, до сих пор RARS бесплатен.

Упрощения, сделанные в RARS, делают его отличной стартовой точкой для улучшения физики без необходимости изобретать второстепенные стороны гоночного симулятора. Отметим, что RARS создавался с открытым кодом, и изначально программа была сделана такой, чтобы её было легко модифицировать. Обычно модификация для соревнующихся заключается в создании новых роботов. Несмотря на это изменять физику настолько просто, насколько мне это необходимо. Теперь, когда у меня есть программа, мне необходимо изменить или роботов, или возможно создать свою новую, публичную гоночную серию, где новых роботов будут писать участники. Только время покажет, что будет работать лучше. Как обычно, я буду делать изменения пошагово, не отклоняясь слишком сильно от работающей базы. Эта стратегия не только позволит держать изменения под контролем, но ещё также позволит мне объяснять, что происходит шаг за шагом. Поэтому я создам копию исходных кодов, изменю название на RARSEP (RARS Enhanced Physics – RARS, улучшенная физика). Я буду выкладывать исходные коды с моими изменениями в интернете, чтобы сделать проект доступным для обновления.

Моя первая цель использования RARSEP это нахождение оптимальной гоночной траектории. В частности, мне необходимо найти способ ответить на вопросы о гоночной траектории, такие как кратчайшая траектория или траектория с наивысшей скоростью в определенном элементе трассе обеспечит кратчайшее время прохождения всего трека с этим элементом. Этот вопрос это часть «чтения трассы» - одной из задач любого пилота. На практике это метод проб и ошибок включающий искусство, опыт и эксперименты.

Для примера, недавно на треке я занимался с двумя инструкторами, которые провели именно на этом треке много часов и которые спорили об одной связке медленных поворотов. После некоторого времени споров и насилования доски они пришли к мнению, что классическая траектория, которой они следовали годами, вероятно, не является самой быстрой. Это порадовало автокроссеров и они решили, что автокроссерская траектория вероятно быстрее, чем классическая траектория.

Автокроссеры тратят много времени и усилий для нахождения быстреей траектории в медленных поворотах, в то время когда классические пилоты ищут быстреей способ пройти быстрые повороты. Нет конца чтению материалов в поддержку классической гоночной траектории, заходить в поворот настолько широко насколько это возможно, трейл-брейкинг, возвращение к ускорению в первой половине поворота, нажатие на газ, поздний апекс итд. Чаще всего, автокроссеры считают, что быстрее просто цепляться за внутреннюю часть поворота, что обеспечит быстреее время. Почему? Есть ли этому, какое то научное объяснение?

Может быть и те и другие правы? Или может быть и те и другие ошибаются? Что насчет промежуточных случаев со среднескоростными поворотами?

Это пример вопроса, который мы хотим решить при помощи программы симулятора. Было интересно, что инструкторы обсуждают медленные повороты, но никто не обсуждал, что быстрые повороты должны проходить классически. Существует предположение, что медленные повороты это те повороты, которые не намного больше машины. Может ли быть так, что классическая гоночная траектория не оптимальна? Опытные автокроссеры, когда проходят такие повороты даже не задумываются и входят внутри и плотно, или бросают и ловят машину (toss-and-catch the car). Инструкторы должны, следуя классической теории, проходить повороты широко, заходя поздно, и тем самым теряя время на то чтобы следовать классической гоночной траектории в поворотах которые не намного больше машины. Но возможно ли что классическая теория не лучший способ прохождения медленных поворотов? Возможно, здесь есть другие факторы? Возможно ли что размер поворота будет такой, что невозможно будет различить «автокроссовый» поворот от «классического»?

Будет ли играть роль контекст, близок ли поворот к другим поворотам или близок к прямым? Кажется, что даже очень опытные пилоты на некоторых трассах иногда открывают новые способы улучшить траекторию. Некоторые из этих улучшений зависят от меняющихся условий, таких как погода, или определенная машина или её настройка. На многих трассах есть каноническая «дождевая траектория» которая отличается от «сухой траектории» Я ещё бы поставил на то, что машины кубка Винстона ездят по другим траекториям в Сирс Поинт и Воткинс Глен, чем машины с граунд эффектом и большой прижимной силой, такие как формулы. Некоторые улучшения траектории будут настолько глубоки, постоянны и не меняться что буду ускользать от пилотов при анализе.

Вкратце, то, что мы собираемся решить сначала с RARSEP это способ найти ответы на эти вопросы. Я начну с Windows порта RARS версии 074, который вы можете взять в исходных кодах по адресам:

<http://users.skynet.be/mgueury/rars/rars.html>

<http://rars.sourceforge.net/>

Я выбрал порт Windows, поскольку это наиболее удобно для меня. У меня уже есть работающие системы разработки для Windows, в то время как для других платформ мне придется тратить деньги и время. Исходный код RARS мультиплатформенный, и может быть использован и под Linux и под Windows. Код очень хорошо разбит на части, таким образом, что платформозависимые биты отделены от платформонезависимых. Всё что я собираюсь делать находится в платформонезависимой части программы, и должно собираться для всех платформ. Тем не менее, у меня не будет возможности проверить мои изменения на всех платформах – Физика Гонок это не упражнение по разработке программного обеспечения. Хотя я и не собираюсь делать изменения которые не будут работать под другими платформами существует риск что я могу по невнимательности сделать что некоторые файлы будут требовать некоторой доработки на других платформах. Я прошу моих читателей сообщить мне, если им будет об этом известно. Веб сайты содержат очень подробное описание, как собрать и запустить программу и написать роботов. Для написания робота необходимо понимание существующей физической модели RARS. Подобно этому, для улучшения физики нам необходимо знать эти-же вещи. Теперь очевидно, что лучший способ улучшить физику пошагово будет заключаться в написании роботов, но этот подход может измениться со временем. Предмет этой установки Физики Гонок, таким образом, заключается в во введении в существующую физику RARS с планированием её возможных улучшений. Я очень благодарен авторам за RARS и надеюсь, им понравится, что я делаю с их работой. Программа очень проста в установке, запуске, понимании и улучшении. Я рекомендую вам скачать её и следовать за мной в её улучшении. Несмотря на это мои статьи самодостаточны, и вам не нужно будет собирать и запускать RARS для того чтобы понять что я делал.

Я нашел другой независимый проект улучшения RARS. Он называется TORCS и его можно найти по адресу <http://torcs.free.fr> Он включает в себя некоторые улучшения, которые я собирался сделать, но его цели близки к целям RARS, а не моим. Он выглядит многообещающе, но в нем есть три особенности, которые делают его неподходящим для меня.

- Он незакончен, в то время как RARS полностью функционален.
- Он предназначен для Linux. У меня нет средств разработки для Linux, и для меня займет много времени и денег получить их сейчас.
- Как обычно, слишком сильное использование чужих наработок делает тему для меня не интересной

Тем не менее, я буду следить за TORCS. Он может стать очень грозным!

Мой первый подход в адаптации RARS для поиска оптимальной траектории это написание робота, который может обучаться поиску новых траекторий, делая небольшие изменения в траектории на каждом круге, как это делал бы пилот-человек. Это один из видов вариативного подхода, общепринятого в физике. Робот для поиска траекторий (РПТ) будет строить внутреннюю память текущей траектории и всего что он обнаружит на треке. Затем он будет улучшать траектории, и если время круга уменьшается продолжать улучшать траекторию в том же направлении, в ином случае он будет сбрасывать улучшения и пробовать другие способы. Для уменьшения возвратов он будет начинать улучшения с другой части траектории.

Робот будет расходовать много вычислительных ресурсов и не будет конкурентоспособным в RARS. Несмотря на это, напомним что в RARSEP у нас другие цели. Также, этот план может потребовать значительных затрат времени и растянутся на несколько статей. Он может вообще не заработать. Как обычно я расскажу Вам об том, как это будет проходить.

Давайте рассмотрим существующую физическую модель. Алгоритм RARS дьявольски прост, это правильный компромисс между физичностью достаточной для того чтобы не слишком сильно усложнить написание роботов. Каждый шаг симулятор дает каждому роботу **ситуационную структуру**, и робот отвечает командой или управляя структурой. Ситуационная структура состоит из текущего положения и скорости автомобиля относительно трассы, стены и другие автомобили.

Управляющая структура объявляет желательный **угол увода** – который грубо представляет собой угол поворота рулевого колеса и желательную скорость – которая грубо представляет собой ускорение (положительная) или торможение (отрицательная). Управление взаимодействует с дорогой посредством фрикционной модели шины, создавая силу, которая ускоряет автомобиль. Сила ограничена мощностью выдаваемой двигателем, поэтому не всегда возможно, что *вся* сила, которую *могут* передать шины будет приложена, поскольку двигатель может её не предоставить. Поэтому желаемая скорость это не та скорость, которая будет достигнута.

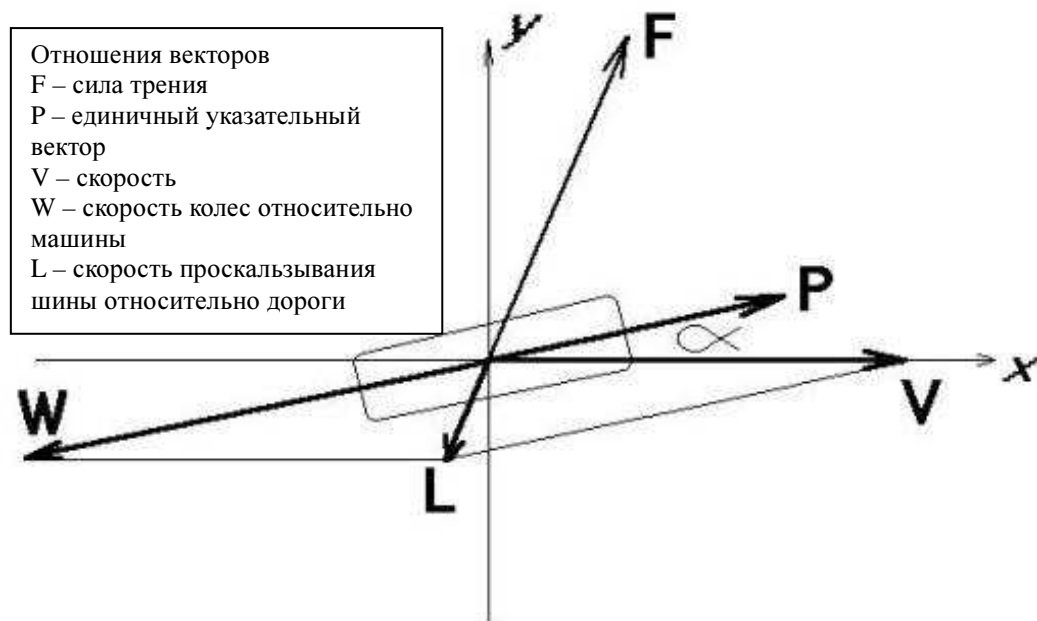
Одна из причин, по которой RARS прост это то, что он двумерный. В нем существует три правосторонних системы координат. Первая это **земля**, почти инерциальная система отсчета неподвижная относительно дороги. Силы и ускорения рассчитанные в этой системе инерциальны. Вторая система отсчета это **автомобиль**. Ось *x* автомобиля указывает вперед, а ось *y* – указывает на левую сторону водителя.

Третья и последняя система отсчета связана с **путем**, и вектором скорости автомобиля. **Касательный** компонент любого вектора направлен по оси *X* **пути**, и **перпендикулярный** компонент вектора направлен по оси *Y*. **Автомобильная система отсчета** будет совпадать с **путевой** только тогда когда угол увода будет равен 0.

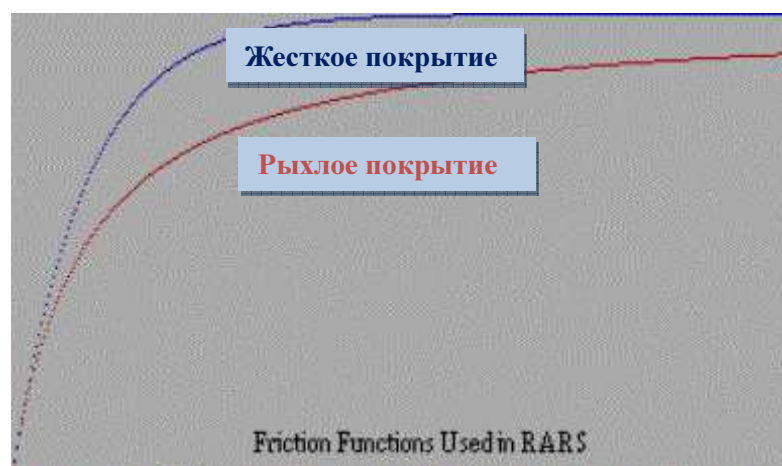
Следующая таблица, это адаптированная документация программы, суммирующая физическую модель:

<i>V</i>	Вектор скорости автомобиля по отношению к земле
<i>v</i>	Наземная скорость – модуль <i>V</i>
<i>P</i>	Вектор указывающий вперед автомобиля
α	Угол увода (выходная команда робота), которая разделяет <i>P</i> и <i>V</i> . А положительная когда автомобиль указывает влево от <i>V</i> , при сносе в левом повороте.
<i>W</i>	Вектор скорости пятна контакта резины относительно машины. Всегда указывает обратно направлению <i>x</i>
<i>vc</i>	желаемая скорость. Выходная команда указывающая вперед автомобильной системы; $W = -P * vc$
$L = V + W = V - P * vc$	Вектор проскальзывания, скорость пятна контакта относительно дороги
<i>Lt</i>	Касательный к пути компонент $L = v - vc * \cos(\alpha)$
<i>Ln</i>	Перпендикулярный к пути компонент $L = v - vc * \sin(\alpha)$
<i>l</i>	Скорость проскальзывания – модуль <i>L</i>

$Q = L/l$	Единичный вектор в направлении L
$\mu(l)$	Коэффициент сцепления, зависящий только от скорости проскальзывания
$F = -Q * \text{масса} * \mu(l)$	Вектор силы толкающей автомобиль в направлении противоположном L
$f = \text{масса} * \mu(l)$	Модуль F
F_t	Касательный к пути компонент $F = -f * L_t/l$
F_n	Перпендикулярный к пути компонент $F = -f * L_n/l$
F_{tP}	Проекция F_t на систему автомобиля $= F_t * \cos(\alpha)$
F_{nP}	Проекция F_n на систему автомобиля $= F_n * \sin(\alpha)$
P_{wr}	Расходуемая двигателем мощность = сумме сил компонентов вдоль P ограниченной мощностью двигателя. Максимально 181 л.с. $(F_{tP} + F_{nP}) * v_c$



Используемая функция трения имеет форму $\mu(l) = F_{MAX} * 1 / (K+1)$, где F_{MAX} и K константы



Суммируем ограничения существующей модели

- Трасса. Плоская, фиксированной ширины, нет неровностей.
- Машина. Материальная точка, нет подвески.

Планируемые улучшения:

- Трасса. Изменения высоты, изменение ширины, бордюры, бэнкинг, корона, профиль, кочки
- Машина. Четыре колеса, дискретная трансмиссия, переключение скоростей, подвеска: пружины, амортизаторы, аэродинамика.

По мере продвижения, будет полезно сохранить эти странички. Мы будем часто обращаться к ним.