Отчет по лабораторной работе №8

по дисциплине: Математическое моделирование

Ким Михаил Алексеевич

Содержание

# 1 Цель работы

Продолжить знакомство с функционалом языка программирования Julia, дополнительных библиотек (DifferentialEquations, Plots), интерактивного блокнота Pluto, а также интерактивной командной строкой REPL. Продолжить ознакомление с языком моделирования Modelica и программным обеспечением OpenModelica. Используя эти средства, описать математическую модель конкуренции двух фирм.

# 2 Задание

Рассмотреть два случая конкуренции двух фирм: в первом случае борьба между фирмами ведется только рыночными методами. Во втором случае, помимо экономических факторов, борьба ведется при помощи социально-психологических факторов.

1. Построить графики изменения оборотных средств фирмы 1 и фирмы 2 без учета постоянных издержек и с веденной нормировкой для случая 1.
2. Постройте графики изменения оборотных средств фирмы 1 и фирмы 2 без учета постоянных издержек и с веденной нормировкой для случая 2.

# 3 Теоретическое введение

## 3.1 Актуальность

Математические модели конкуренции двух фирм широко используются для изучения поведения рынка и принятия бизнес-решений.

Модели конкуренции позволяют предсказать, как изменения в цене, спросе, затратах и других факторах влияют на прибыль и рыночную долю фирмы. Они также могут помочь определить оптимальные стратегии для достижения конкурентного преимущества на рынке.

Математические модели конкуренции двух фирм могут быть полезными для анализа различных сценариев и прогнозирования результатов, что может помочь фирмам принимать более обоснованные и продуманные решения. Они могут быть использованы для определения оптимальной цены продажи товара, управления запасами, оценки рисков и многого другого.

В целом, математические модели конкуренции позволяют более точно понимать конкуренцию на рынке и разрабатывать стратегии для улучшения позиций фирмы на нем [1].

## 3.2 Предварительные замечания

Перед рассмотрением модели уместно сделать ряд предварительных замечаний:

1. Конкуренция имеет место между производителями взаимозаменяемых, часто однотипных, товаров.
2. Производители жизненно необходимых товаров, как правило, контролируются либо государством и часто являются естественными монополиями. В этих условиях роль конкуренции существенно снижается. Математические модели, рассматриваемые в данной лабораторной работе, не используются для моделирования конкуренции фирм, производящих данные товары.
3. Важным фактором конкуренции является качество товара. Понятие «качество» включает множество факторов: долговечность, прочность, удобство в эксплуатации, эстетика, и т.п. При этом первостепенным фактором при оценке товара является не само его качество, а отношение его цены к качеству. Для каждой страны данный фактор будет являться разным в связи с особенностью культуры, достатком граждан.
4. Производители принципиально новых товаров, не имеющих в данный момент времени взаимозаменяемых аналогов, создают свою рыночную нишу. Конкуренция в ней возникает, когда в неё внедряются другие производители. При этом, «качество» конечного товара может и не меняться, но себестоимость его снижается.
5. Очень важную роль при конкуренции играет реклама. По существу, речь идет о формировании общественного мнения, о преимуществах того или иного товара. Строго говоря, эта задача выходит за рамки экономических и связана с более общей проблемой: возникновения, эволюции и борьбы условных информаций.
6. Вступая в конкурентную борьбу, предприниматель может ставить следующие цели:
   * полностью вытеснить конкурента из определенной рыночной ниши;
   * Обеспечить себе определенную долю потребителей в условиях сосуществования с конкурентом (наиболее распространенный вариант);
   * Войти в рынок. Эта цель актуальна, если рынком владеет экономически сильный (обладающий большими средствами) конкурент, но не использующий инноваций.

Среди методов конкурентной борьбы можно условно выделить следующие группы.

1. **Чисто экономические (рыночные) методы**, не влияющие прямо на конкурента, но влияющие на рыночную цену. К ним относятся: сокращение производственного цикла, снижение себестоимости продукта. В компетенцию фирмы входит также и качество товара. Однако, как отмечалось выше, понятие «качества» многогранно и условно. Важно, что рыночная цена товара устанавливается в результате баланса спроса и предложения. Влиять на неё предприниматель может, только изменяя объем производства. В этом случае конкуренты непосредственно не взаимодействуют и получают информацию друг о друге через ситуацию на рынке. Эта модель в вербальной форме была рассмотрена Курно.
2. **Финансовые методы конкуренции**. Имеются в виду случаи, когда один из партнеров «назначает» низкую цену своего товара (ниже себестоимости), и в результате конкурент разоряется. Такой метод имеет специальное название – демпинг. Речь идет о наводнении рынка товаром, в результате чего рыночная цена опускается ниже уровня себестоимости товара конкурента. При этом оба конкурента терпят убытки, и вопрос заключается в том, кто из них раньше разорится. Ясно, что на демпинг может решиться конкурент, обладающий запасом средств, которые он использует для дотаций своего производства в течение большого (но не бесконечного) времени. В целом, эта акция может иметь смысл, если в результате ее конкурент полностью вытесняется с рынка.
3. **Методы, выходящие за рамки чисто экономических**. Легальным методом такого типа является реклама. Не меньшую роль играет антиреклама, то есть, создание негативного отношения к товару конкурента. Формально она запрещена, но реально всегда имеет место даже вне зависимости от действий предпринимателя. К этой же группе относятся криминальные методы [2].

## 3.3 Модель одной фирмы

ОДУ, представляющая собой модель изменения числа оборотных средств одного предприятия:

где — оборотные средства предприятия; — доля оборотных средств, идущая на покрытие переменных издержек; — себестоимость продукта; — рыночная цена товара; — длительность производственного цикла; — число потребителей производимого продукта; — максимальная потребность одного человека в продукте в единицу времени; — постоянные издержки, не зависящие от количества выпускаемой продукции.

## 3.4 Модель конкуренции двух фирм

### 3.4.1 Случай 1

Рассмотрим случай конкуренции между двумя фирмами, которые ведут борьбу только рыночными методами (конкуренты могут влиять на противника только путем изменения параметров своего производства). При этом товары, производимые обоими фирмами, имеют одинаковое качество, находятся в одной рыночной нише, а у потребителей нет априорных предпочтений, товар какой фирмы выбирать. В этом случае, на рынке устанавливается единая цена, которая определяется балансом суммарного предложения и спроса.

Система уравнений для первого случая принимает вид:

где — нормировка времени (безразмерное время), , , , , .

При этом считается, что ценовое равновесие устанавливается быстро, а постоянные издержки пренебрежимо малы.

Также заметим, что указаны в тысячах единиц, а значения — в миллионах единиц.

### 3.4.2 Случай 2

Рассмотрим случай конкуренции между двумя фирмами, при котором, помимо рыночной борьбы, компаниями используются еще и социально-психологические факторы — формирование общественного предпочтения одного товара другому, не зависимо от их качества и цены.

В данном случае система уравнений принимает вид:

где все обозначения остаются прежними, а коэффициент, появляющийся во втором слагаемом в первом уравнении, отвечает за социально-психологические факторы [3].

# 4 Выполнение лабораторной работы

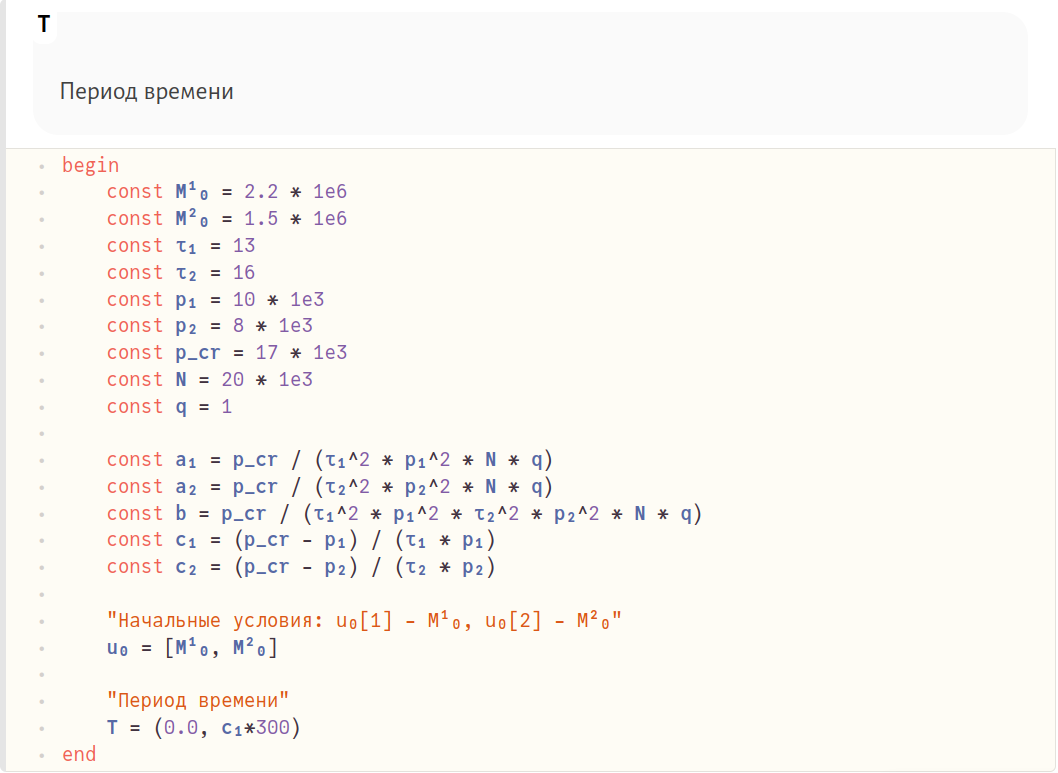
## 4.1 Pluto.jl

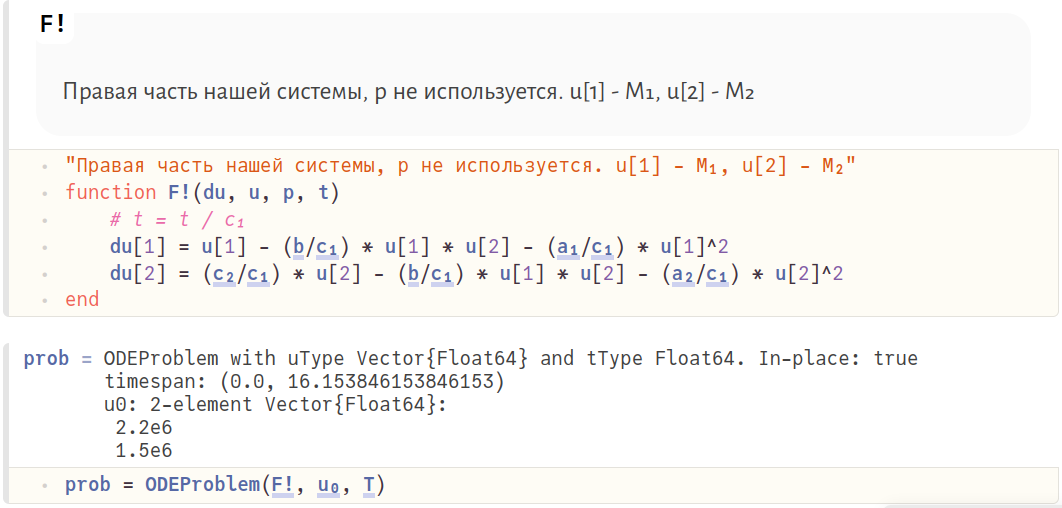
### 4.1.1 Задание №1

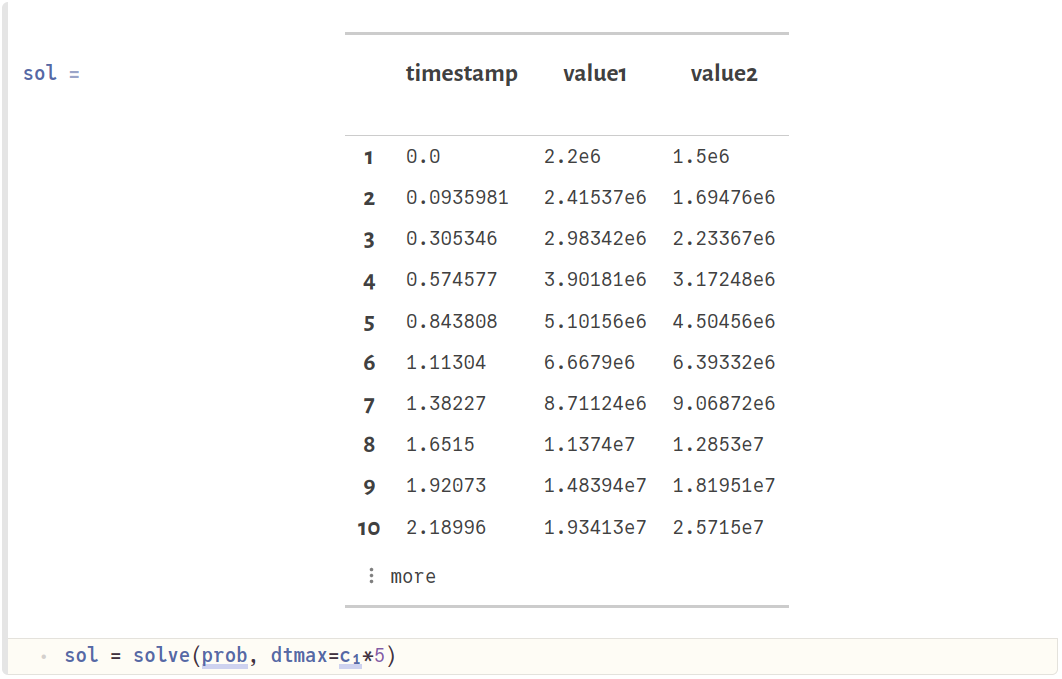
1. Пишем программу, воспроизводящую модель на языке программирования Julia с использованием интерактивного блокнота Pluto (рис. [1](#fig:01), [2](#fig:02), [3](#fig:03), [4](#fig:04), [5](#fig:05), [6](#fig:06)).

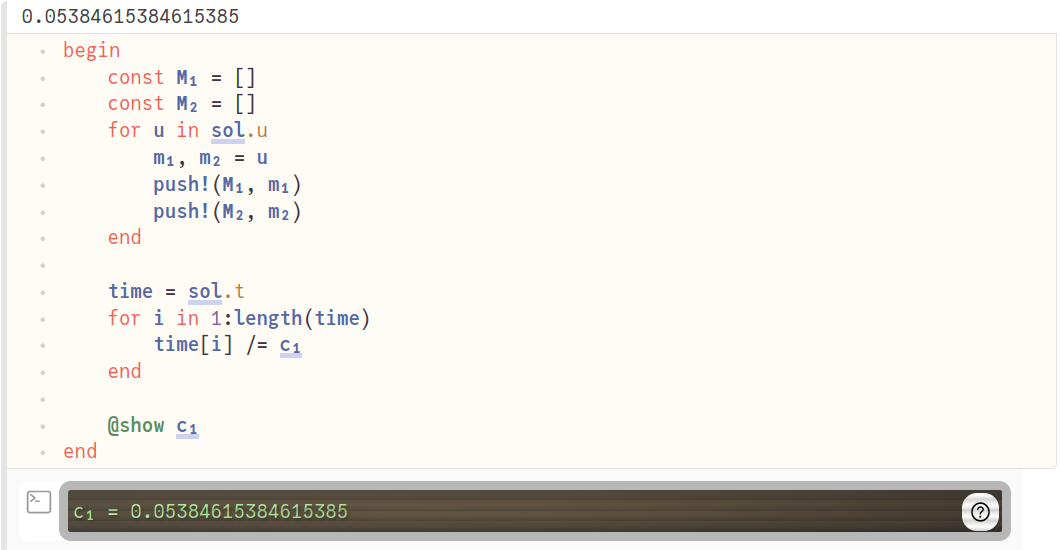
* begin  
   import Pkg  
   Pkg.activate()  
   using DifferentialEquations  
   using LaTeXStrings  
   import Plots  
  end
* begin  
   const M¹₀ = 2.2 \* 1e6  
   const M²₀ = 1.5 \* 1e6  
   const τ₁ = 13  
   const τ₂ = 16  
   const p₁ = 10 \* 1e3  
   const p₂ = 8 \* 1e3  
   const p\_cr = 17 \* 1e3  
   const N = 20 \* 1e3  
   const q = 1  
    
   const a₁ = p\_cr / (τ₁^2 \* p₁^2 \* N \* q)  
   const a₂ = p\_cr / (τ₂^2 \* p₂^2 \* N \* q)  
   const b = p\_cr / (τ₁^2 \* p₁^2 \* τ₂^2 \* p₂^2 \* N \* q)  
   const c₁ = (p\_cr - p₁) / (τ₁ \* p₁)  
   const c₂ = (p\_cr - p₂) / (τ₂ \* p₂)  
    
   "Начальные условия: u₀[1] - M¹₀, u₀[2] - M²₀"  
   u₀ = [M¹₀, M²₀]  
    
   "Период времени"  
   T = (0.0, c₁\*300)  
  end
* "Правая часть нашей системы, p не используется. u[1] - M₁, u[2] - M₂"  
  function F!(du, u, p, t)  
   # t = t / c₁  
   du[1] = u[1] - (b/c₁) \* u[1] \* u[2] - (a₁/c₁) \* u[1]^2  
   du[2] = (c₂/c₁) \* u[2] - (b/c₁) \* u[1] \* u[2] - (a₂/c₁) \* u[2]^2  
  end
* prob = ODEProblem(F!, u₀, T)
* sol = solve(prob, dtmax=c₁\*5)
* begin  
   const M₁ = []  
   const M₂ = []  
   for u in sol.u  
   m₁, m₂ = u  
   push!(M₁, m₁)  
   push!(M₂, m₂)  
   end  
    
   time = sol.t  
   for i in 1:length(time)  
   time[i] /= c₁  
   end  
    
   @show c₁  
  end
* begin  
   fig = Plots.plot(  
   dpi=150,  
   grid=:xy,  
   gridcolor=:black,  
   gridwidth=1,  
   size=(800, 400),  
   legend=:outerbottom,  
   xlabel="θ = t/c₁",  
   ylabel="M₁(t), M₂(t)",  
   plot\_title="Модель конкуренции двух фирм. Случай 1")  
    
   Plots.plot!(fig[1], time, [M₁, M₂], color=[:blue :green], label=["M₁ — оборотные средства предприятия №1" "M₂ — оборотные средства предприятия №2"])  
  end

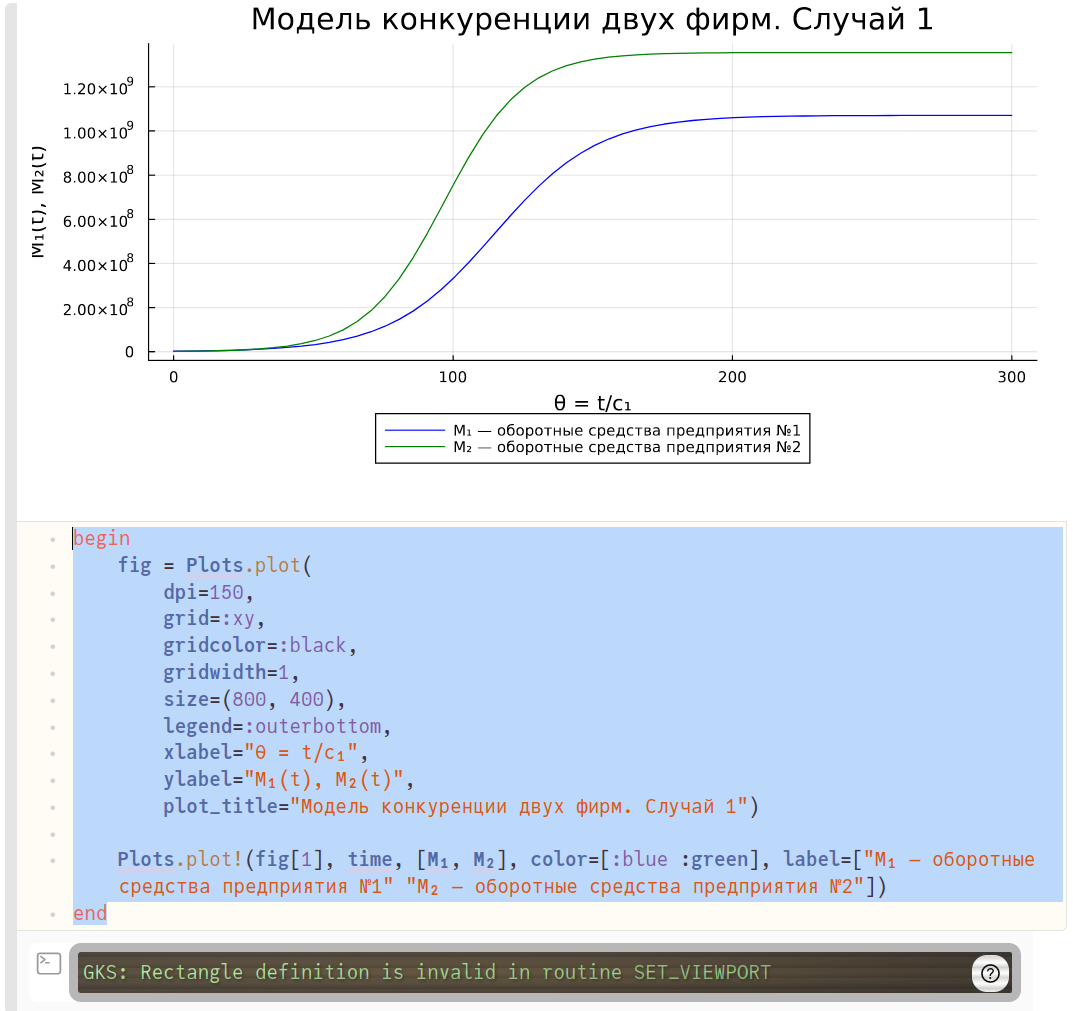
* 
* Figure 1: Импорт библиотек

* 
* Figure 2: Задание и вычисление коэффициентов с учетом единиц измерения, определение начальных условий и периода времени

* 
* Figure 3: Запись системы уравнений в виде функции. Постановка проблемы

* 
* Figure 4: Решение задачи (также задается максимальное значение шага относительно нормировки )

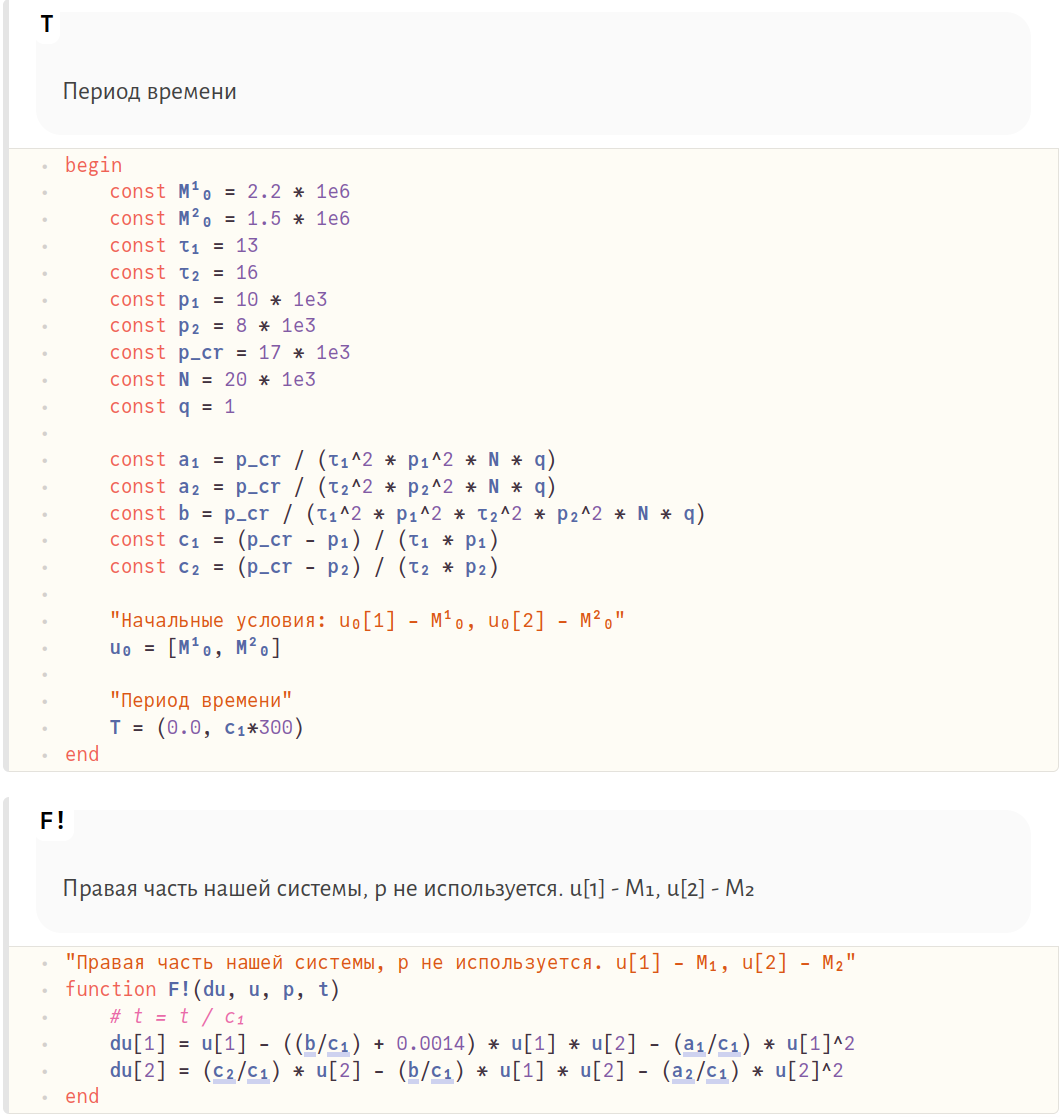
* 
* Figure 5: Формирование массивов, содержащих значения функций в момент времени . Формирование массива безразмерного времени ()

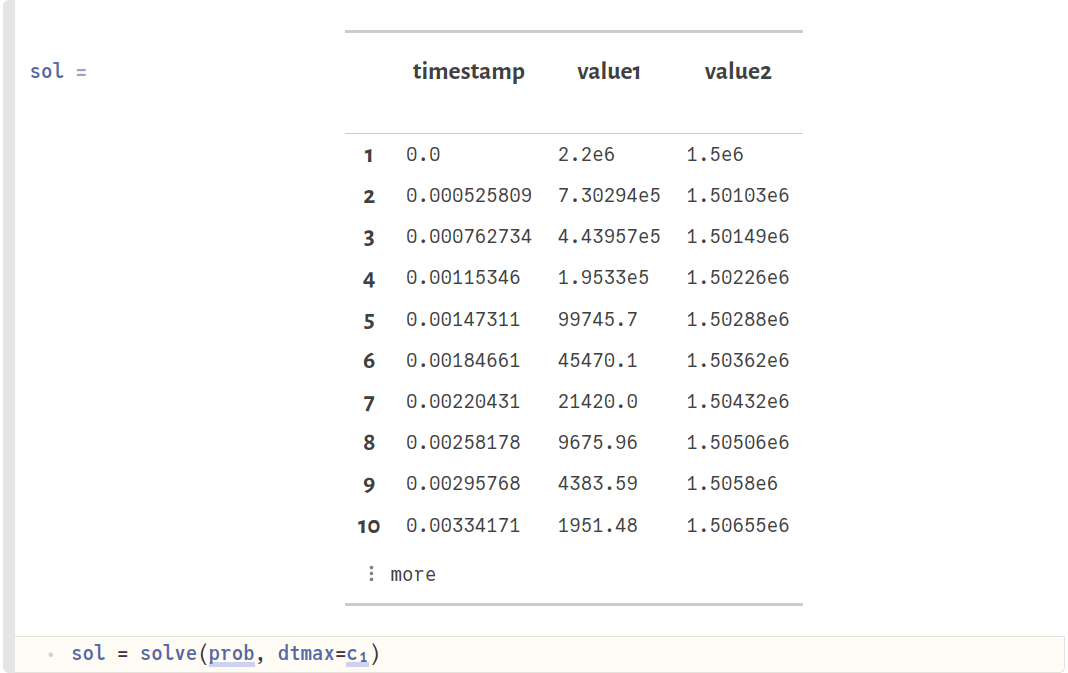
* 
* Figure 6: Отрисовка графика

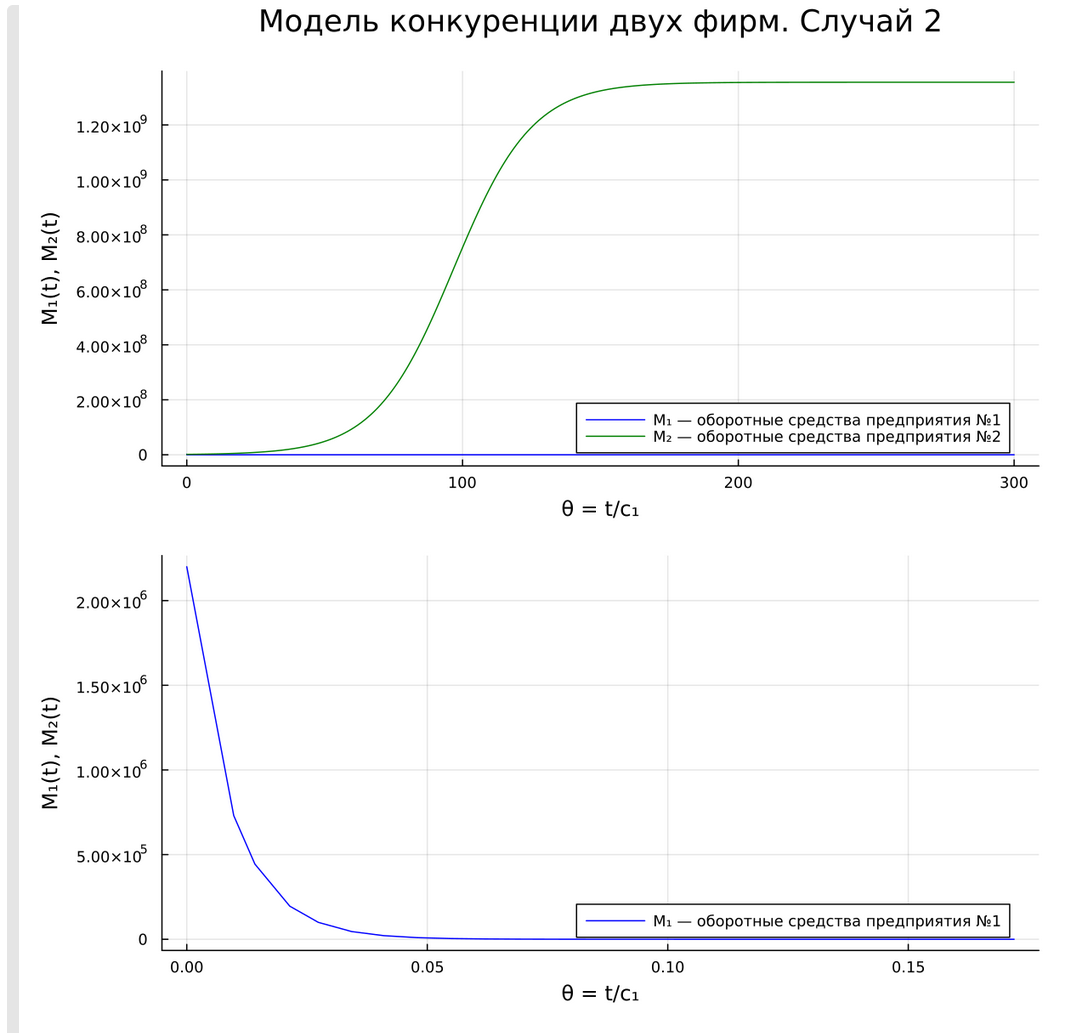
### 4.1.2 Задание №2

1. Изменены период времени T, первое уравнение в функции F!, максимальный размер шага при дифференцировании по времени dtmax. Остальные блоки кода оставлены без изменений. Любуемся результатом (рис. [7](#fig:07), [8](#fig:08), [9](#fig:09)).

* begin  
   "Период времени"  
   T = (0.0, c₁\*300)  
  end
* "Правая часть нашей системы, p не используется. u[1] - M₁, u[2] - M₂"  
  function F!(du, u, p, t)  
   # t = t / c₁  
   du[1] = u[1] - ((b/c₁) + 0.0014) \* u[1] \* u[2] - (a₁/c₁) \* u[1]^2  
   du[2] = (c₂/c₁) \* u[2] - (b/c₁) \* u[1] \* u[2] - (a₂/c₁) \* u[2]^2  
  end
* sol = solve(prob, dtmax=c₁)

* 
* Figure 7: Изменение периода времени T и первого уравнения в функции F!

* 
* Figure 8: Изменение шага разбиения dtmax

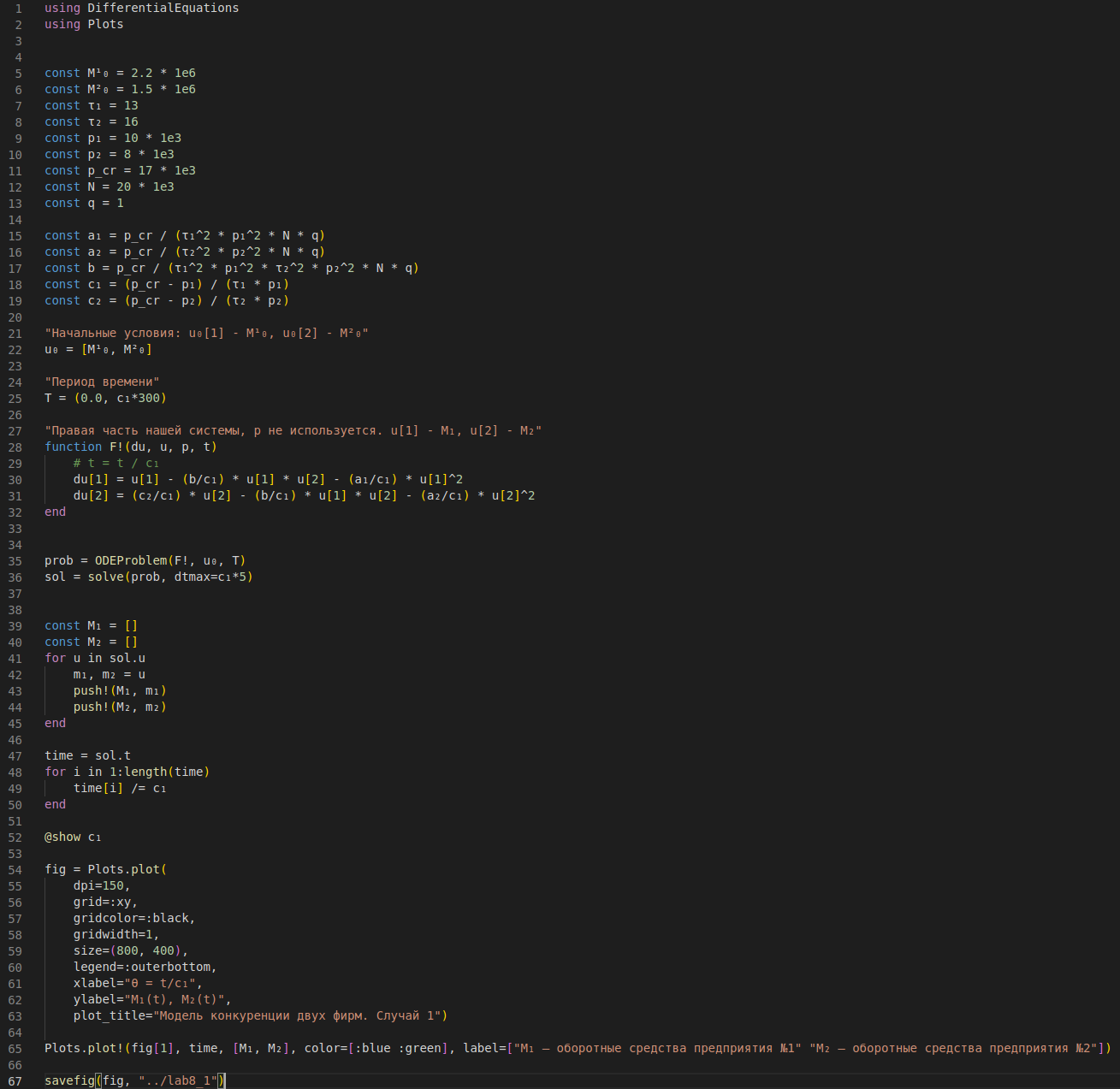
* 
* Figure 9: Результат в виде графиков. На втором графике показана динамика изменения функции на малом промежутке времени, т.к. на общем графике изменений не видно. Предприятие №1 почти сразу же терпит банкротство

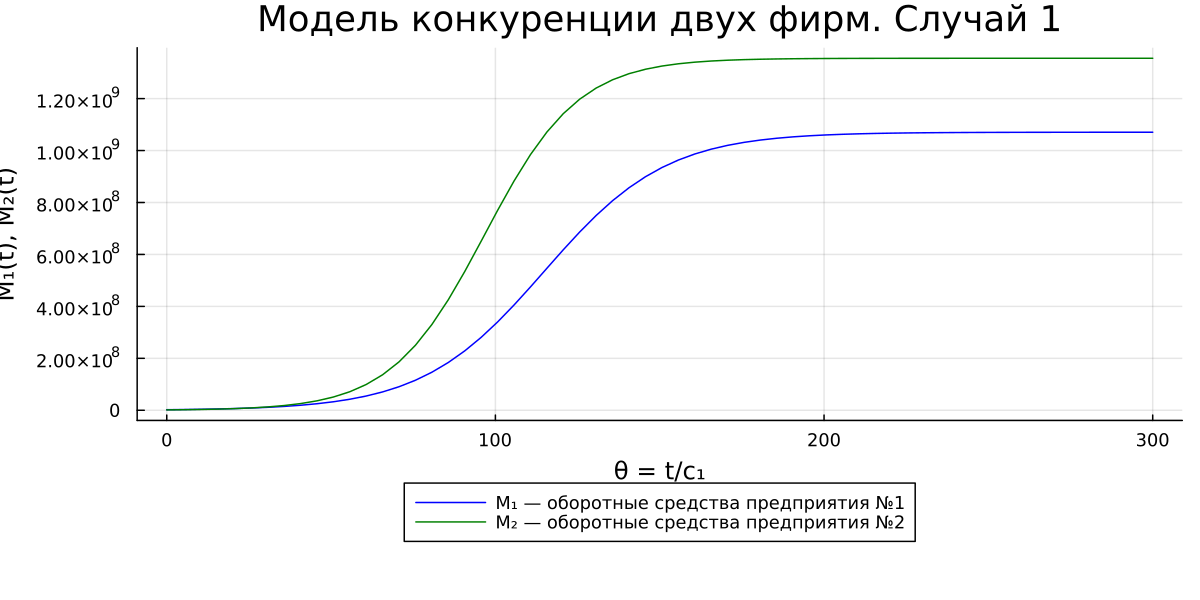
## 4.2 Julia

### 4.2.1 Задание №1

1. Код на Julia в файле аналогичен тому же, что написан с использованием Pluto (рис. [10](#fig:10), [11](#fig:001)). Единственные различия:
   * блоки перенесены в файл в виде построчного алгоритма без повторяющихся begin и end;
   * измененный синтаксис подключения библиотек;
   * выгрузка графиков в виде изображений при помощи метода savefig() в последней строчке кода.

* using DifferentialEquations  
  using Plots  
    
    
  const M¹₀ = 2.2 \* 1e6  
  const M²₀ = 1.5 \* 1e6  
  const τ₁ = 13  
  const τ₂ = 16  
  const p₁ = 10 \* 1e3  
  const p₂ = 8 \* 1e3  
  const p\_cr = 17 \* 1e3  
  const N = 20 \* 1e3  
  const q = 1  
    
  const a₁ = p\_cr / (τ₁^2 \* p₁^2 \* N \* q)  
  const a₂ = p\_cr / (τ₂^2 \* p₂^2 \* N \* q)  
  const b = p\_cr / (τ₁^2 \* p₁^2 \* τ₂^2 \* p₂^2 \* N \* q)  
  const c₁ = (p\_cr - p₁) / (τ₁ \* p₁)  
  const c₂ = (p\_cr - p₂) / (τ₂ \* p₂)  
    
  "Начальные условия: u₀[1] - M¹₀, u₀[2] - M²₀"  
  u₀ = [M¹₀, M²₀]  
    
  "Период времени"  
  T = (0.0, c₁\*300)  
    
  "Правая часть нашей системы, p не используется. u[1] - M₁, u[2] - M₂"  
  function F!(du, u, p, t)  
   # t = t / c₁  
   du[1] = u[1] - (b/c₁) \* u[1] \* u[2] - (a₁/c₁) \* u[1]^2  
   du[2] = (c₂/c₁) \* u[2] - (b/c₁) \* u[1] \* u[2] - (a₂/c₁) \* u[2]^2  
  end  
    
    
  prob = ODEProblem(F!, u₀, T)  
  sol = solve(prob, dtmax=c₁\*5)  
    
    
  const M₁ = []  
  const M₂ = []  
  for u in sol.u  
   m₁, m₂ = u  
   push!(M₁, m₁)  
   push!(M₂, m₂)  
  end  
    
  time = sol.t  
  for i in 1:length(time)  
   time[i] /= c₁  
  end  
    
  @show c₁  
    
  fig = Plots.plot(  
   dpi=150,  
   grid=:xy,  
   gridcolor=:black,  
   gridwidth=1,  
   size=(800, 400),  
   legend=:outerbottom,  
   xlabel="θ = t/c₁",  
   ylabel="M₁(t), M₂(t)",  
   plot\_title="Модель конкуренции двух фирм. Случай 1")  
    
  Plots.plot!(fig[1], time, [M₁, M₂], color=[:blue :green], label=["M₁ — оборотные средства предприятия №1" "M₂ — оборотные средства предприятия №2"])  
    
  savefig(fig, "../lab8\_1")

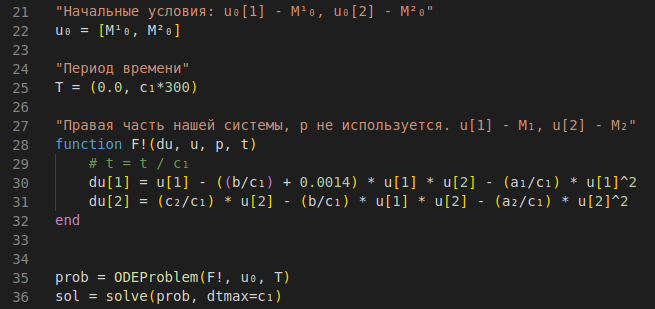
* 
* Figure 10: Код программы на Julia. Аналогичен коду задания для Pluto.jl

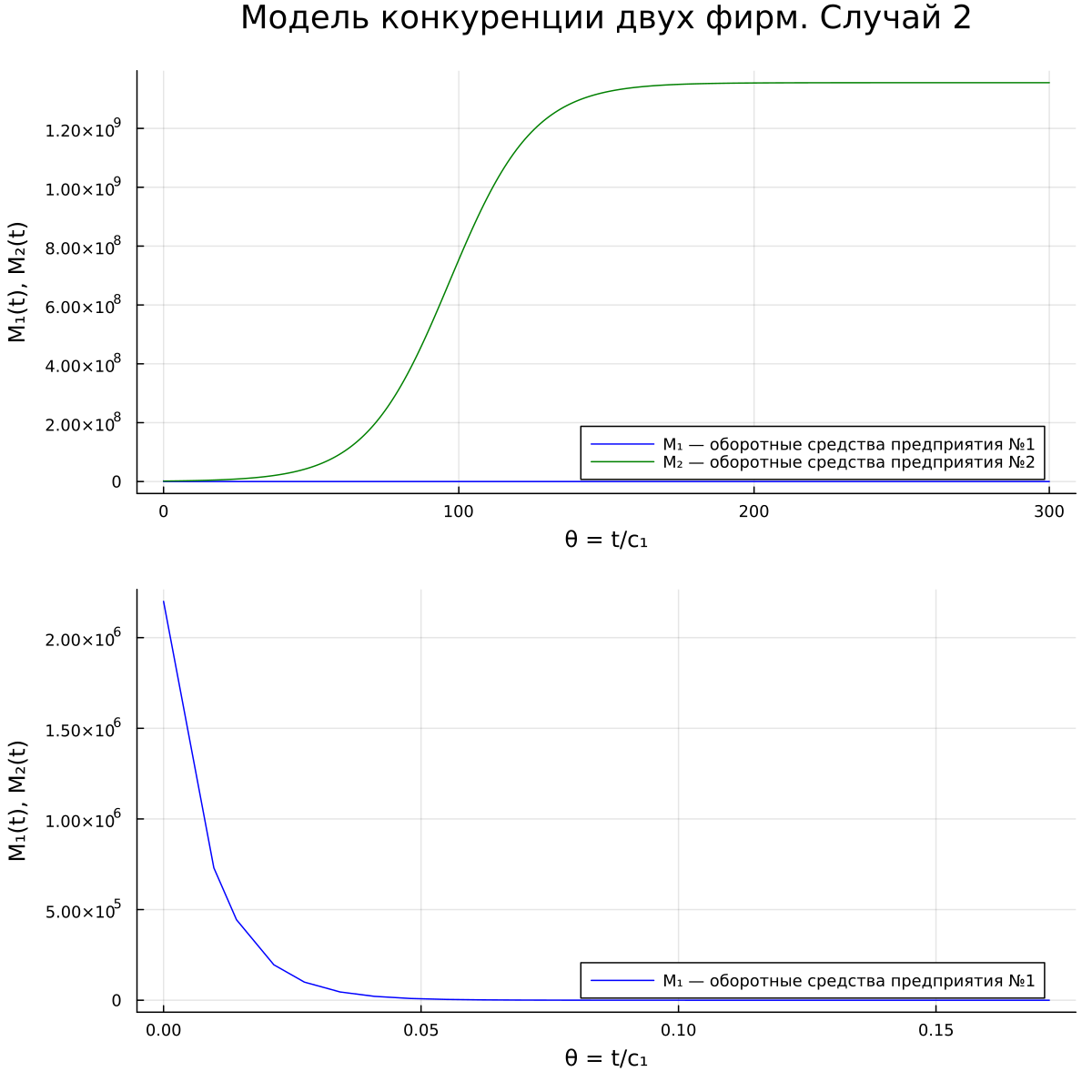
* 
* Figure 11: Результат в виде графика

### 4.2.2 Задание №2

1. Изменяем период времени T, первое уравнение в функции F!, максимальный размер шага при дифференцировании по времени dtmax. Остальные блоки кода оставлены без изменений. Любуемся результатом (подробное объяснение давалось в предыдущей главе) (рис. [12](#fig:11), [13](#fig:002)).

* "Начальные условия: u₀[1] - M¹₀, u₀[2] - M²₀"  
  u₀ = [M¹₀, M²₀]  
    
  "Период времени"  
  T = (0.0, c₁\*300)  
    
  "Правая часть нашей системы, p не используется. u[1] - M₁, u[2] - M₂"  
  function F!(du, u, p, t)  
   # t = t / c₁  
   du[1] = u[1] - ((b/c₁) + 0.0014) \* u[1] \* u[2] - (a₁/c₁) \* u[1]^2  
   du[2] = (c₂/c₁) \* u[2] - (b/c₁) \* u[1] \* u[2] - (a₂/c₁) \* u[2]^2  
  end  
    
    
  prob = ODEProblem(F!, u₀, T)  
  sol = solve(prob, dtmax=c₁)

* 
* Figure 12: Измененная часть кода

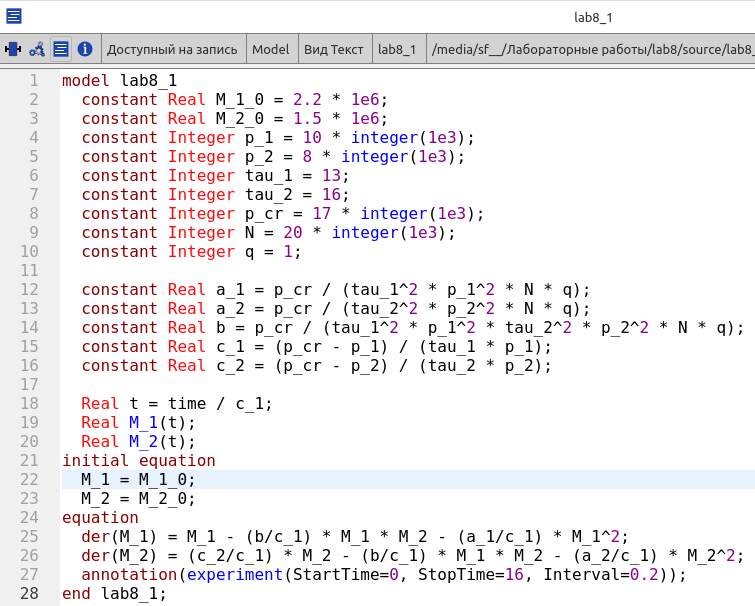
* 
* Figure 13: Результат в виде графиков

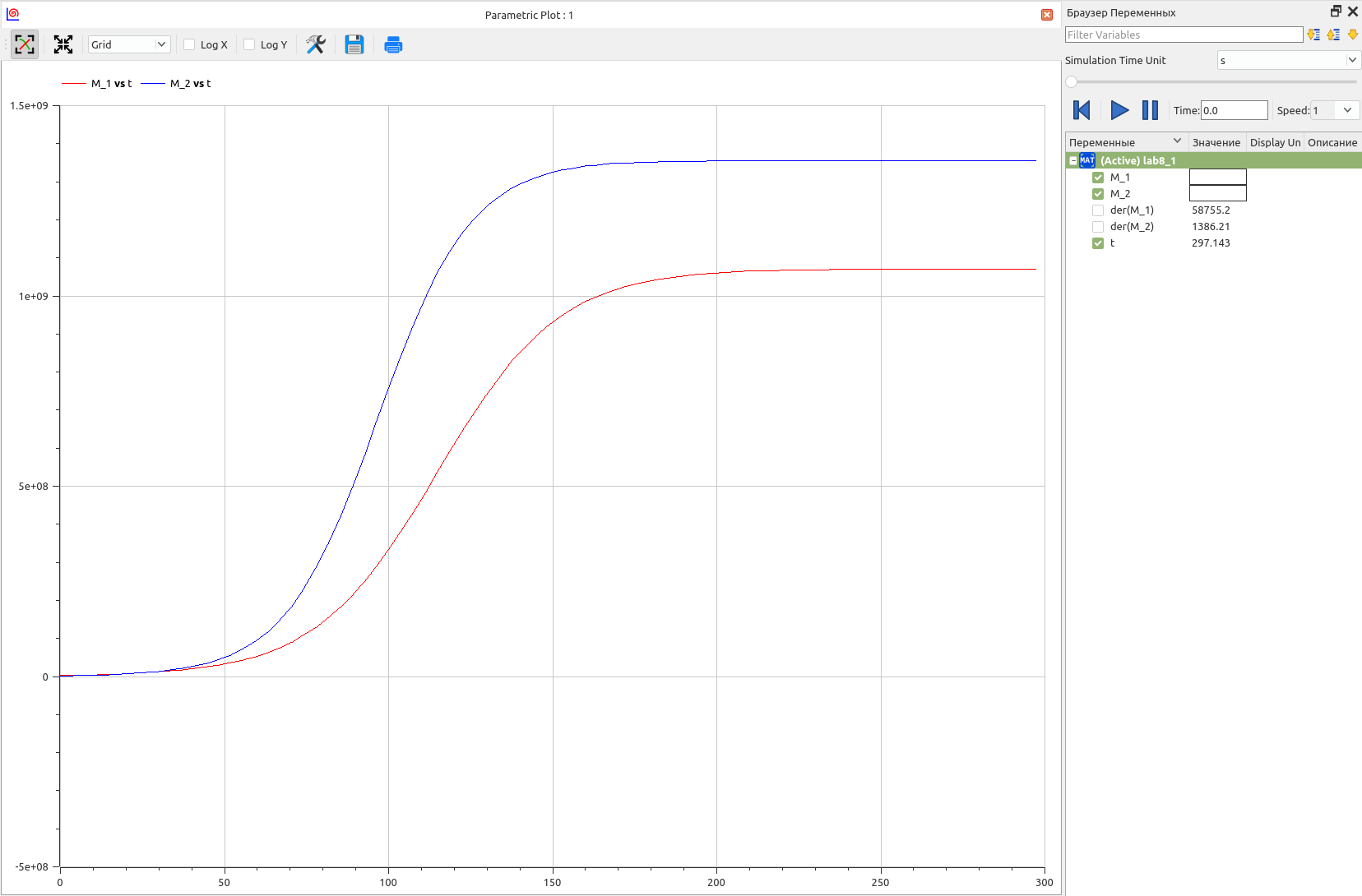
## 4.3 Modelica

### 4.3.1 Задание №1

1. По аналогии с Julia пишем программу, воспроизводящую модель конкуренции двух фирм на языке моделирования Modelica с использованием ПО OpenModelica. Любуемся результатами (рис. [14](#fig:12), [15](#fig:13)).

* model lab8\_1  
   constant Real M\_1\_0 = 2.2 \* 1e6;  
   constant Real M\_2\_0 = 1.5 \* 1e6;  
   constant Integer p\_1 = 10 \* integer(1e3);  
   constant Integer p\_2 = 8 \* integer(1e3);  
   constant Integer tau\_1 = 13;  
   constant Integer tau\_2 = 16;  
   constant Integer p\_cr = 17 \* integer(1e3);  
   constant Integer N = 20 \* integer(1e3);  
   constant Integer q = 1;  
    
   constant Real a\_1 = p\_cr / (tau\_1^2 \* p\_1^2 \* N \* q);  
   constant Real a\_2 = p\_cr / (tau\_2^2 \* p\_2^2 \* N \* q);  
   constant Real b = p\_cr / (tau\_1^2 \* p\_1^2 \* tau\_2^2 \* p\_2^2 \* N \* q);  
   constant Real c\_1 = (p\_cr - p\_1) / (tau\_1 \* p\_1);  
   constant Real c\_2 = (p\_cr - p\_2) / (tau\_2 \* p\_2);  
    
   Real t = time / c\_1;  
   Real M\_1(t);  
   Real M\_2(t);  
  initial equation  
   M\_1 = M\_1\_0;  
   M\_2 = M\_2\_0;  
  equation  
   der(M\_1) = M\_1 - (b/c\_1) \* M\_1 \* M\_2 - (a\_1/c\_1) \* M\_1^2;  
   der(M\_2) = (c\_2/c\_1) \* M\_2 - (b/c\_1) \* M\_1 \* M\_2 - (a\_2/c\_1) \* M\_2^2;  
   annotation(experiment(StartTime=0, StopTime=16, Interval=0.2));  
  end lab8\_1;

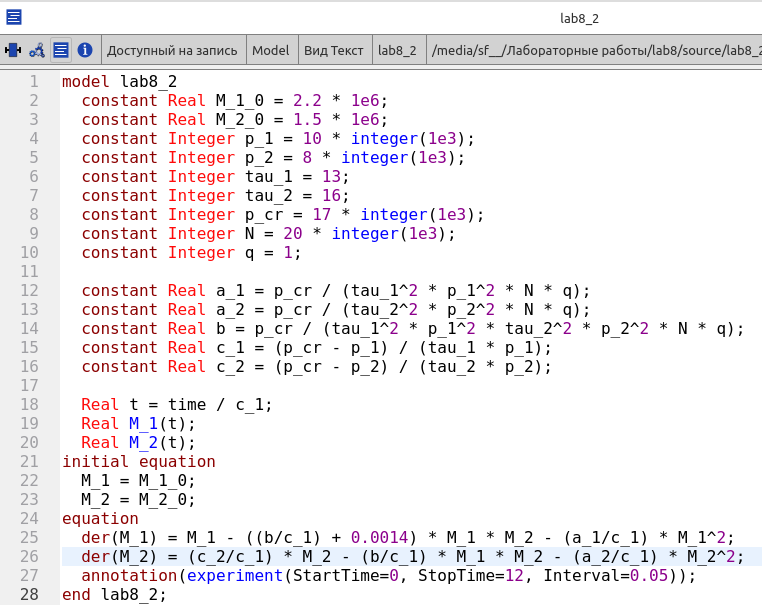
* 
* Figure 14: Определяем коэффициенты, функции M\_1 и M\_2 от времени, нормировку времени t, систему ОДУ, а также начальное/конечное время и частоту разбиения при симуляции

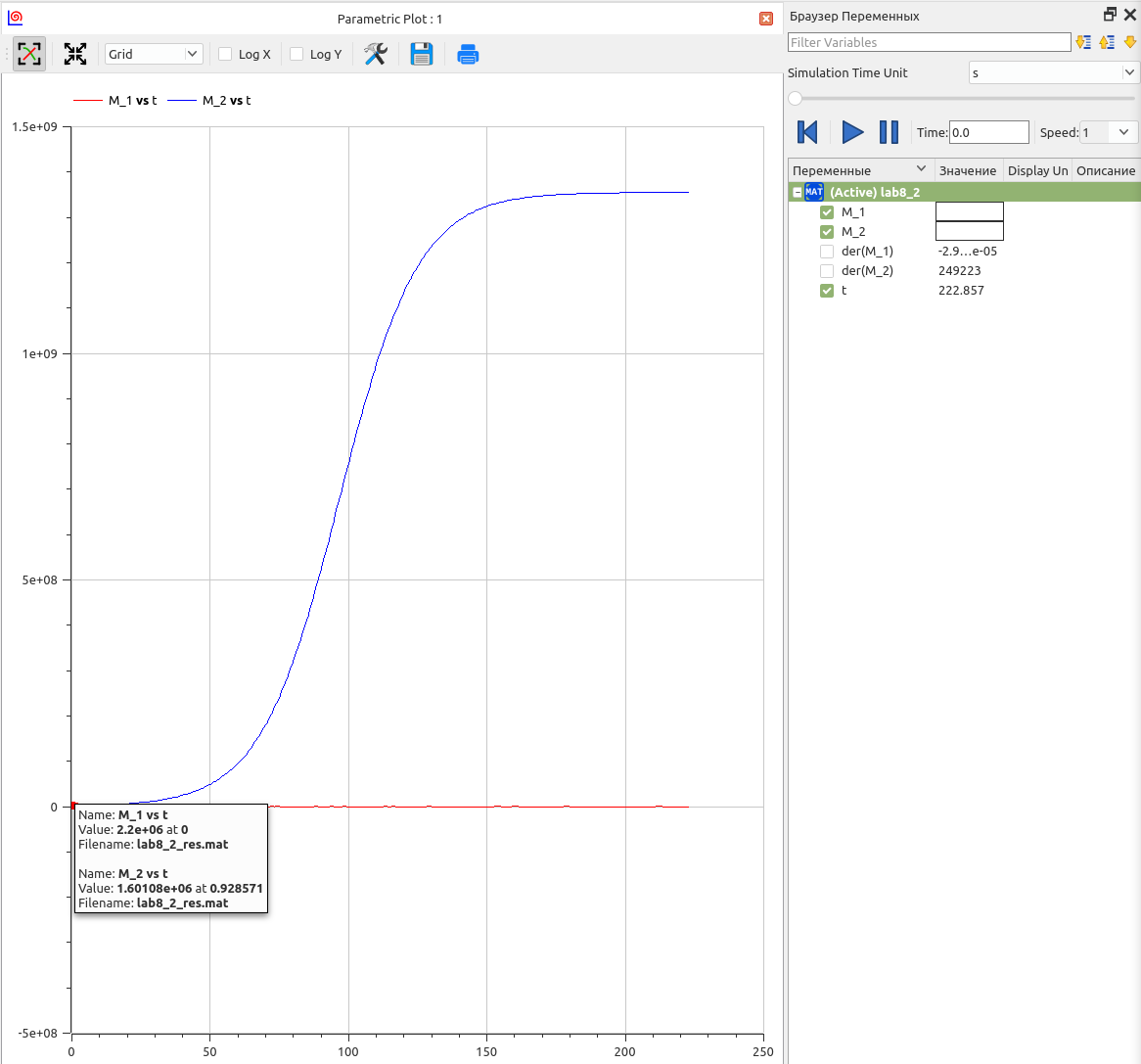
* 
* Figure 15: Результат в виде графика зависимости , от

### 4.3.2 Задание №2

1. По аналогии с Julia пишем программу для второго случая. Любуемся результатами (рис. [16](#fig:14), [17](#fig:15)).

* model lab8\_2  
   constant Real M\_1\_0 = 2.2 \* 1e6;  
   constant Real M\_2\_0 = 1.5 \* 1e6;  
   constant Integer p\_1 = 10 \* integer(1e3);  
   constant Integer p\_2 = 8 \* integer(1e3);  
   constant Integer tau\_1 = 13;  
   constant Integer tau\_2 = 16;  
   constant Integer p\_cr = 17 \* integer(1e3);  
   constant Integer N = 20 \* integer(1e3);  
   constant Integer q = 1;  
    
   constant Real a\_1 = p\_cr / (tau\_1^2 \* p\_1^2 \* N \* q);  
   constant Real a\_2 = p\_cr / (tau\_2^2 \* p\_2^2 \* N \* q);  
   constant Real b = p\_cr / (tau\_1^2 \* p\_1^2 \* tau\_2^2 \* p\_2^2 \* N \* q);  
   constant Real c\_1 = (p\_cr - p\_1) / (tau\_1 \* p\_1);  
   constant Real c\_2 = (p\_cr - p\_2) / (tau\_2 \* p\_2);  
    
   Real t = time / c\_1;  
   Real M\_1(t);  
   Real M\_2(t);  
  initial equation  
   M\_1 = M\_1\_0;  
   M\_2 = M\_2\_0;  
  equation  
   der(M\_1) = M\_1 - ((b/c\_1) + 0.0014) \* M\_1 \* M\_2 - (a\_1/c\_1) \* M\_1^2;  
   der(M\_2) = (c\_2/c\_1) \* M\_2 - (b/c\_1) \* M\_1 \* M\_2 - (a\_2/c\_1) \* M\_2^2;  
   annotation(experiment(StartTime=0, StopTime=12, Interval=0.05));  
  end lab8\_2;

* 
* Figure 16: По сравнению с предыдущим случаем измянются первое уравнение системы, период времени и частота разбиения

* 
* Figure 17: Результат в виде графика зависимости , от

# 5 Анализ результатов

На текущем примере построения математической модель конкуренции двух фирм мы можем продолжить сравнивать язык программирования Julia и язык моделирования Modelica. Хотелось бы ещё раз подчеркнуть, что в Modelica в разы удобнее составлять уравнения, т.к. все переменные, зависящие от времени, подписываются заданными ранее символами в отличие от Julia, где каждой переменной соответствует элемент массива. Такая реализация может запутать, что может привести к ошибкам, связанным с усидчивостью, при описании модели. При выполнении данной лабораторной работы данный недостаток проявил себя, значительно усложнив отладку кода.

В связи с тем, что данная лабораторная работа является последней в рамках курса, было бы разумно подвести определенные итоги в сравнении языков Julia и Modelica.

Если быть откровенным, язык Julia мне понравился больше, нежели Modelica. В первую очередь это связано с тем, что Julia является языком программирования, и процесс написания программ на нем в разы более понятен и эффективен для студента информационного направления. Также алгоритмическая «гибкость» позволяет реализовывать более сложные и информативные структуры (к примеру, анимированные графики), четко отражающие все важные аспекты математической модели.

С другой стороны, язык моделирования Modelica в большинстве случаев позволяет не тратить длительное время на разработку программы, и почти сразу же после начала выполнения лабораторной работы получить приемлемый результат. Меньшая длина кода и большая его читабельность позволяет в разы проще реализовывать математические модели с помощью данного языка, при этом в определенных моментах теряя дополнительную информативность в результате симуляции модели.

# 6 Выводы

Продолжил знакомство с функционалом языка программирования Julia, дополнительных библиотек (DifferentialEquations, Plots), интерактивного блокнота Pluto, а также интерактивной командной строкой REPL. Продолжил ознакомление с языком моделирования Modelica и программным обеспечением OpenModelica. Используя эти средства, описал математическую модель конкуренции двух фирм.

# Список литературы

1. Dynamic model of firms competitive interaction on the market with taxation [Электронный ресурс]. St.Petersburg State University. URL: <https://arxiv.org/pdf/1905.06364>.

2. МОДЕЛЬ КОНКУРЕНЦИИ [Электронный ресурс]. ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ имени М.В. Келдыша Российской академии наук, 2006. URL: <https://www.mathnet.ru/links/a64b164676d285c84118a5a0e280837f/ipmp612.pdf>.

3. Модель конкуренции двух фирм [Электронный ресурс]. RUDN, 2023. URL: <https://esystem.rudn.ru/mod/resource/view.php?id=967253>.