

Project Report

Polytech Nice

Simple Road Traffic Modeling

11th November 2023

Gerbaud Florent
Fatima Rharrou

9-th october to 20-th December september 2023

Academic tutor : Didier Auroux



Table of Contents

1	Presentation of the Subject	4
1.1	Useful Definition	4
1.2	Simple Road Traffic Modeling	4
1.2.1	The objective of SMRT	4
1.2.2	Use case to explain the interest of these simulations	5
1.2.3	Field of application	5
1.2.4	Using GitHub for Project Management	5
2	The equation for SMRT	6
2.1	Differential Ordinary Equation	6
2.1.1	Linear Model	6
2.1.2	Newell's Model	10
2.2	Partial Differential Equations	10
3	Project Objectives	10
3.1	Problems Encountered	10
4	Types of Simulations Performed	10
4.1	Simulation with Drunk Drivers	10
4.2	Simulation with Unpredictable Drivers	10
4.3	Simulation with Drivers Reacting Similarly	10
4.3.1	Accordion Phenomenon	10
4.3.2	Accident Phenomenon	11
4.4	Study of Equilibrium, Stability, and Instability of the Solution	12
4.4.1	System Stability and Equilibrium for the Linear Model	12
4.4.2	System Stability and Equilibrium for the Newell's Model	15
5	Summary	17
6	Annexe	17
6.1	Calculation of the Analytical solutions	17
6.1.1	Linear Model for Two Cars	17
6.1.2	Linear Model for Three Cars	17

List of Figures

1	<u>Road traffic :</u> In this picture, you can see an example of a road traffic phenomenon that we could study	4
2	<u>Road traffic :</u> In this picture, you can see an example of a road traffic phenomenon that we could study	5
3	<u>Simulation of accordion phenomenon :</u> In this picture, you can observe the position evolutions of two cars, the first one (blue) maintaining a constant speed, and the second one (orange) following behind the first one. We could see that orange graph behaves like a periodic accordion with respect to time	11
4	<u>Simulation of Accident case between two cars :</u> In this picture, you can observe the position evolutions of two cars, the first one (blue) maintaining a constant speed, and the second one (orange) following behind the first one. We could see on The graph shows that the acceleration of the orange car is too large. This implies an accident between two cars	12
5	View of the old window of temrec3D	13
6	View of the new window of temrec3D	13
7	View of the old window of temrec3D	14
8	View of the new window of temrec3D	14
9	<u>Road traffic :</u> In this picture, you can see an example of a road traffic phenomenon that we could study	15

List of Algorithms

1	\dot{x}_1	7
2	\dot{x}_i	7
3	sinusoidal_model	8
4	Update Positions and Velocities	9

1 Presentation of the Subject

1.1 Useful Definition

Ordinary Differential Equation (ODE):

An ODE is a mathematical equation that relates a function to its derivatives with respect to one or more independent variables. ODEs are commonly represented given a function F of x , y , and derivatives of y . Then, an equation of the form

$$F(x, y, y', \dots, y^{(n-1)}) = y^{(n)}$$

What is Road Traffic Modeling

The Road Traffic modeling is the study of how vehicles behave on road networks, aiming to simulate and analyze various aspects of traffic flow, congestion, and driver behavior. This field involves creating mathematical and computer models to understand and predict traffic patterns, particularly in scenarios such as congestion, erratic driving, and other relevant factors affecting road transportation. Road traffic modeling plays a crucial role in urban planning, traffic management, and the development of intelligent transportation systems (you could see an exemple on the 1).

1.2 Simple Road Traffic Modeling

Here explain what is the objective how we are going to do the study and so on [1]



Figure 1: **Road traffic :** In this picture, you can see an example of a road traffic phenomenon that we could study

1.2.1 The objective of SMRT

The primary objective of simple road simulation is to analyze and understand the behavior of vehicles in a controlled and reproducible environment. By simulating basic road conditions, re-

searchers and engineers can study the impact of various factors on traffic flow, safety, and efficiency. This helps in the development and testing of traffic management strategies and vehicle control systems. Simple road simulations provide a valuable tool for investigating the dynamic interactions between vehicles and the road environment, contributing to advancements in transportation research and technology.

1.2.2 Use case to explain the interest of these simulations

Simple road simulations find applications in various fields, such as urban planning, transportation engineering, and autonomous vehicle development. For instance, they can be used to assess the effectiveness of new traffic signal timings, study the implications of road design changes, or test the performance of autonomous vehicles in different traffic scenarios. These simulations provide a cost-effective and risk-free way to evaluate potential real-world interventions and innovations. The ability to model and analyze diverse road scenarios enhances decision-making processes and supports the development of sustainable and efficient transportation systems.

1.2.3 Field of application

Simple road simulation is applied across a wide range of fields, including but not limited to transportation research, traffic engineering, and urban planning. Additionally, it plays a crucial role in the development and validation of autonomous vehicle technologies. The ability to model and simulate various road conditions helps researchers and practitioners make informed decisions and improvements in the design and management of transportation systems. Simple road simulation serves as a powerful tool for enhancing our understanding of traffic dynamics and contributes to the ongoing evolution of smart and adaptive transportation systems.

1.2.4 Using GitHub for Project Management

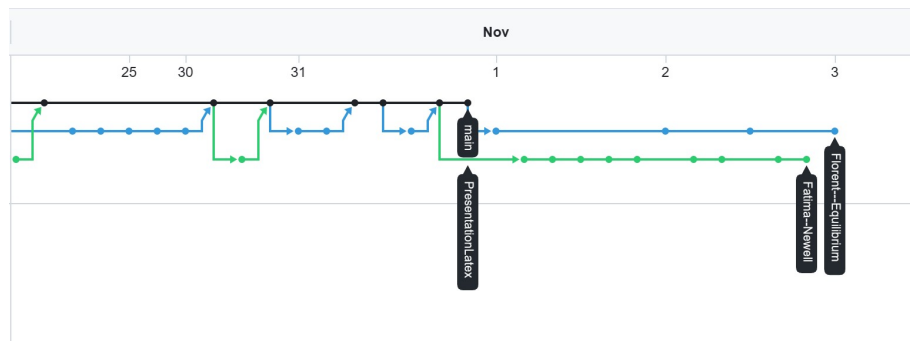


Figure 2: **Road traffic :** In this picture, you can see an example of a road traffic phenomenon that we could study

2 The equation for SMRT

2.1 Differential Ordinary Equation

In this part, the idea is to resolve two types of systems of Ordinary Differential Equations (ODEs) that allow us to simulate traffic flow. To achieve this, we will use the Euler Explicit method to numerically solve the solutions. The Euler Explicit method is given by the following equation :

- EDO to solve: $y'(t) = f(t, y(t))$.
- First step of the resolution: $y_0 = y(t_0)$.
- Recursive process to find the n-th solution of the EDO: $y_{n+1} = y_n + hf(t_n, y_n)$

2.1.1 Linear Model

Mathematical Theory:

In this section, we're going to explain the math behind our model for understanding how cars behave in traffic. To make things simple, we use a discrete model, which means we look at cars one at a time and how they interact on the road.

Each car's movement is governed by a basic equation:

$$\dot{x}_i(t) = V_i = \alpha_i(x_{i-1} - x_i)$$

In this equation, $x_i(t)$ represents where the i -th car is at a given time, V_i is how fast the i -th car is going, and α_i is a number that describes how that car behaves. The right side of the equation, $\alpha_i(x_{i-1} - x_i)$, tells us how the car's speed changes based on how close it is to the car in front.

When we put this equation to work for all the cars, we end up with a bunch of equations (one for each car), which helps us understand how they all move together in traffic. These equations give us a dynamic view of how cars influence each other as they drive.

The system of equations is written like this for each car, where i can be 1, 2, and so on, up to the number of cars:

$$\begin{cases} \dot{x}_1 &= V_1 \\ \dot{x}_2(t) &= \alpha_2(x_1 - x_2) \\ &\vdots \\ \dot{x}_n(t) &= \alpha_n(x_{n-1} - x_n) \end{cases}$$

These equations help us understand how traffic flows and how individual cars influence one another on the road.

In the first part of the study, we consider α_i as a constant. However, in the next part of the simulation, we define some functions $\alpha_i(t)$.

In fact, we have three types of simulations:

1. **Constant value:**

$$\alpha_i(t) = C_i$$

2. **Sinusoidal with noise:**

$$\alpha_i(t) = |W \cdot \sin(\omega t + \phi) + \mathcal{N}(0, 0.1)|$$

3. **Stochastic Driver Model:**

Consider a random driver model with an output labeled as $\alpha_i(t)$, which relies on various factors like the average, spread, and time t . This model produces a random noise part from a usual distribution with an average and spread of $\sqrt{\text{spread}}$. If a limit is given, the generated noise is confined within the interval $[-\text{limit}, \text{limit}]$.

Mathematically, we can express this as:

$$\alpha_i(t) = \begin{cases} |\mathcal{N}(\text{average}, \sqrt{\text{spread}})|, & \text{if } -\text{limit} \leq \alpha_i(t) \leq \text{limit}, \\ -\text{limit}, & \text{if } \alpha_i(t) < -\text{limit}, \\ \text{limit}, & \text{if } \alpha_i(t) > \text{limit}. \end{cases}$$

Here, average signifies the mean of the acceleration, and spread stands for the acceleration's spread. The limit is defined to prevent or create specific situations, like accidents, depending on the desired result.

Implementation :

Algorithm 1 \dot{x}_1

Output:

- $x_1 = V_1$

Algorithm:

- $x_1 = 130 \times \frac{1000}{3600}$
 - **return** \dot{x}_1
-

Algorithm 2 \dot{x}_i

Input:

- $t := \text{"time step"}$
- $x_i := \text{Table of position for the } i\text{-th Car}$
- $x_{i-1} := \text{Table of position for the } (i-1)\text{-th Car}$

Output:

- $\dot{x}_i = V_i$

Algorithm:

- $\dot{x}_i(t) = V_i = \alpha_i(x_{i-1}[t] - x_i[t])$
 - **return** \dot{x}_i
-

This algorithm allows to define an orthonormed local reference frame and we delete the lines to eliminate for reducing matrix

Algorithm 3 sinusoidal_model

Input:

- W := "Amplitude of the perturbation"
- ω := "Angular frequency"
- t := "time"
- ϕ := "Phase (in radians)"

Output:

- This function returns a value of acceleration ($\alpha_i(t)$) following a sinusoidal model with random noise.

Algorithm:

- $\alpha_i(t) = |W \cdot \sin(\omega \cdot t + \phi) + N|$
 - N := "random noise such as $N \sim \mathcal{N}(0, 0.01)$ "
 - **return** $\alpha_i(t)$
-

Algorithm 4 Update Positions and Velocities

Input:

- $t :=$ time step
- $x_1Pos :=$ Table of position for the 1st Car
- $x_2Pos :=$ Table of position for the 2nd Car
- $v_1 :=$ Table of velocity for the 1st Car
- $v_2 :=$ Table of velocity for the 2nd Car
- $time :=$ Table of time values
- $h :=$ time step size

Output:

- x_1Pos
-
- x_2Pos
- $accident :=$ Flag indicating whether an accident occurred

Algorithm:

- **for** t in range(1, n_steps) **do**
 - $x_1Pos[t] = x_1Pos[t - 1] + \dot{x}_1() \times h$
 - $x_2Pos[t] = x_2Pos[t - 1] + \dot{x}_2(t - 1) \times h$
 - $v_1[t] = \dot{x}_1()$
 - $v_2[t] = \dot{x}_2(t)$
 - $time[t] = time[t - 1] + h$
 - **if** isAccident(t) **then**
 - $accident = \text{True}$
 - **break**
 - **end if**
 - **end for**
 - **Return** x_1Pos, x_2Pos
-

2.1.2 Newell's Model

2.2 Partial Differential Equations

3 Project Objectives

3.1 Problems Encountered

4 Types of Simulations Performed

4.1 Simulation with Drunk Drivers

4.2 Simulation with Unpredictable Drivers

4.3 Simulation with Drivers Reacting Similarly

4.3.1 Accordion Phenomenon

The initial simulation we aimed to conduct was intended to demonstrate and illustrate the behavior of drivers in real-life scenarios. Figure 3 accurately depicts this phenomenon. Specifically, we make the following assumptions:

- The first car maintains a constant speed over time.
- The second car reaction is based on the behavior of the first one.
- The acceleration of the second car is governed by a constant function, denoted as alpha, which remains fixed. In the algorithm 2 we took $\alpha_2 = 2.0$

Similar to real-life situations, we observe that if the second car is too far behind the first one, it decelerates. Conversely, when it is adequately distant from the first car, it increases its speed.

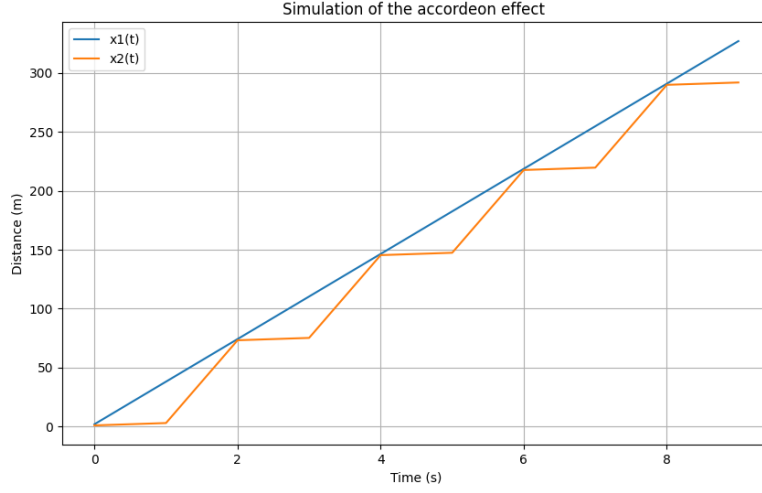


Figure 3: **Simulation of accordion phenomenon :** In this picture, you can observe the position evolutions of two cars, the first one (blue) maintaining a constant speed, and the second one (orange) following behind the first one. We could see that orange graph behaves like a periodic accordion with respect to time

4.3.2 Accident Phenomenon

For this case, the objectif is to show that if one driver have a bigger time reaction capacity, there is an accident. Onto this case, we took the same assumptions than before, but with a capacity of accelleration a little bit lower and a higher time reaction. In the algorithm 2 we took $\alpha_2 = 1.75$ and in the algorithm 4 we took $h = 1.5$

On the figure 4 accident is confirmed because there is a curve intersection (the position of the car number two (orange) is greater than the first car (blue)) So it means that the second car is crashed into the first car.

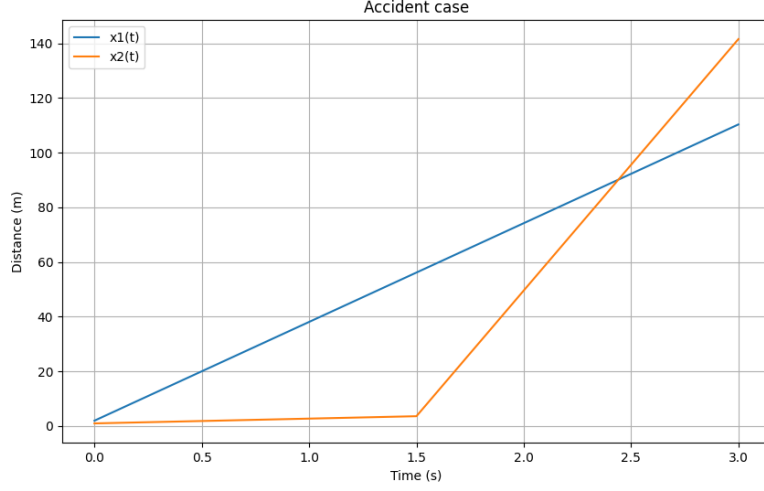


Figure 4: **Simulation of Accident case between two cars :** In this picture, you can observe the position evolutions of two cars, the first one (blue) maintaining a constant speed, and the second one (orange) following behind the first one. We could see on The graph shows that the acceleration of the orange car is too large. This implies an accident between two cars

4.4 Study of Equilibrium, Stability, and Instability of the Solution

4.4.1 System Stability and Equilibrium for the Linear Model

In this part, the idea is to study the distance difference between two cars in order to determine in which case the solutions is stable and if there is an equilibrium for the solution. For doing that, we studied the solutions of the following system of equation :

$$\begin{cases} \dot{d}_1 &= V_1 - \alpha_2 \cdot d_1 \\ &\vdots \\ \dot{d}_n &= \alpha_n \cdot d_{n-1} - \alpha_{n+1} \cdot d_n \end{cases}$$

So, In our context, for the two cars, we have a 1D system. With 3 cars, we have a 2D system of equations.

Now talk about the system stability and equilibrium :

With 2 cars :

So, the equation to study is the following one :

$$\dot{d}_1 = V_1 - \alpha_2 \cdot d_1$$

After the resolution (6.1.1) we obtain the next equation for $d_1(t)$:

$$d_1(t) = \frac{V_1 - V_1 \cdot e^{-\alpha_2(t-t_0)} + \alpha_2 \cdot d_1(t_0) \cdot e^{-\alpha_2(t-t_0)}}{\alpha_2}$$

The first intuition before seeing the equation was to think that when the $\alpha_2 < 0$, the solution is divergent because one car is go back and the other go forward. Now, if we study the Equation, the terms V_1 and $\alpha_2 \cdot d_1(t_0)$ are constant so they are not interesting.

However, it's important to note that both terms, $e^{-\alpha_2(t-t_0)}$, are exponential functions of time. If the value of α_2 is negative, these exponentials will increase exponentially over time.

So we could easily see that the equation is stable and converge onto the equilibrium $d_1(t) = \frac{V_1}{\alpha_2}$. graphically we can see it by :

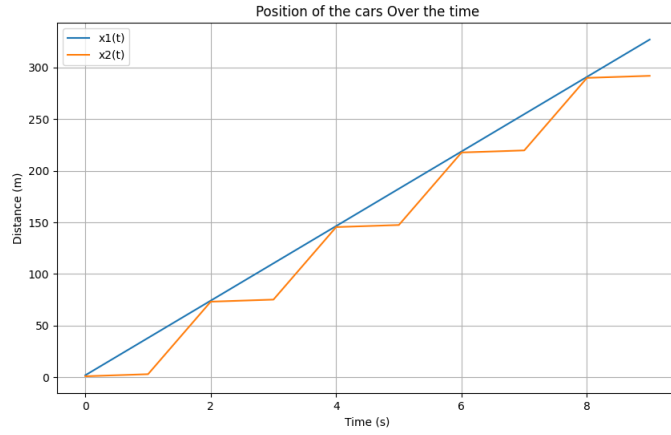


Figure 5: View of the old window of temrec3D

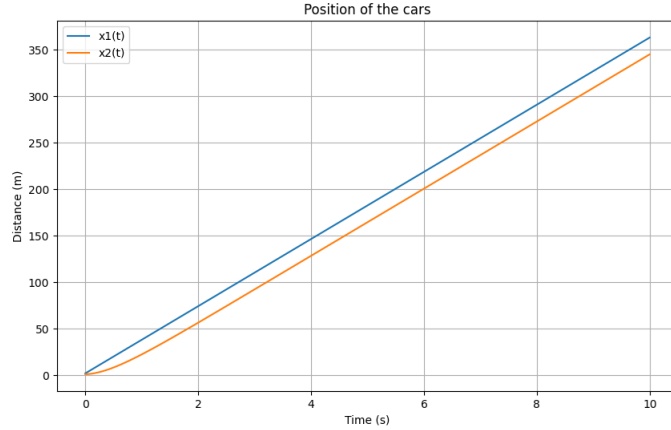


Figure 6: View of the new window of temrec3D

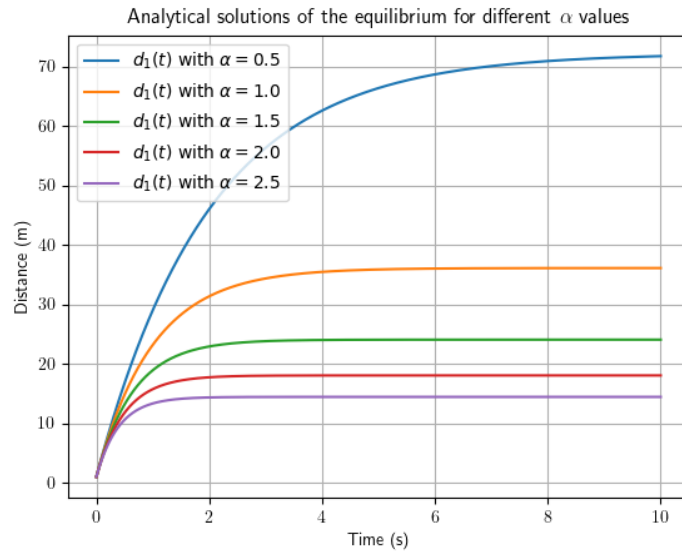


Figure 7: View of the old window of temrec3D

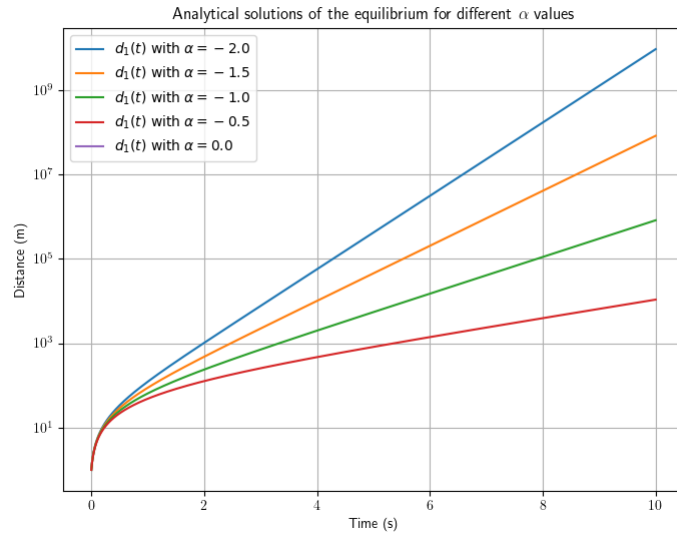


Figure 8: View of the new window of temrec3D

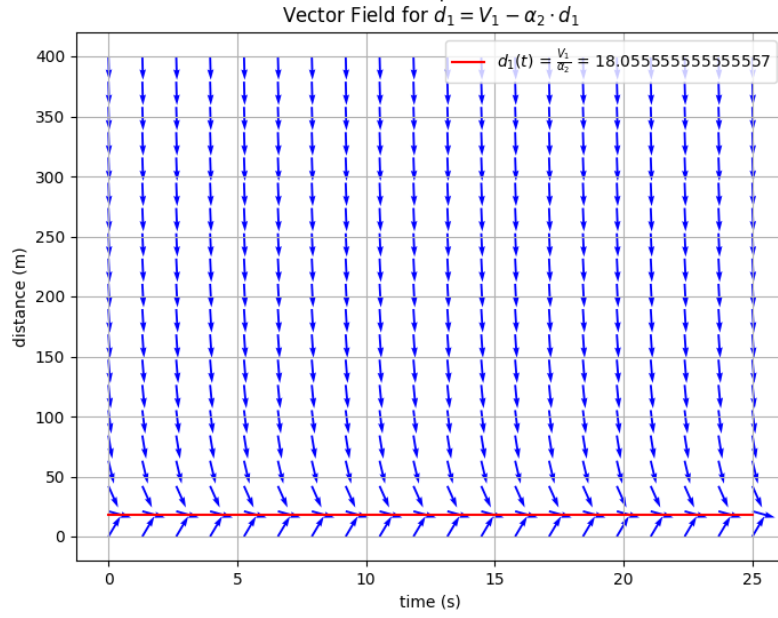


Figure 9: **Road traffic :** In this picture, you can see an example of a road traffic phenomenon that we could study

With 3 cars :

After the resolution (6.1.2) we obtain the next equation for $d_1(t)$ and $d_2(t)$:

4.4.2 System Stability and Equilibrium for the Newell's Model

The Study of the equilibrium for two cars :

The objective here, as before, is to study the stability of the model. The 1D equation is as follows:

$$\dot{d}_1(t) = \dot{x}_1(t) - \dot{x}_2(t) = V_1 - V_2 + V_2 \cdot e^{-\frac{\alpha_2}{V_2}(d_1 - d_2^{sec})} = f(d_1, t),$$

$$\begin{cases} V_1 := \text{Maximum speed for car number 1,} \\ V_2 := \text{Maximum speed for car number 2,} \\ \alpha_2 := \text{capacity of acceleration for car number 2,} \\ d_2^{sec} := \text{security distance that car number 2 maintains.} \end{cases}$$

Firstly, let's determine the equilibrium:

$$\begin{aligned}
V_1 - V_2 + V_2 \cdot e^{-\frac{\alpha_2}{V_2}(d_1 - d_2^{sec})} &= 0 \\
e^{-\frac{\alpha_2}{V_2}(d_1 - d_2^{sec})} &= \frac{V_2 - V_1}{V_2} \\
e^{-\frac{\alpha_2}{V_2}d_1} &= \frac{V_2 - V_1}{V_2} e^{-\frac{\alpha_2}{V_2}d_2} \\
-\frac{\lambda_2}{V_2}d_1 &= \ln\left(\frac{V_2 - V_1}{V_2} e^{-\frac{\alpha_2}{V_2}d_2}\right) \\
d_1^* &= -\frac{V_2}{\lambda_2} \ln\left(\frac{V_2 - V_1}{V_2} e^{-\frac{\alpha_2}{V_2}d_2}\right)
\end{aligned}$$

The challenge with this equation is the inability to find an analytical solution. Consequently, we apply Lyapunov's Indirect Theorem to investigate the equilibrium.

In the context of 1D analysis, we begin by calculating $f'(d_1, t)$, which results in:

$$f'(d_1, t) = -\alpha_2 e^{-\frac{\alpha_2}{V_2}(d_1 - d_2^{sec})}$$

It is evident that $f'(d_1, t)$ is negative for all values of α_2 greater than zero. Therefore, in our specific scenario, the equilibrium remains stable at all times, as negative acceleration is not considered.

From the equilibrium, we also observe that the equilibrium exists if and only if $V_2 > V_1$.

Clearly, the condition for a stable equilibrium is $\alpha_2 > 0$ and $V_2 > V_1$.

The Study of the equilibrium for three cars :

$$\begin{cases} \dot{d}_1(t) = x_1(t) - x_2(t) = V_1 - V_2 + V_2 \cdot e^{-\frac{\alpha_2}{V_2}(d_1 - d_2^{sec})} = f(d_1, t), \\ \dot{d}_2(t) = x_2(t) - x_3(t) = V_2 - V_2 \cdot e^{-\frac{\alpha_2}{V_2}(d_1 - d_2^{sec})} - V_3 + V_3 \cdot e^{-\frac{\alpha_3}{V_3}(d_2 - d_3^{sec})} \end{cases}$$

$$\begin{cases} V_1 := \text{Maximum speed for car number 1,} \\ V_2 := \text{Maximum speed for car number 2,} \\ \alpha_2 := \text{capacity of acceleration for car number 2,} \\ d_2^{sec} := \text{security distance that car number 2 maintains.} \end{cases}$$

again, we apply Lyapunov's Indirect Theorem to investigate the equilibrium.

We are going to calculate the Jacobienne matrix :

$$J_{\bar{x}} = \begin{pmatrix} -\alpha_2 e^{-\frac{\alpha_2}{V_2}(d_1 - d_2^{sec})} & 0 \\ \alpha_2 e^{-\frac{\alpha_2}{V_2}(d_1 - d_2^{sec})} & -\alpha_3 e^{-\frac{\alpha_3}{V_3}(d_2 - d_3^{sec})} \end{pmatrix}$$

However, in the 2D case, Lyapunov tells us that the equilibrium is stable if:

$$\begin{cases} Tr(J_{\bar{x}}) < 0 \\ \det(J_{\bar{x}}) > 0 \end{cases}$$

The both condition are respected iff $\lambda_2 > 0, \lambda_3 > 0$

5 Summary

6 Annexe

6.1 Calculation of the Analytical solutions

6.1.1 Linear Model for Two Cars

$$\dot{d}_1(t) = V_1 - \alpha_2 \cdot d_1$$

We use the variable separation method to Solve this EDO, so we obtain :

$$\begin{aligned} \int_{x(t_0)}^{x(t_f)} \frac{1}{V_1 - \alpha_2 \cdot d_1(t)} d d_1(t) &= \int_{t_0}^{t_f} dt \\ -\frac{1}{\lambda_2} [\ln |V_1 - \alpha_2 \cdot d_1(t)|]_{x(t_0)}^{x(t_f)} &= t_f - t_0 \\ -\frac{1}{\lambda_2} \ln \left| \frac{V_1 - \alpha_2 \cdot d_1(t_f)}{V_1 - \alpha_2 \cdot d_1(t_0)} \right| &= t_f - t_0 \end{aligned}$$

We could remove the $|\cdot|$ because the sign is always the same. And we get :

$$\begin{aligned} \ln \left(\frac{V_1 - \alpha_2 \cdot d_1(t_f)}{V_1 - \alpha_2 \cdot d_1(t_0)} \right) &= -\lambda_2(t_f - t_0) \\ \frac{V_1 - \alpha_2 \cdot d_1(t_f)}{V_1 - \alpha_2 \cdot d_1(t_0)} &= e^{-\lambda_2(t_f - t_0)} \\ V_1 - \alpha_2 \cdot d_1(t_f) &= (V_1 - \alpha_2 \cdot d_1(t_0))e^{-\lambda_2(t_f - t_0)} \\ d_1(t) &= \frac{V_1 - [V_1 - \alpha_2 \cdot d_1(t_0)]e^{-\lambda_2(t_f - t_0)}}{\alpha_2} \end{aligned}$$

6.1.2 Linear Model for Three Cars

The system under matricial form could be write like that :

$$\dot{D} = (t) \begin{pmatrix} -\alpha_2 & 0 \\ \alpha_2 & -\alpha_3 \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} + \begin{pmatrix} V_1 \\ 0 \end{pmatrix}$$

The first step is to perform the eigenvectors of the matrix:

$$\begin{vmatrix} X + \alpha_2 & 0 \\ \alpha_2 & X + \alpha_3 \end{vmatrix} = (X + \alpha_2)(X + \alpha_3) = X^2 + X(\alpha_3 + \alpha_2) + \alpha_2\alpha_3$$

by simple resolution of the second order polynomial, we could fin two eigenvalues:
 $\lambda_1 = -\alpha_2, \lambda_2 = -\alpha_3$ we looking for the two eigen vectors :

$$\left\{ \begin{array}{cc|c} 0 & 0 & 0 \\ \alpha_2 & -\alpha_3 + \alpha_2 & 0 \end{array} \right. \quad \left\{ \begin{array}{cc|c} -\alpha_2 + \alpha_3 & 0 & 0 \\ \alpha_2 & 0 & 0 \end{array} \right.$$

It is easy to see that the two following vector are vector that verify the condition :

$$\left\{ \begin{array}{l} u_1 = \begin{pmatrix} 1 \\ \frac{\alpha_3 - \alpha_2}{\alpha_2} \end{pmatrix} \\ u_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{array} \right.$$

Then, we know that a general solution of the equation is :

$$\bar{D}(t) = \sum_{i=1}^2 X_i$$

where $X_i = C_i e^{\lambda_i t} u_i$,

with C_i being constants to be determined, and u_i as the eigenvectors, **and** λ_i **the eigenvalues**.

So :

$$X_1 = \begin{pmatrix} \frac{C_1(\alpha_3 - \alpha_2)}{\alpha_2 e^{\alpha_2 t}} \\ \frac{C_1}{e^{\alpha_2 t}} \end{pmatrix}$$

$$X_2 = \begin{pmatrix} 0 \\ \frac{C_2}{e^{\alpha_3 \cdot t}} \end{pmatrix}$$

Finally the general solution is given by :

$$\bar{X} = \begin{pmatrix} \frac{C_1(\alpha_3 - \alpha_2)}{\alpha_2 e^{\alpha_2 t}} \\ \frac{C_1}{e^{\alpha_2 t}} + \frac{C_2}{e^{\alpha_3 \cdot t}} \end{pmatrix}$$

Then we are going to find for particular solution :

$$\begin{cases} \dot{d}_1 = 0 \\ \dot{d}_2 = 0 \end{cases} \iff \begin{cases} d_1^* = \frac{V_1}{\alpha_2} \\ d_2^* = \frac{V_1}{\alpha_3} \end{cases}$$

In clear, we have :

$$\begin{cases} \dot{d}_1(t) &= \frac{C_1(\alpha_3 - \alpha_2)}{\alpha_2 e^{\alpha_2 t}} + \frac{V_1}{\alpha_2} \\ \dot{d}_2(t) &= \frac{C_1}{e^{\alpha_2 t}} + \frac{C_2}{e^{\alpha_3 t}} + \frac{V_1}{\alpha_3} \end{cases}$$

we set the initial condition to
 $d_1(0) = m, d_2(0) = n$ So we find easily C_1 and C_2 :

$$\begin{aligned} C_1 &= \frac{m\alpha_2 - V_1}{\alpha_3 - \alpha_2} \\ C_2 &= n - \frac{V_1}{\alpha_3} - C_1 \end{aligned}$$

References

- [1] Mansoureh Jeihani, Shiva NarooieNezhad and Kaveh Bakhsh Kelarestaghi. “Integration of a driving simulator and a traffic simulator case study: Exploring drivers’ behavior in response to variable message signs”. In: *IATSS Research* 41.4 (2017), pp. 164–171. ISSN: 0386-1112. DOI: 10.1016/j.iatssr.2017.03.001. URL: <https://www.sciencedirect.com/science/article/pii/S0386111217300304>.