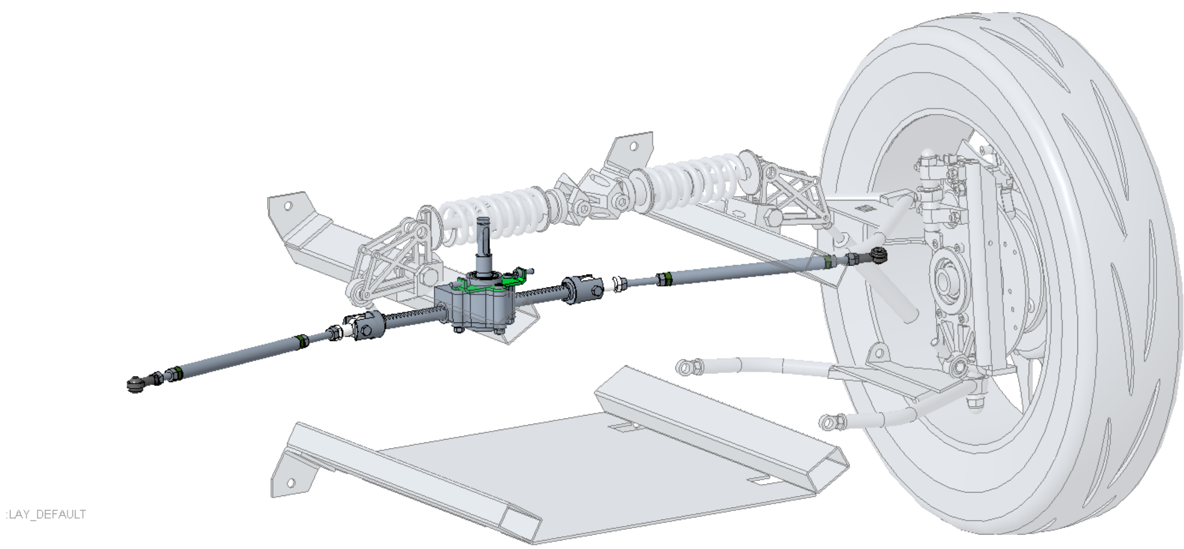
Optimization of steering geometry in passenger cars by means of reinforcement learning

1st milestone



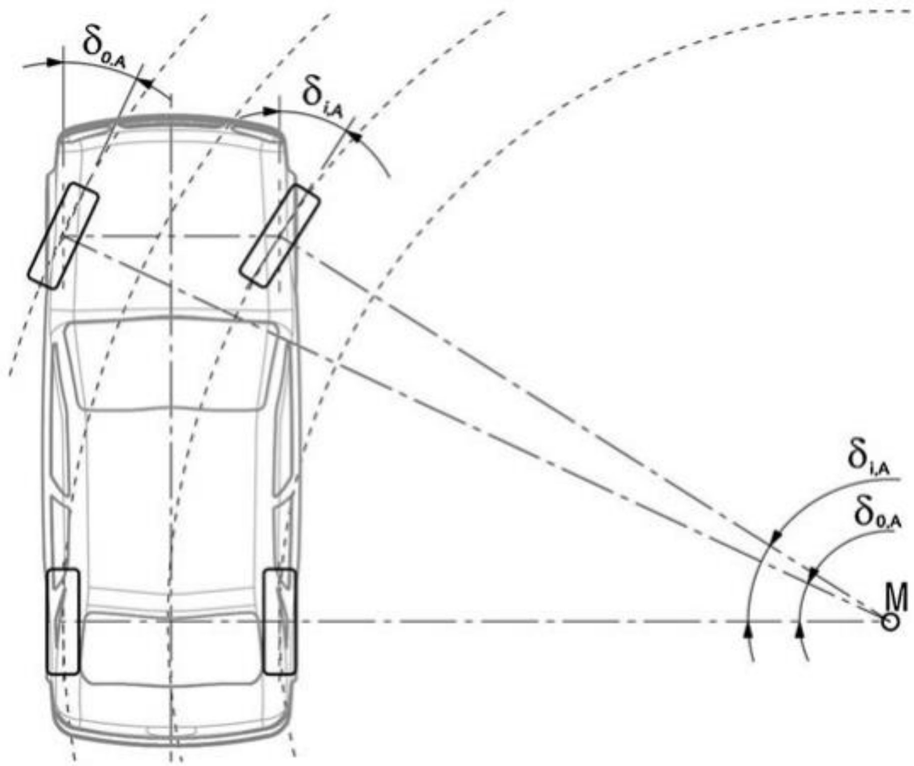
2019.10.15

Table of contents

Introduction

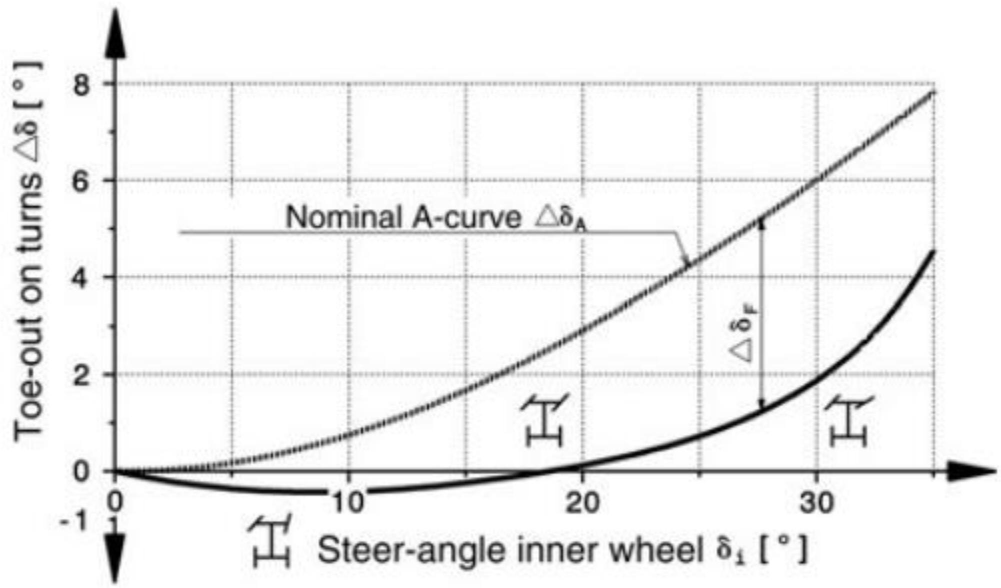
In the past, estimations, intuitions and experience-based solutions were common when the designer faced a complex problem. Nowadays powerful computers can help our work, not only for calculations but visualizing, and helping us to make decisions. Today the demand for the possible best solutions is high, especially in the engineering fields.

An important part of every vehicle is its steering mechanism. In most passenger cars steering is achieved by a set of rigid links and joints. This steering mechanism is responsible for the desired alignment of the wheels for every turning radius set by the steering wheel.



1. Figure - Ideal turning

During slow cornering, when a curve is negotiated slowly, the wheels try to roll towards their center planes without any occurrence of a slip angle. For this to happen, the normals to the center planes of the wheels must intersect at a point, the so-called instant center (Fig. 1, point M). These conditions are met by the so-called Ackermann steering angles at the front wheels ($\delta_{O,A}, \delta_{i,A}$). The Ackermann law states that the front wheel on the inside of the curve has to be steered to a steeper angle than the wheel on the outside. This condition has to be met at least approximately by steering kinematics. However steering mechanisms usually are not able to meet the Ackermann condition. If we calculate the outer and inner Ackermann steering angles for a given steering radius, the difference between the two angles is the A-angle. The deviation form this A-angle is called the artificial steering divergence $\delta_{\Delta F}$. For a typical application the relation between $\delta_{i,A}$ and $\delta_{\Delta F}$ is shown on fig. 2.

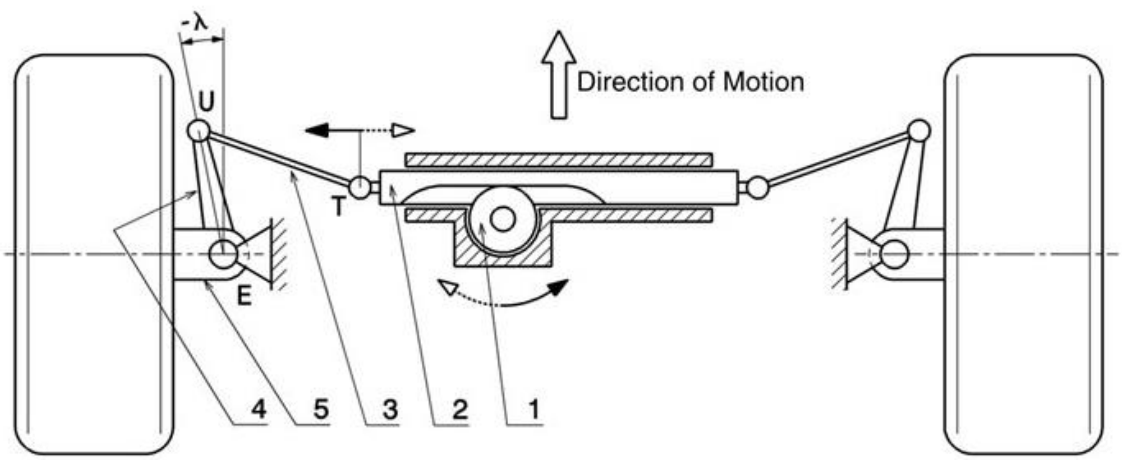


2. Figure - Artificial steering divergence

A steering mechanism is optimal for an exact application, if the two curves on fig. 2 are as close as possible.

The Problem

The definition of the steering geometry happens in an iterative way because it is affected by other factors, for example the drivetrain or chassis geometry.



3. Figure - Rack pinion steering

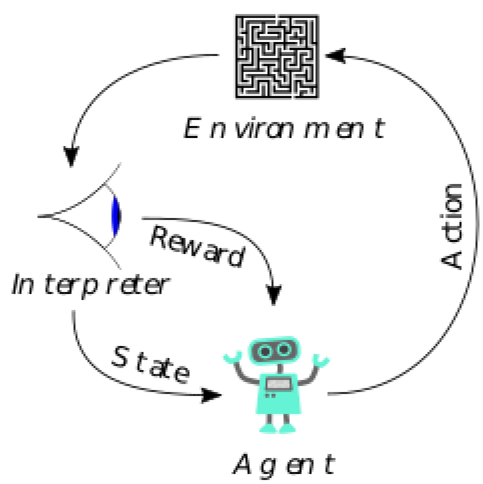
The most common way to steer the wheels in passenger cars is the so-called rack-pinion steering. It consists of the pinion (1) which transmits the rotary motion of the steering wheel, from the rack (2) that converts the pinions rotary motion into linear motion, the tie rods (3), the steering arm (4) and from the wheels (5). By rotating the steering wheel, the rack starts to move to one direction, causing the motion of the tie rods' inner mounting points with it. Because of the fixed length of the tie rods, they pull and push the steering arms of the wheels to the corresponding direction and therefore they rotate the wheels around the kingpin axes (for the left wheel marked with E on fig. 3.)

To determine the geometry of the steering mechanism, estimations and best practices are used along with analytical simplifications. And as the experts of the discipline suggests: *"The designer will 'approach' the exact definition of the linkage by drawings or, more up-to-date, at the computer. This will be successful faster than extensive auxiliary designs considering the 3D nature of steering and chassis"* (Matschinsky 2007).

The Solution

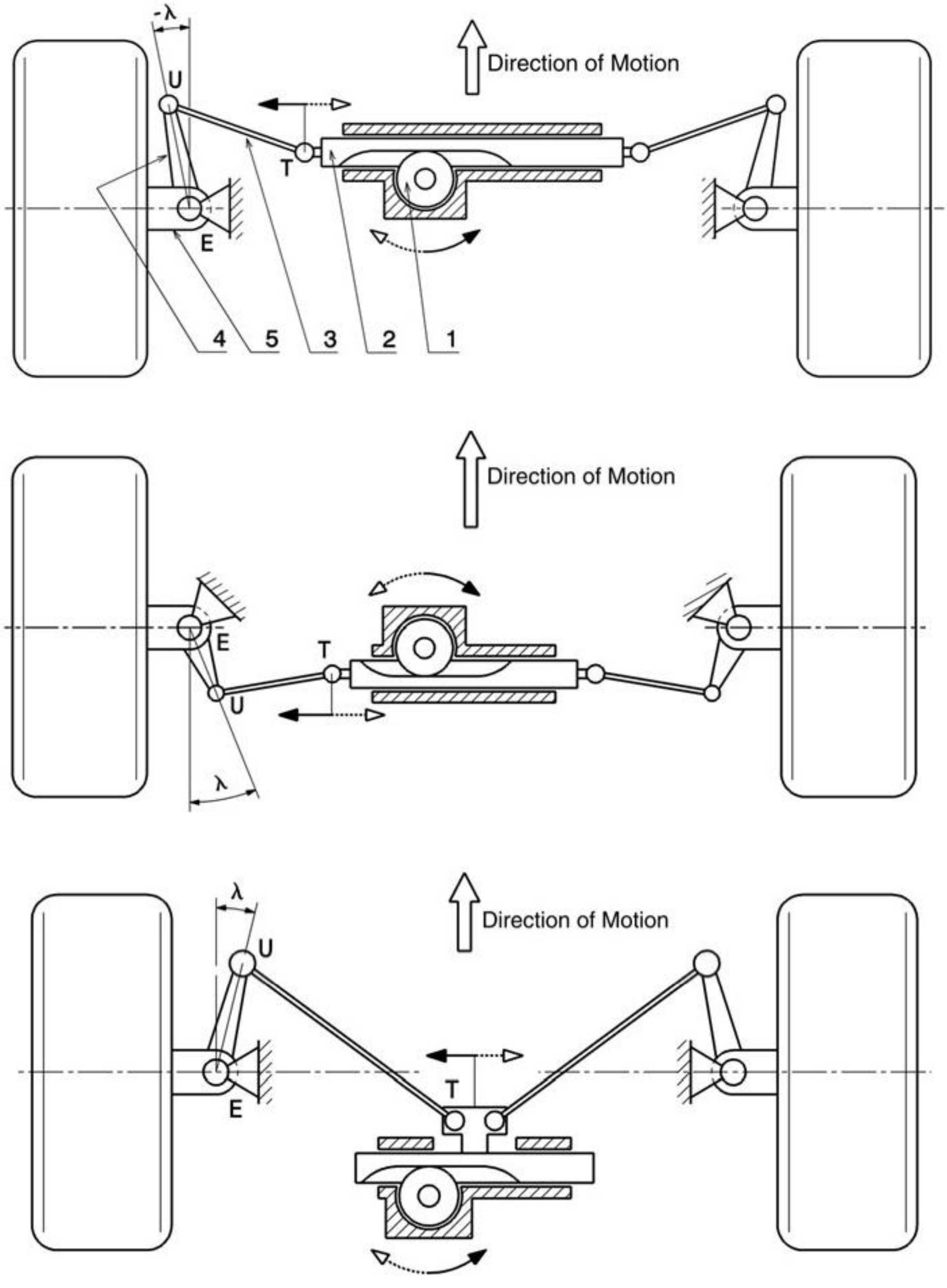
Our goal is to make an optimization algorithm using reinforcement learning, to determine the optimal parameters of a rack-pinion steering geometry such as steering arm length, rack length, rack distance, etc. instead of estimations and best practices.

Knowing that for a given wheelbase and track width several possible solutions exists, we would like to use the agent to discover the solution space and achieve the optimal geometry.



4. Figure - Reinforcement learning process

The environment for the learning process is a basic simulation which can calculate the artificial steering divergence curve, so the reward or penalty could be calculated from the mean squared error of the divergence from the nominal A-curve for example.



5. Figure - Different configuartions of the steering geometry

Reinforcement Learning

Reinforcement learning (RL) is an area of machine learning concerned with how software agents ought to take actions in an environment so as to maximize some notion of cumulative reward. Reinforcement learning is one of three basic machine learning paradigms, alongside supervised learning and unsupervised learning. Reinforcement learning involves an agent, a set of states, and a set of actions per state. By performing an action, the agent transitions from state to state. Executing an action in a specific state provides the agent with a reward. The goal of the agent is to maximize its total reward. It does this by adding the maximum reward attainable from future states to the reward for achieving its current state, effectively influencing the current action by the potential future reward. This potential reward is a weighted sum of the expected values of the rewards of all future steps starting from the current state. There are various methods for reinforcement learning, in our project we will use Q learning algorithm.

Q learning

Q-learning is a model-free reinforcement learning algorithm. The goal of Q-learning is to learn a policy, which tells an agent what action to take under what circumstances. It does not require a model (hence the connotation "model-free") of the environment, and it can handle problems with stochastic transitions and rewards, without requiring adaptations. For any finite Markov decision process (FMDP), Q-learning finds a policy that is optimal in the sense that it maximizes the expected value of the total reward over any and all successive steps, starting from the current state. Q-learning can identify an optimal action-selection policy for any given FMDP, given infinite exploration time and a partly-random policy."Q" names the function that returns the reward used to provide the reinforcement and can be said to stand for the "quality" of an action taken in a given state.

Markov decision process

Some thoughts from the Markov decision process. The Markov decision process (MDP) is a discrete time stochastic control process. It provides a mathematical framework for modeling decision making in situations where outcomes are partly random and partly under the control of a decision maker. MDPs are useful for studying optimization problems solved via dynamic programming and reinforcement learning. MDPs were known at least as early as the 1950s.