SOFTWARE DIFFERENTIAL FORMULAS

Table of Contents

Introduction

During the implementation of my R/C Car I decided to design the formulas for the software differential. The R/C Car will have 4 motors, so the software differential is the only way. Apart from that, I believe that mechanical differentials:

- Cannot be improved further
- Cannot be smaller in size with the stiffness which a car needs
- Cannot be lighter (probably the can but they could be much more expensive)
- Cannot be eco-friendly (need fluids, power loss due to their moving parts and extra axles)
- Cannot take less space in the cabin (sometimes the place where you rest your hand is a camouflage for the axle that goes to the rear).

On the other side, electric motors just started to exist in automotive industry and probably they have more secrets to reveal! How about changing your car's differential from open to locked in a single update? Now that I have your attention, we can continue!

Formulas

Symbols

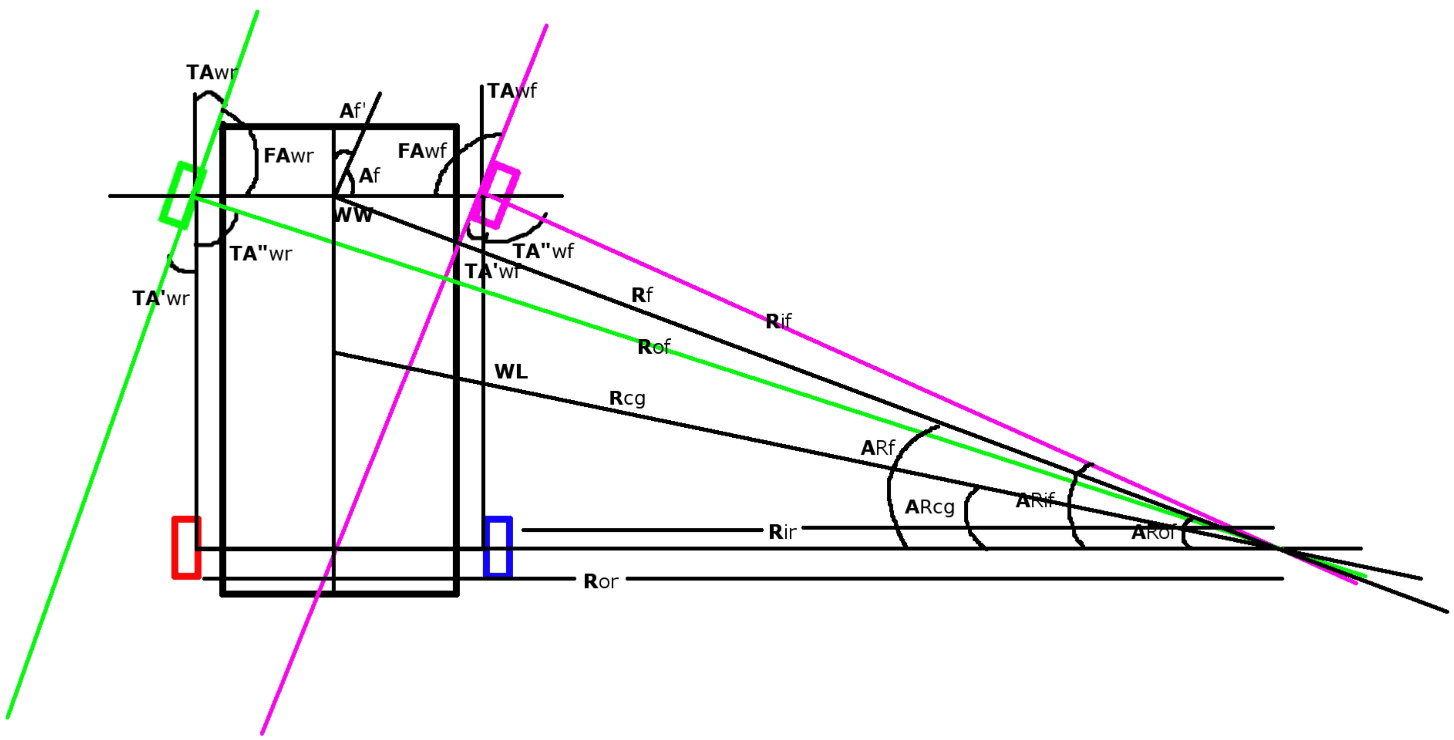
Before starting the calculation, we have to say that we know the following:

- **WL** (*wheel length distance, kingpin to kingpin*),
- **WW** (*wheel width distance, kingpin to kingpin*),
- **WR** (*wheel radius*),
- **TA_{wr} & FA_{wr}** (*outer front wheel turning angle*),
- **TA_{wf} & FA_{wf}** (*inner front wheel turning angle*),
- **Af** (*car turning angle*).

The inner wheel formula should be equal to the real turning angle in order to achieve concentric circles.

The formulas for the Ackerman steering according to this (based on simple trigonometry of course, interior-exterior corresponding angles and triangle angles equality to 180 ^{o}), are:

- $TA_{wf} = AR_{if}$ (**a**)
- $TA_{wr} = AR_{of}$ (**b**)
- $A'_{f} = A_{Rf}$ **(c)**



Prerequisite Formulas

Analyzing the angle equality, we have:

$$TA''_{wf} + TA'_{wf} = 90^o \Rightarrow (\textit{TA}'_{wf} \textit{ and } \textit{TA}_{wf} \textit{ angles are equal})$$

$$TA''_{wf} + TA_{wf} = 90^o \Rightarrow$$

$$TA''_{wf} = 90 - TA_{wf} \quad (\mathbf{d})$$

And

$$TA'_{wr} + TA''_{wr} = 90^o \Rightarrow (\textit{TA}'_{wr} \textit{ and } \textit{TA}_{wr} \textit{ angles are equal})$$

$$TA_{wr} + TA''_{wr} = 90^o \Rightarrow$$

$$TA''_{wr} = 90^o - TA_{wr} \ (\mathbf{e})$$

We could use inverse tangent because we know the angle:

$$TA''_{wf} = \tan^{-1}(R_{ir} / WL) \Rightarrow (\textit{due to } (\boldsymbol{d}))$$

$$(90 - TA_{wf}) = \tan^{-1}(R_{ir} / WL) \quad (\mathbf{f})$$

And

$$TA''_{wr} = \tan^{-1}(R_{or} / WL) \Rightarrow (\textit{due to } (\boldsymbol{e}))$$

$$(90^o - TA_{wr}) = \tan^{-1}(R_{or} / WL) \ (\mathbf{g})$$

Rear Inner Wheel

According to the formula (**f**) we have:

$$(90 - TA_{wf}) = \tan^{-1}(R_{ir} / WL) \Rightarrow$$

$$(90 - TA_{wf}) = \arctan(R_{ir} / WL) \Rightarrow$$

$$\tan(90 - TA_{wf}) = \tan(\arctan(R_{ir} / WL)) \Rightarrow$$

$$\tan(90^o - TA_{wf}) = R_{ir} / WL \Rightarrow$$

$$\tan(90^o - TA_{wf}) * WL = R_{ir}$$

Therefore, the rear inner wheel is moving into a circle with radius:

$$\boxed{R_{ir} = \tan(90^o - TA_{wf}) * WL \ (\mathbf{h})}$$

Rear Outer Wheel

Without further ado, we have:

$$R_{or} = WW + R_{ir}$$

Therefore, the rear outer wheel is moving into a circle with radius:

$$R_{or} = WW + R_{ir} \quad \mathbf{(i)}$$

Front Outer Wheel

Using the steering angles, we have:

$$\sin(A_{Rof}) = WL / R_{of} \Rightarrow$$

$$R_{of} = (WL / \sin(A_{Rof})\ ) \Rightarrow (\textit{due to } \boldsymbol{(b)})$$

$$R_{of} = (WL / \sin(TA_{wr})\ )$$

Therefore, the front outer wheel is moving into a circle with radius:

$$R_{of} = (WL / \sin(TA_{wr})\ ) \quad \mathbf{(j)}$$

Front Inner Wheel

Using the steering angles, we have:

$$\sin(A_{Rif}) = WL / R_{if} \Rightarrow$$

$$R_{if} = (\ WL / \sin(A_{Rif})\ ) \Rightarrow (\textit{due to } \boldsymbol{(a)})$$

$$R_{if} = (\ WL / \sin(TA_{wf})\ )$$

Therefore, the front inner wheel is moving into a circle with radius:

$$R_{if} = (\ WL / \sin(TA_{wf})\ )\ \mathbf{(k)}$$

Center of the car

We are going to use also data about the car as an object. In true Ackerman we have every wheel moving in concentric circles, but in parallel and reverse steering the front wheels are moving in different circles. So, we must find out the circle in which the center of the car is moving.

Every point on the car is moving in concentric circles. This mean that the center of the front axle, the center of the car and the center of the rear axle are moving into concentric circles.

By having a device measuring the angle of the car (A_f) we could calculate the A_f angle.

Let's define R'_{ir} that it is the radius for the center of the circle to the inner rear wheel.

In true Ackerman steering we have $R_{ir}' = R_{ir}$ **(l)** because wheels and car are moving into concentric circles, but at the other two options (parallel, reverse) the car could be moving in non-concentric circle with the wheels. Even the wheels themselves are probably moving in non-concentric circles. Then we have:

$$\tan(A_{Rf}) = WL / (R'_{ir} + WW/2) \Rightarrow (\textit{due to } (\boldsymbol{c}))$$

$$\tan(A'_f) = WL / (R'_{ir} + WW/2) \Rightarrow$$

$$R'_{ir} = (WL / \tan(A'_f)) - WW/2 \ \textbf{(l')}$$

As said before, every point on the car is moving in concentric circles, so the chassis is not able to steer like the front wheels, so the whole car chassis is moving in concentric circles. This means that the straight lines R_f, R_{cg} are starting from the car chassis and they are ending in the same point (center of concentric circles).

Applying Pythagorean theorem and due to above explanation, we have:

$$(R_{cg})^2 = (R'_{ir} + WW/2)^2 + (WL/2)^2$$

Continuing further from Pythagorean equation, for true Ackerman steering:

$$(R_{cg})^2 = (R_{ir}' + WW/2)^2 + (WL / 2)^2 \Rightarrow \textit{(due to (l))}$$

$$(R_{cg})^2 = (R_{ir} + WW/2)^2 + (WL / 2)^2$$

Continuing further from Pythagorean equation, for parallel and reverse steering:

$$(R_{cg})^2 = (R_{ir}' + WW/2)^2 + (WL / 2)^2 \Rightarrow (\textit{due to (l')})$$

$$R_{cg} = SQRT (((WL / \tan(A_f')) - WW/2 + WW/2)^2 + (WL / 2)^2) \Rightarrow$$

$$R_{cg} = SQRT (((WL / \tan(A_f')))^2 + (WL / 2)^2)$$

The formulas should be:

True Ackerman Steering formula: $R_{cg} = SQRT ((R_{ir} + WW/2)^2 + (WL / 2)^2)$ (**m**) Parallel, Reverse Steering Formula $R_{cg} = SQRT (((WL / \tan(A_f')))^2 + (WL / 2)^2)$ (**m'**)

Wheel Rotations

The front outer wheel will travel the longest distance of all four wheels. Our calculations will be based on this wheel because if the driver is in 100% throttle, we cannot make a motor to work at more than 100%. So, the we apply the throttle to the outer front wheel (instead of a motor in simple Evs, or combustion engine on fossil fuel cars).

On the other side, if the outer wheel is not moving, we could apply 0 power, or negative (in a future implementation…probably!).

From paper “Electronic Differential System for an Electric Vehicle with In-Wheel Motor” we have the following formulas for Estimated Rotations (ER):

$ER_{of} = W_{of} = V \times R_{of} / (R_{cg} \times WR)$ (**n**)
$ER_{if} = W_{if} = V \times R_{if} / (R_{cg} \times WR)$ (**o**)
$ER_{or} = W_{or} = V \times R_{or} / (R_{cg} \times WR)$ (**p**)
$ER_{ir} = W_{ir} = V \times R_{ir} / (R_{cg} \times WR)$ (**q**)

for open differential. For locked differential just apply the same rotation (**not PWM value**) value to each wheel. For in-between values just pick some values in-between!

For example:

- Open Differential -> Outer front wheel will have max rotations and the others according to the previous calculations
- Semi-open Differential Front -> Outer front wheel will have max rotations and the opposing wheel will have an in-between (from open to locked) value
- Semi-open Differential Rear -> Outer rear wheel will have max rotations (as max as possible in relation with the outer front wheel) and the opposing wheel will have an in-between (from open to locked) value
- Locked Differential -> Apply everywhere the same rotations

Predefined Table Example

We could use a predefined table in order to apply a close-to-correct PWM value faster. This is an example and should not be used as-is neither in the real world scenario nor in this document!

PWM value (% throttle value)	Rotations @ wheel (per second)			
	Front Left	Front Right	Rear Left	Rear Right
50 (19.61%)	5	6	5	4
100 (39.21%)	12	14	16	11
150 (58.82%)	22	22	18	25
190 (74.51%)	44	43	45	48
220 (86.27%)	50	49	47	53
255 (100%)	56	57	57	55

Vehicle Dynamics

Imagine that you could apply more throttle in the wheel which have more traction…

Later…

Use case (Ackerman scenario)

For a car during a single right turn we have the input:

$$WL = 4m,\ WW = 1.5m,\ TA_{wr} = 19^o,\ TA_{wf} = 21.5^o,\ A_f' = 20.15^o,\ 195/50R15,$$

$$throttle = 50\%$$

and according to the previous equations we have:

(**h**) => $R_{ir} = \tan(90 - TA_{wf}) * WL = \tan(90^o - 21.5^o) * 4 = 10.155m$

(**i**) => $R_{or} = WW + R_{ir} = 1.5 + 10.154 = 11.655m$

(**j**) => $R_{of} = (WL / \sin(TA_{wr})) = 4 / \sin(19^o) = 12.286m$

(**k**) => $R_{if} = (WL / \sin(TA_{wf})) = 4 / \sin(21.5^o) = 10.914m$

And the rim calculations are:

RimRadius = (15 * 0.0254) / 2 = 0.1905m

TyreProfile = (0.195 * 0.5) = 0.0975m

WR = RimRadius + TyreProfile = 0.1905 + 0.0975 = 0.288m

Calculating the wheel rotations, we have:

For throttle = 50%, we have 127.5 of 255 PWM value, so we apply this PWM to outer front wheel and calculate its rotations by a speed sensor. Let's say we have measured 17 rotations per second.

Parallel, reverse	Ackerman only
(**m'**) => $R_{cg} = SQRT (((WL / \tan(A_f')))^2 + (WL / 2)^2)$ => $R_{cg} = SQRT (((4 / \tan(20.15)))^2 + (4 / 2)^2)$ => $R_{cg} = SQRT ((10.901)^2 + 4)$ => $R_{cg} = 11.083m$	(**m**) => $R_{cg} = SQRT ((R_{ir} + WW/2)^2 + (WL / 2)^2)$ => $R_{cg} = SQRT((10.155 + 0.75)^2 + (4/2)^2)$ => $R_{cg} = 11.087m$
(**n**) => $W_{of} = V \times R_{of} / R_{cg} \times WR$ => 17 = V * 12.286 / 11.083 * 0.288 => 17 = V * 12.286 / 3.191904 =>	(**n**) => $W_{of} = V \times R_{of} / R_{cg} \times WR$ => 17 = V * 12.286 / 11.087 * 0.288 => 17 = V * 12.286 / 3.193056 =>

$V = 17 * 3.191904 / 12.286 \Rightarrow$ $V = 4.417\ {}^{m}/_{s}$	$V = 17 * 3.193056 / 12.286 \Rightarrow$ $V = 4.418\ {}^{m}/_{s}$
(**o**) => $W_{if} = V \times R_{if} / (R_{cg} \times WR) \Rightarrow$ $W_{if} = 4.417 * 10.914 / 11.083 * 0.288 \Rightarrow$ $W_{if} = 15.103$ rps	(**o**) => $W_{if} = V \times R_{if} / (R_{cg} \times WR) \Rightarrow$ $W_{if} = 4.418 * 10.914 / 11.087 * 0.288 \Rightarrow$ $W_{if} = 15.101$ rps
(**p**) => $W_{or} = V \times R_{or} / (R_{cg} \times WR) \Rightarrow$ $W_{or} = 4.417 * 11.655 / 11.083 * 0.288 \Rightarrow$ $W_{or} = 16.12$ rps	(**p**) => $W_{or} = V \times R_{or} / (R_{cg} \times WR) \Rightarrow$ $W_{or} = 4.418 * 11.655 / 11.087 * 0.288 \Rightarrow$ $W_{or} = 16.126$ rps
(**q**) => $W_{ir} = V \times R_{ir} / (R_{cg} \times WR) \Rightarrow$ $W_{ir} = 4.417 * 10.155 / 11.083 * 0.288 \Rightarrow$ $W_{ir} = 14.053$ rps	(**q**) => $W_{ir} = V \times R_{ir} / (R_{cg} \times WR) \Rightarrow$ $W_{ir} = 4.418 * 10.155 / 11.087 * 0.288 \Rightarrow$ $W_{ir} = 14.051$ rps
$W_{FrontLeft} = 17$rps $W_{FrontRight} = 15.103$ rps $W_{rearLeft} = 16.12$ rps $W_{rearRight} = 14.053$ rps	$W_{FrontLeft} = 17$rps $W_{FrontRight} = 15.101$ rps $W_{rearLeft} = 16.126$ rps $W_{rearRight} = 14.051$ rps

Flowchart

This flowchart is in early stage; so:

- It is not describing an algorithm in actual code,
- It needs an algorithm to calculate the PWM value for specified rotations,
- It has some restrictions (such as the differential settings could be applied when car is not moving).
- It cannot calculate ranges 29.5^o in a turning wheel is different from 29.6^o. This should not happen, at least in small speeds!
- It lacks hardware features (such us interruption) because I still don’t have a lot of experience on these. Programmed I/O is a good start, though!
- I have not decided if this software will work on a controller or on a OS using threads.