OULUN YLIOPISTO
UNIVERSITY of OULU

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CLARA GIFRE OLIVERAS

ADAPTIVE MOTION PLANNING FOR A MOBILE ROBOT

Master's Thesis
Degree Programme in Computer Science and Engineering
June 2013

Gifre Oliveras, C. (2013) Adaptive motion planning for a mobile robot. University of Oulu, Department of Computer Science and Engineering. Master's Thesis, 76 p.

ABSTRACT

Historically, trapezoidal velocity profiles have been widely used to control engines. Nevertheless, the evolution of robots and their uses has led to the need of using smoother profiles, due to the demand of high precision and delicate movements. It has been shown that this can be achieved by minimizing the change of acceleration and using s-curve profiles. Moreover, to provide a good control of the movement of a robot, it is necessary to ensure that it will meet the desired velocity profile. Therefore, a way to prevent how the wheels will react on the soil becomes highly useful, in order to adapt the supplied torque.

This thesis suggests a model to define an appropriate s-curve velocity profile given the desired starting and ending kinematic states for a mobile robot. The study is then focused on a one-wheel system to define the interaction between the soil and a wheel. This interaction is modelled and extended in order to calculate the required torque, drawbar pull and power needed to fulfil the desired s-curve velocity profile. Finally, an introduction to unicycle robots is given as an example of how the proposed models could be applied in the motion planning of a mobile robot.

Key words: terramechanics, s-curve, jerk, velocity profile

TABLE OF CONTENTS

FOREWORD

This thesis was carried out at the Department of Computer Science and Engineering at the University of Oulu in fulfilment of the requirements for acquiring a degree in Industrial Engineering by Escola Tècnica Superior Industrial de Barcelona, Universitat Politècnica de Catalunya.

The topic under consideration was proposed by Professor Juha Röning, head of the CSE Department of University of Oulu, and has been supervised by Mr. Antti Tikanmaki, M. Sc. (Eng).

The aim of this project is to provide a model to obtain an s-curve velocity profile given the desired starting and ending kinematic states for a mobile robot. Afterwards, a soil-wheel interaction model is proposed in order to calculate the required torque, drawbar pull and power needed by one wheel to accomplish the desired s-curve velocity profile. Then, an example of motion planning of a two-wheeled mobile robot is given.

I would like to express my sincere gratitude to Prof. Röning and Mr.Tikanmaki for including me in their team and guiding me through the writing of this project. Besides, I thank almighty my parents, who have provided me with invaluable support and have helped broaden my horizons during my whole life. I deeply thank my cousin Minerva, who has proofread the whole project. I also take this opportunity to generally thank all my family and friends, the ones who have been supporting me from anywhere in the world and the ones who are here, in Oulu, and have become like a second family. Without their constant encouragement this thesis would not have been possible.

LIST OF ABBREVIATIONS AND SYMBOLS

$\widetilde{F_x}$	rolling resistance force
${F_\eta}^{(a)}$	resulting lateral contact forces for area (a)
${F_\eta}^{(b)}$	resulting lateral contact forces for area (b)
${F_\xi}^{(a)}$	resulting longitudinal contact forces for area (a)
${F_\xi}^{(b)}$	resulting longitudinal contact forces for area (b)
${X_q}^{max}$	maximum value of a kinematic feature
$\bar{\sigma}_p$	mean of the passive stresses from the grousers in contact with the terrain
A_g	amplitude of the oscillation
A_{max}	maximum acceleration value
A_γ	amplitude of oscillation related to the change in the local soil density around the wheel and grouser caused by the soil deformation due to the wheel
A_σ	amplitude of oscillation related to the active and passive stresses
C_d	constant
C_m	constant
F_x	horizontal load on the wheel
F_z	vertical force
F_η	lateral contact force
F_ξ	longitudinal contact force
J_k	jerk absolute value at the kth phase
N_i	total number of interpolation steps
N_ϕ	flow value
R_c	compaction resistance
S_s	longitudinal slip rate
S_{sc}	critical value of the longitudinal slip rate
S_α	lateral slip rate
$S_{\alpha c}$	critical value of the lateral slip rate
T_k	duration of the kth phase
T_s	control loop sampling period
V_L	deformation rate in the tangential direction
V_{NS}	deformation rate in the normal direction
V_{max}	maximum velocity value
V_s	relative slippage velocity between wheel and soil
V_x	translational velocity of the wheel centre
X_k^{peak}	peak values of kinematic features
a_{0k}	acceleration value at the beginning of the kth phase
d_γ	local change in weight density of soil around the wheel
f_{0k}	federate value at the beginning of the kth phase
f_e	final feedrate
f_k	feedrate at the end of the kth phase
f_s	initial feedrate
$j_{N_{max}}$	maximum accumulated deformation in the normal direction
j_{0k}	jerk value at the beginning of the kth phase
j_L	accumulated soil deformation in the tangential direction

j_N	accumulated deformation in the normal direction
$\hat{k}$	fitting constant
k_1	constant from soil penetration plate tests
k_2	constant from soil penetration plate tests
$k_a{}'$	empirical dimensionless coefficient
k_c	cohesive moduli of sinkage
$k_c{}'$	dimensionless soil parameter
k_e	relative model of elasticity for rebound stress in the soil
k'_g	empirical dimensionless coefficient
k_z	vertical stiffness
k_ϕ	frictional moduli of sinkage
$k_\phi{}'$	dimensionless soil parameter
l_c	contact length of the wheel on the soil
l_k	distance travelled during the kth phase
l_r	length of area (a) in the model proposed in [1]
l_s	length of area (b) in the model proposed in [1]
$\widehat{m}$	fitting constant
$minu_0$	minimum deformation rate for speed influence
$\hat{n}$	fitting constant
n_g	number of grousers of the wheel
s_{0k}	displacement value at the beginning of the kth phase
s_{kn}	total distance travelled from the beginning of the kth phase until the nth interpolation step in this phase
$t_0, \dots, t_k$	time value at instant k
v_b	vehicle’s speed
z_0	distance from the soil surface level to the point of the deepest sinkage, under the centroid of the wheel
δ_r	vertical deformation for the wheel
δ_s	vertical deformation for the soil
δ_{target}	target distance
θ_1	entry angle
θ_2	exit angle
θ_f	forward wheel-soil contact angle (entry angle of the interaction)
θ_m	angle at which maximum normal stress occurs
θ_r	rear wheel-soil contact angle(departure angle of the interaction)
θ_s	static wheel-soil contact angle
μ_0	static friction coefficient
μ_D	dynamic friction coefficient
μ_{cx}	friction coefficient μ_c with respect to longitudinal axel
μ_{cy}	friction coefficient μ_c with respect to lateral axel
τ_k	relative time parameter that starts at the beginning of the kth phase
τ_m	maximum shear stress
τ_{max}	shear strength of the soil
ω_g	frequency of the oscillations
I	inertia
L	angular momentum
T	torque or momentum

A	acceleration value at the constant acceleration phase
C	apparent soil cohesion. It measures the attractive force between soil particles
D	deceleration value at the constant deceleration phase
D	wheel diameter
DP	drawbar pull
F	desired feedrate
H	thrust available to a vehicle on deformable terrain
J	desired jerk magnitude
K	soil shear deformation modulus
L	total distance of travel
Q	torque
R	total soil resistance
TE	tractive efficiency
V	velocity value at the constant velocity phase
W	wheel's vertical load
Z	height of the wheel axle from the surface
a	acceleration
b	wheel width
c	soil cohesion
f	feedrate
j	jerk
j	the shear displacement along the wheel-soil interface
k	sinkage constant, being b the width of the wheel or plate
l	wheel-soil contact patch length
m	constant from soil penetration plate tests
n	constant from soil penetration plate tests
n	sinkage exponent
r	wheel's radius
s	slip
t	time
v	velocity
x	position, displacement
z	distance from the soil level to the point of maximum wheel pressure
α	constant
α	slip angle
β	slip velocity's angle of direction from the longitudinal axle
η	experimental parameter that relates the measured sinkage to the depth of the track after the wheel has passed
θ	arbitrary angle along the stress arc
κ	shear deformation parameter
μ	friction coefficient between soil and wheel
σ	normal stress
$\sigma(\theta)$	normal stress at angle θ
τ	tangential stress
$\tau(\theta)$	shear stress at angle θ
ω	wheel's angular velocity in rad/s
ϕ	soil's internal friction angle

1. INTRODUCTION

A mobile robot is an automatic machine capable of moving in a given environment. With respect to its trajectory, one of its main requirements is not to collide with any obstacle. Furthermore, it should satisfy certain constraints concerning time, velocity, acceleration or jerk, while moving from a starting point to a target point as smoothly as possible.

The process of obtaining the needed sequence of actions to be executed by the robot is known as motion planning. The performance of the system is called adaptive because it has to be adjusted during the execution of these actions according to the environment.

Assuming that the environment and soil characteristics are well known, the motion of the robot could be divided into two areas of study. The first one is the determination of the required trajectory, velocity, acceleration and jerk profiles by the robot at all times. The second issue would be the determination of the necessary torque and power for each wheel at every instant, in order to fulfil the calculated kinematic parameters.

The main objective of this thesis is to define an appropriate s-curve velocity profile for a robot to follow, starting from the assumption that the desired kinematic parameters and the starting and target place and time are given. Afterwards, the power, drawbar pull and torque profiles required to accomplish this velocity profile are calculated, taking into consideration the dynamic restrictions of the robot and the interaction between the wheels and the soil. Finally, two-wheeled robots are studied as an example of how the proposed models could be applied in the motion planning of a mobile robot.

First, a review of basic kinematics and a research on the state of the art in the field of velocity profiles is performed, in order to define a model that could be implemented in Python language to provide s-curve velocity profiles from given kinematic parameters.

Then, the study is focused on terramechanics, which is the area of mobility research that studies the performance of vehicles relating to its interaction with the environment. A model for the interaction between a moving wheel and the soil in which it moves is proposed, and implemented as part of another Python program in order to calculate the required torque, drawbar pull and power. Besides, a study of the effects of the dimension of the wheels and the characteristics of the soil is performed.

Finally, a suggestion on how the proposed one-wheel system model could be adapted into a robot is given, illustrating it in the form of a two-wheel Segway-based robot.

2. BASIC KINEMATICS

Historically, trapezoidal velocity profiles have been used as a reference, mainly because they are easy to be modelled and implemented. Despite the fact that it is not possible to obtain a perfect real trapezoidal response, their performance in the field has been quite good.

Nevertheless, given the evolution experienced by robots during the last years, these profiles are no longer suitable enough for engines involved in precision or delicate works. Modern machines need to operate at high feedrates and accelerations, which can result in undesirable high frequency harmonics in the reference trajectory and even a saturation of the actuators [2]. Due to these facts, it is necessary to develop a way to generate smooth profiles. S-curve models have shown to meet this requirement.

2.1. Jerk

Given the position of a particle, x, successively deriving, its speed, v, acceleration, a and jerk, j can be obtained (Eq. 1). Thus, jerk is the time rate of change of acceleration. Minimizing acceleration changes results on a smoother velocity profile. Hence, the appropriate way to proceed when designing a velocity profile is to minimize jerk.

$$j = \frac{\partial a}{\partial t} = \frac{\partial^2 v}{\partial t} = \frac{\partial^3 x}{\partial t} \left[\frac{m}{s^3}\right] \qquad \text{(Eq. 1)}$$

2.2. Velocity profiles

If the jerk profile is known, acceleration (Eq. 2), velocity or feedrate (Eq. 3) and displacement (Eq. 4) profiles can be obtained by integrating the jerk profile $j(t)$:

$$a(t) = a(t_i) + \int_{t_i}^{t} j(\tau_i) d\tau_i \qquad \text{(Eq. 2)}$$

$$v(t) = v(t_i) + \int_{t_i}^{t} a(\tau_i) d\tau_i \qquad \text{(Eq. 3)}$$

$$x(t) = s(t_i) + \int_{t_i}^{t} v(\tau_i) d\tau_i \qquad \text{(Eq. 4)}$$

Given a velocity profile, the area under the curve corresponds to the position of the particle. When moving from one position to another in a given time, there are infinite possible velocity profiles. The only similitude they must hold is to include the same area under the curve. Nevertheless, most of them can be summarized in the following three cases: constant velocity window, trapezoidal velocity profile and S-curve models [3].

For a constant velocity window, acceleration has impulses at an infinite value at the beginning and at the end of the motion, which would require an infinite torque. In contrast, for simple trapezoidal velocity models, acceleration makes abrupt changes, which cause jerk to get impulses at an infinite value. Moreover, due to mechanical limitations, it is difficult for a real machine to follow a trapezoidal profile. S-curve models aim to reduce these problems [4].

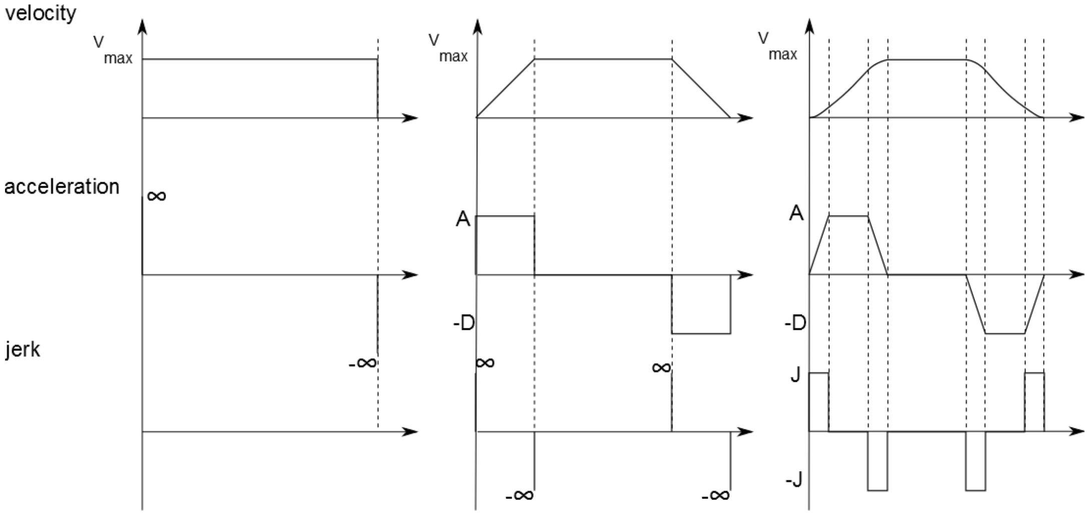

Figure 1. Constant, trapezoidal and s-curve velocity profiles.

From Figure 1, it can be seen that, instead of jumping from zero to an acceleration (or deceleration) peak value, the acceleration curve related to a velocity S-curve changes gradually. Therefore, jerk does not exhibit infinite values, but a certain value.

2.2.1. Trapezoidal profile description

Among all of the possible trapezoidal profiles, the best option is to choose the one that minimizes the needed power to be supplied by the engine. According to [5], this would be the ⅓-⅓-⅓ law, which is schematized in Figure 2.

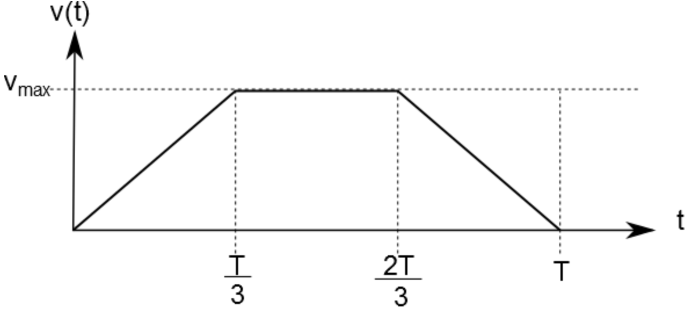

Figure 2. Trapezoidal ⅓-⅓-⅓ velocity profile law.

2.2.2. S-curve profile description

The aim of s-curve profiles is to keep precision and speed while reducing the residual vibration. As pointed before, jerk achieves finite values for an s-curve, which makes the movement more fluent. Many different approaches have been suggested on the generation of s-curves. Some of them are briefly explained under these lines. More information on the subject can be found in Appendix A.

State of the art

One of the reference models in the field of s-curve profile generation is the one proposed by Erkorkmaz and Altintas [2]. The suggested model imposes limits to both the first and second derivatives of the feedrate in order to obtain 3rd order smooth velocity profiles using an iterative algorithm. Yong Jeong et al. [6] propose a similar approach, but without the need of iterating, while in [4] general equations are used in order to allow the description of an asymmetric curve. An even more general algorithm is the one in [3], which can be used to define generic s-curves of any order.

Proposed algorithm

The proposed algorithm aims to be as generic as possible, so that it can be adapted to the desired characteristics. Therefore, the possibilities of having required values for initial and final acceleration, velocity, position and time are considered. This is, both zero and non-zero values are considered.

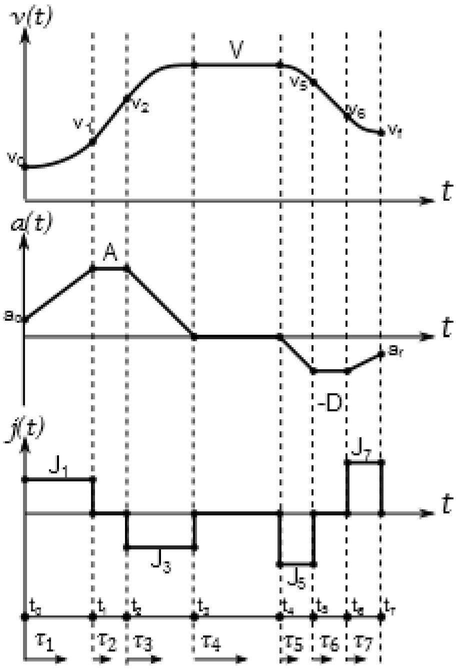

Figure 3. Velocity, acceleration and jerk profiles of the proposed model.

The suggested motion equations are similar to the ones introduced in [2]. They refer to a 3rd order s-curve model, which is simpler than a 4th order one and, consequently, has less computational load when being implemented. Besides, it fulfils the specifications related to the jerk values. In Figure 3 there is an example of the profiles that can be obtained.

From the start, the integrating constants are kept during the generation of the equations. When defining the profiles as piecewise equations, it is necessary to indicate the continuity between consecutive parts. Then, the values of most of the integration constants become defined. If possible, they are assigned their resulting value. Otherwise, the correspondent equations are added as restrictions. In addition, local time variables are introduced in order to ease the reading and computation of the formulas.

$$\tau_k = \mathrm{t} - t_{k-1}, \qquad k = 1, \dots, 7 \qquad \text{(Eq. 5)}$$

In Table 1 the initial conditions of the model are summarized. Then, the motion equations are presented, followed by the restrictions that the constants should meet in order to ensure continuity between sections of the piecewise equations.

Initial conditions

Table 1. Initial conditions for the proposed model for s-curve generation

acceleration		velocity	position
$a(t=0) = a_0$ $a(t=t_7) = a_f$ $a(\tau_2) = A$	$a(\tau_4) = 0$ $a(\tau_6) = -D$	$v(t=0) = v_0$ $v(t=t_7) = v_f$ $v(\tau_4) = V$	$x(t=0) = x_0$ $x(t=t_7) = x_f$

Motion equations

$$j(\tau) = \begin{cases} J_1, & 0 \le t < t_1 \\ 0, & t_1 \le t < t_2 \\ -J_3, & t_2 \le t < t_3 \\ 0, & t_3 \le t < t_4 \\ -J_5, & t_4 \le t < t_5 \\ 0, & t_5 \le t < t_6 \\ J_7, & t_6 \le t \le t_7 \end{cases} \qquad \text{(Eq. 6)} \qquad a(\tau) = \begin{cases} J_1\tau_1 + a_0, & 0 \le t < t_1 \\ A, & t_1 \le t < t_2 \\ A - J_3\tau_3, & t_2 \le t < t_3 \\ 0, & t_3 \le t < t_4 \\ -J_5\tau_5, & t_4 \le t < t_5 \\ -D, & t_5 \le t < t_6 \\ -D + J_7\tau_7, & t_6 \le t \le t_7 \end{cases} \qquad \text{(Eq. 7)}$$

(Eq. 8) and (Eq. 9) apply for the same intervals that those for jerk and acceleration, although this information has been suppressed in order to improve readability.

$$v(\tau) = \begin{cases} v_0 + a_0\tau_1 + \frac{1}{2}J_1{\tau_1}^2 \\ v_1 + A\tau_2 \\ v_2 + A\tau_3 - \frac{1}{2}J_3{\tau_3}^2 \\ V \\ V - \frac{1}{2}J_5{\tau_5}^2 \\ v_5 - D\tau_6 \\ v_6 - D\tau_7 + \frac{1}{2}J_7{\tau_7}^2 \end{cases} \qquad \text{(Eq. 8)} \qquad x(\tau) = \begin{cases} v_0\tau_1 + \frac{a_0{\tau_1}^2}{2} + \frac{J_1{\tau_1}^3}{6} + x_0 \\ v_1\tau_2 + \frac{A{\tau_2}^2}{2} + x_1 \\ v_2\tau_3 + \frac{A{\tau_3}^2}{2} - \frac{J_3{\tau_3}^3}{6} + x_2 \\ V\,\tau_4 + x_3 \\ V\tau_5 - \frac{J_5{\tau_5}^3}{6} + x_4 \\ v_5\tau_6 - \frac{D{\tau_6}^2}{2} + x_5 \\ v_6\tau_7 - \frac{D{\tau_7}^2}{2} + \frac{J_7{\tau_7}^3}{6} + x_6 \end{cases} \qquad \text{(Eq. 9)}$$

Where t_k is the time value at instant k, τ_k is a relative time parameter that starts at the beginning of the k^th phase, x, v, a and j stand for position, velocity, acceleration and jerk, $x_0, \dots, x_k, v_0, \dots, v_k, a_0, \dots, a_k$ are position, velocity and acceleration values at t_k, J_1,J_3,J_5,J_7, are the jerk values at $\tau_1, \tau_3, \tau_5, \tau_7$, respectively, and A, D, V are the values for acceleration, deceleration and velocity at their correspondent constant-value phases.

Restrictions

Since velocity and position have more complex expressions than acceleration, it is not that trivial to get the values from all of the integration constants. This is why most of them are not substituted by their expressions in the equations, but listed below (Eq. 10)(Eq. 26).

$$v_1 = J_1 \frac{(t_1 - t_0)^2}{2} + a_0(t_1 - t_0) + v_0 \qquad \text{(Eq. 10)}$$

$$v_2 = A(t_2 - t_1) + v_1 \qquad \text{(Eq. 11)}$$

$$V = -J_3 \frac{(t_3 - t_2)^2}{2} + A(t_3 - t_2) + v_2 \qquad \text{(Eq. 12)}$$

$$v_5 = -J_5 \frac{(t_5 - t_4)^2}{2} + V \qquad \text{(Eq. 13)}$$

$$v_6 = -D(t_6 - t_5) + v_5 \qquad \text{(Eq. 14)}$$

$$v_6 = D(t_7 - t_6) + v_f - J_7 \frac{(t_7 - t_6)^2}{2} \qquad \text{(Eq. 15)}$$

$$v_0(t_1 - t_0) + \frac{a_0(t_1 - t_0)^2}{2} + \frac{J_1(t_1 - t_0)^3}{6} + x_0 = x_1 \qquad \text{(Eq. 16)}$$

$$v_1(t_2 - t_1) + \frac{A(t_2 - t_1)^2}{2} + x_1 = x_2 \qquad \text{(Eq. 17)}$$

$$v_2(t_3 - t_2) + \frac{A(t_3 - t_2)^2}{2} - \frac{J_3(t_3 - t_2)^3}{6} + x_2 = x_3 \qquad \text{(Eq. 18)}$$

$$V\,(t_4 - t_3) + x_3 = x_4 \qquad \text{(Eq. 19)}$$

$$V(t_5 - t_4) - \frac{J_5(t_5 - t_4)^3}{6} + x_4 = x_5 \qquad \text{(Eq. 20)}$$

$$v_5(t_6 - t_5) - \frac{D(t_6 - t_5)^2}{2} + x_5 = x_6 \qquad \text{(Eq. 21)}$$

$$v_6(t_7 - t_6) - \frac{D(t_7 - t_6)^2}{2} + \frac{J_7(t_7 - t_6)^3}{6} + x_6 = x_7 \qquad \text{(Eq. 22)}$$

$$a_0 + J_1(t_1 - t_0) = A \qquad \text{(Eq. 23)}$$

$$A - J_3(t_3 - t_2) = 0 \qquad \text{(Eq. 24)}$$

$$-J_5(t_5 - t_4) = -D \qquad \text{(Eq. 25)}$$

$$-D + J_7(t_7 - t_6) = a_f \qquad \text{(Eq. 26)}$$

The scope of this thesis is such that the values of D, A and jerk are considered input data. On further studies, these values should be computed regarding mechanical limitations and external specifications, in order to minimize the power to be supplied by the engine that moves the wheel.

3. MOVEMENT OF A WHEEL: INVOLVED PARAMETERS

One of the most critical issues of the movement of a robot is to neutralize how terrain properties affect the mobility of the wheels. The area of mobility research that studies the performance of vehicles in relation to its operating environment is called terramechanics. According to [7], the aim of terramechanics is to provide guiding principles for the rational design, evaluation, selection and operation of terrestrial and extra-terrestrial vehicles or machines.

For autonomous robots moving on soft soils, for instance sand or snow, the effects on the wheel-soil interface become important. Slippage and sinkage can result in low mobility or even immobility. The comprehension of these characteristics is relevant to ensure a good movement control. As stated in [8], it is desirable to somehow sense excessive wheel slippage and sinkage in order to prevent the robot to become immobile.

In order to anticipate the requirements of the robot in terms of power and torque when fulfilling a predetermined velocity profile, it is necessary to include these effects in the calculus, so that it does not meet with problems as excessive sinkage or lack of power when moving.

3.1. Pressure-sinkage

Sinkage is a measure of a soil's vertical deformation. Depending on whether the vehicle is moving or not, the sinkage can be dynamic or static, respectively. In soft soils, wheels can sink sufficiently to be immobilized.

The parameters with more influence on this value are the compactness of the soil, the load on the wheel and the velocity that the vehicle moves with. The load on the wheel interacts with the soil in the form of a pressure applied along the contact area between the wheel and the ground.

3.1.1. Classic terramechanics

Most of the studies related to terramechanics that are carried on nowadays, are based on the fundamentals of this field [9], which lead in the theory developed in the 1960's by M.G. Bekker [10], which was later improved by Wong and Reece [11]. Given that, it is convenient to review these before deepening in the state of the art on the subject.

Bekker theory

Bekker theory [10] is the base of classical terramechanics. The proposed model is mostly based on a set of semi-empirical equations that describe the locomotion of a vehicle on a deformable soil. The assumption of the approximation of the contact patch between wheel and soil as a flat plate is taken. This hypothesis leads to accurate predictions for big, heavy vehicles, but leads to errors for small, light vehicles.

A plate sinkage can be defined using (Eq. 27), where σ is the applied pressure, z is the resultant sinkage and k and n are soil parameters chosen to fit experimental data for a particular soil:

$$\sigma = kz^{n} \qquad \text{(Eq. 27)}$$

$$k = \frac{k_c}{b} + k_{\phi} \qquad \text{(Eq. 28)}$$

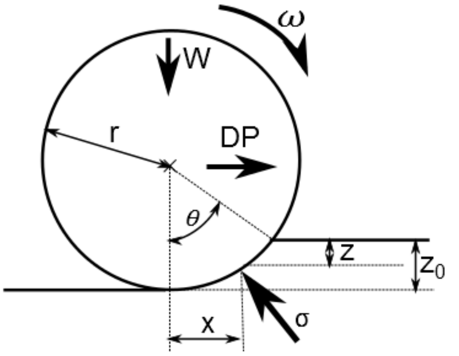

Figure 4. Rigid wheel sinkage according to Bekker's model.

From the geometric relations between parameters from Figure 4, (Eq. 29) and (Eq. 30) can be obtained. Besides, under the supposition that the value of sinkage is small compared to the diameter of the wheel, (Eq. 30) can be simplified as (Eq. 31).

$$W = b\int_0^{\theta} \sigma\, r\, cos\theta\; d\theta = \int_0^{z_0} \left(\frac{k_c}{b} + k_{\phi}\right) z^{n} dx \qquad \text{(Eq. 29)}$$

$$x^2 = [d - (z_0 - z)]\,(z_0 - z) \qquad \text{(Eq. 30)}$$

$$x^2 = d(z_0 - z) \qquad \text{(Eq. 31)}$$

Where d and r are the diameter and radius of the wheel, W is the load on it and z_0 is the maximum sinkage. Using (Eq. 29) and (Eq. 31) and integrating, the Bekker expression for large rigid wheel sinkage is obtained:

$$z_0 = {\frac{3W}{b(3-n)\left(\frac{k_c}{b} + k_{\phi}\right)\sqrt{d}}}^{\frac{2}{2n+1}} \qquad \text{(Eq. 32)}$$

Then, the compaction resistance, R_c, can be calculated using:

$$R_c = b\int_0^{\theta} \sigma r\; sin\theta\; d\theta = b\int_0^{z_0} \left(\frac{k_c}{b} + k_{\phi}\right) z^{n} dz \qquad \text{(Eq. 33)}$$

$$R_c = \frac{\left(\frac{3W}{\sqrt{d}}\right)^{\frac{2n+2}{2n+1}}}{(3-n)^{\frac{2n+2}{2n+1}}(n+1)b^{\frac{1}{2n+1}}} \frac{1}{\left(\frac{k_c}{b} + k_{\phi}\right)^{\frac{1}{2n+1}}} \qquad \text{(Eq. 34)}$$

As stated before, this approach leads to errors when studying small vehicles. According to Bekker [10]: "Predictions for wheels smaller than 20 inches in diameter become less accurate as wheel diameter decreases, because sharp curvature of the loading area was neither considered in its entirety nor is it reflected in bevameter tests".

In [12] an experiment is carried out with the aim to prove the accuracy of Bekker's model for a 0.17m diameter wheel. For a 46N load, the results show errors in sinkage estimation of 33%, 50.8% in R_c value and 40.5% for the drawbar pull. Moreover, the error values grow when increasing the applied load.

In conclusion, Bekker's equations are not accurate for small wheels. Nevertheless, several modifications on the equation have been proposed along these years in order to account for the sharp curvature of small wheels.

Wong and Reece

Wong and Reece [11] proposed another formulation of the relationship between pressure and sinkage, which later would be converted to a polar form to describe the normal stress acting on the wheel.

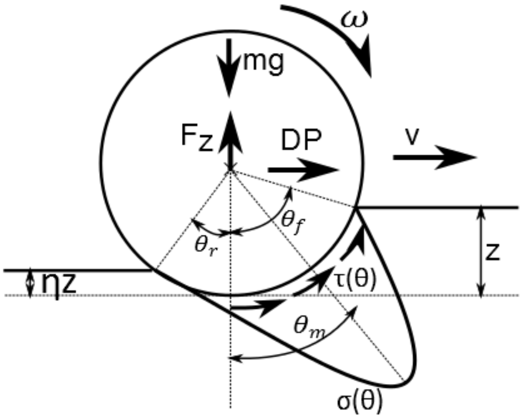

Figure 5. Forces and stress acting on a rigid wheel according to Wong and Reece's model.

$$p(z) = \left(ck_c' + \gamma bk_\phi'\right)\left(\frac{z}{b}\right)^n \quad \text{(Eq. 35)}$$

$$\sigma(\theta)\begin{cases} r^n k(cos\theta - cos\theta_f)^n, & \theta_m \le \theta < \theta_f \\ r^n k\left(cos\theta_f - \dfrac{\theta-\theta_r}{\theta_m-\theta_r}\left(\theta_f-\theta_m\right) - cos\theta_f\right)^n, & \theta_r \le \theta < \theta_m \end{cases} \quad \text{(Eq. 36)}$$

Where k_c' and $k_\phi{}'$ are dimensionless soil parameters,$\sigma(\theta)$ is the normal stress at angle θ, θ_r is the rear wheel-soil contact angle, θ_f is the forward wheel-soil contact angle and θ_m is the angle at which maximum normal stress occurs.

$$\theta_f = arccos\left(1 - \frac{z}{r}\right) \quad \text{(Eq. 37)}$$

$$\theta_r = arccos\left(1 - \frac{\eta z}{r}\right) \quad \text{(Eq. 38)}$$

$$\theta_m = (b_0 + b_1 s)\theta_f \quad \text{(Eq. 39)}$$

With $b_0 \approx 0.4$ and $0.0 \le b_1 \le 0.3$.

In loose soil, the shear stress, which is the stress acting parallel to the wheel at the soil-wheel contact point, exhibits an exponential relationship with respect to the shear displacement along the wheel-soil interface ($j(\theta)$).

$$\tau(\theta) = (c + \sigma(\theta) tan\phi)\left[1 - e^{-\frac{j(\theta)}{K}}\right] \qquad \text{(Eq. 40)}$$

Being $j(\theta)$ the resulting expression from (Eq. 43).

$$j(\theta) = r\left[\theta_f - \theta - (1 - s)\left(sin\theta_f - sin\theta\right)\right] \qquad \text{(Eq. 41)}$$

Where s is the slippage, $\tau(\theta)$ is the shear stress at angle θ and η is an experimental parameter that relates the measured sinkage to the depth of the track after the wheel has passed.

Then, the vertical force, F_z, and the drawbar pull, DP, can be calculated by integrating the stresses in the vertical and horizontal directions, respectively.

$$F_z = rb\int_{\theta_r}^{\theta_f} (\tau(\theta) sin\theta + \sigma(\theta)\cos\theta) d\theta \qquad \text{(Eq. 42)}$$

$$DP = F_x = rb\int_{\theta_r}^{\theta_f} (\tau(\theta) cos\theta - \sigma(\theta) sin\theta) d\theta \qquad \text{(Eq. 43)}$$

3.1.2. State of the art based on classic terramechanics

The previously introduced models are the basics of terramechanics, which mainly focus on big terrestrial vehicles. According to [13], it is necessary to prove its appliance on planetary rovers, since they have certain different characteristics, such as the size of the wheels or the weight of the vehicle. This also applies to terrestrial robots.

Historically, the main assumption has been that the soil reaction to the stresses at the wheel is similar to the soil reaction given under penetration and shear tests. The approach given by Wong and Reece is useful for isotropic materials. They proposed penetration plate measurements to characterize the soil compression stresses in the normal direction and shear tests for the tangential ones. Grahm [14] used the penetration plate test to describe soil behaviour in the vertical direction and shear in the horizontal one, which is appropriate for anisotropic materials.

Meiron-Griffith and Spenko [12] modify Bekker’s equations to take into account the effect of the diameter of the wheels. However, these equations do not include the semi-elliptical distribution of normal stress that exists beneath a wheel. This is what they introduce to the model in [15], starting from Wong and Reece’s model.

In [1], Djohor et al. propose the decomposition of the contact surface in two parts, one behaving as a rigid wheel-deformable soil system and another where the wheel gets both vertical and tangential reactions from the soil. The resistive forces are computed using Bekker’s formula.

In [16], the approach used by Wong and Reece is slightly modified in order to be able to predict off-road wheel performances, while considering the soil deformation. The performed simulations show that the effect of velocity on wheel performances cannot be neglected.

With the purpose of improving its tractive effort, some terrestrial robots and vehicles use wheels with grousers. The grousers cause oscillations in sinkage, drawbar and normal force, and these effects are not represented in the traditional terramechanics models, as shown in [9].

Up to the date, finite and discrete element methods have been suggested to solve this problem, but these algorithms need powerful computers to model the wheel-soil interaction. Irani et al. [9] include an oscillating term in the equations from Wong and Reece so that the final model behaves according to the effect of the grousers. Their approach slightly improves the traditional terramechanics equations, but further work would be necessary to understand all of the involved parameters, as well as to obtain a precise model. In [13], some experiments are conducted to analyse the effect of several factors on the wheel performance of planetary rovers. The main conclusion is that in order to improve the drawbar pull performance, an increase of the width, radius and lug height of the wheels would be useful.

3.2. Slippage

On loose soil, it is easy for the wheels to slip or spin and, consequently, loose traction. When a wheel slips, the soil beneath it is removed so that the wheel sinks. Slip is the difference between the theoretical circumference velocity and the actual travelling velocity of the centre of the wheel. Slip ratio is used to quantify slip in a relative form.

As stated in [17], the slip ratio can be calculated using (Eq. 44) for a smooth wheel.

$$s = \begin{cases} \dfrac{r\omega - v}{r\omega}, & r\omega \geq v, \quad 0 \leq s \leq 1 \\ \dfrac{r\omega - v}{v}, & r\omega < v, \quad -1 \leq s < 0 \end{cases} \qquad \text{(Eq. 44)}$$

Depending on the value of s, there is slippage ($s > 0$), the wheel rolls without slipping or skidding ($s = 0$) or it skids ($s < 0$). The corresponding velocity profiles are shown in Figure 6.

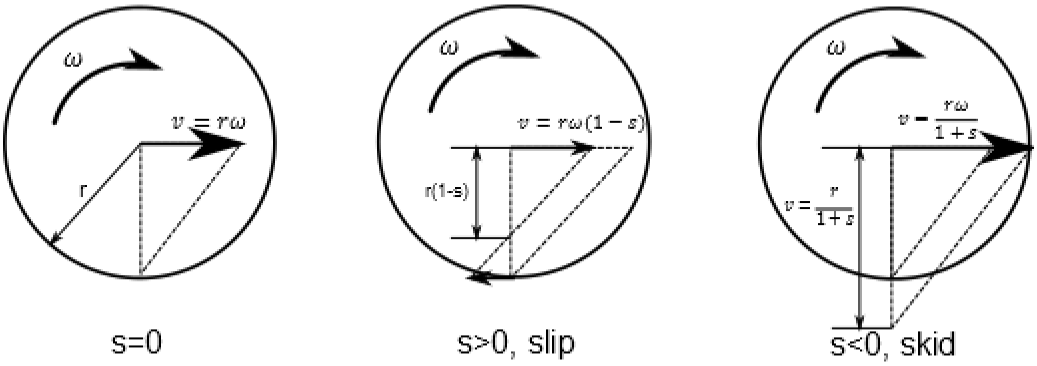

Figure 6. Instantaneous velocity for wheels according to their slip ratio.

3.3. Required torque and power

A torque or momentum is the ability of a force to rotate a mechanical system around a given point. The intensity of the torque depends on the orthogonal applied force ($F_{\perp}$), the length of the lever arm (r) and the angle between them (θ), as shown in Figure 7.

In Figure 8, a sketch of the angular momentum is given. The angular momentum describes the state of rotation of a physical system. It depends on the body's rotational inertia (I) and the rotational velocity around a particular axis (ω).

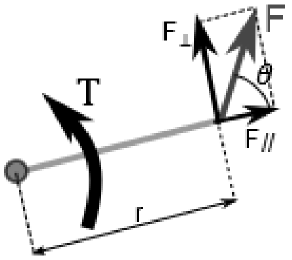

Figure 7. Torque produced by a force.

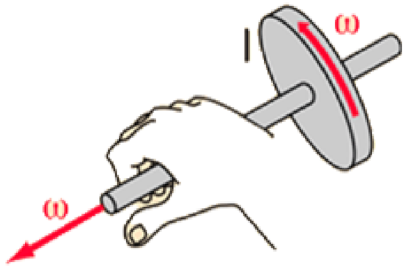

Figure 8. Angular momentum.

Torque or momentum: $$\mathrm{T} = r\wedge F = rFsin\theta = F_{\perp} \quad \text{(Eq. 45)}$$

Angular momentum: $$\mathrm{L} = \mathrm{I}\omega \quad \text{(Eq. 46)}$$

Once the involved forces and parameters are computed, the calculation of the torque to be applied to the wheel is almost trivial. Then, the appropriate conversion has to be done in order to obtain the torque to be delivered by the motor.

The nominal torque of a motor is the torque that it can supply continuously without failing suddenly. Every motor has its own characteristics curve. The manufacturer of a motor also provides design parameters, like nominal and maximum torque and velocity, nominal power or rotor moment of inertia. Consequently, the velocity and torque that can be provided by the motor are limited, which should be taken in consideration when planning the movement of the wheel.

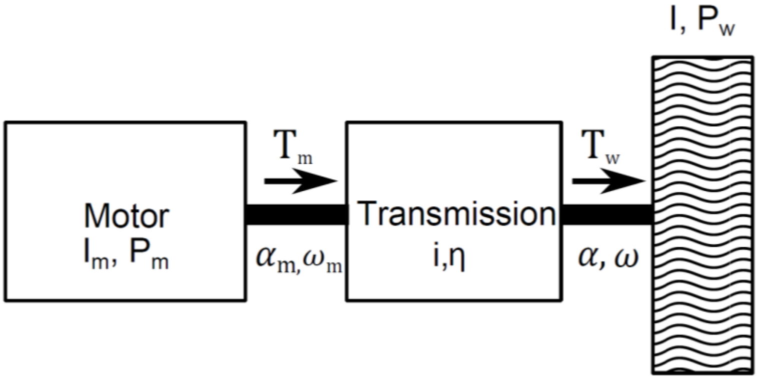

Figure 9. Torques and velocities on the transmission chain

Besides torque, it is necessary to calculate the required power that should be delivered to the system. The calculation will be done by applying a power balance (Eq. 47), which is schematized in Figure 9.

$$P_m = \mathrm{T}_m\omega_m; \quad P_w = \mathrm{T}_w\omega \quad \text{(Eq. 47)}$$

$$\omega = \frac{\omega_m}{i} \quad \text{(Eq. 48)}$$

$$P_m = P_w + (1-\eta)P_m + \frac{dE_c}{dt} \quad \text{(Eq. 49)}$$

Where P_m and P_w are the power delivered by the motor and the power used by the wheel, the term $(1-\eta)P_m$ refers to the power lost due to passive resistances and the remaining term represents the power accumulated by the motor.

3.4. Proposed model

As proved in [12], the addition of a diameter-dependant factor in Bekker's sinkage equation (Eq. 32) improves the performance of the Bekker's model for small wheels, which is the case of most of the terrestrial robots. Consequently, this is the formulation that will be used for the calculation of sinkage. Figure 10 shows the distribution of forces and stress on a moving wheel.

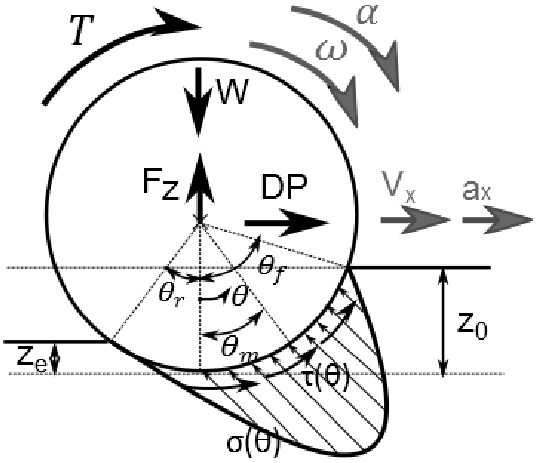

Figure 10. Distribution of forces and stress on a moving wheel.

Considering that $\hat{k}$, $\hat{n}$, and $\hat{m}$ are fitting constants that depend on the soil type and have to be determined empirically, and being D and b the diameter and width of the wheel and W the vertical load on the wheel, static sinkage is obtained using (Eq. 50).

$$z_0 = \frac{3W}{b(3-\hat{n})\hat{k}D^{\hat{m}+0.5}}^{\frac{2}{2\hat{n}+1}} \qquad \text{(Eq. 50)}$$

If the wheel is moving, the possibility of slippage between the wheel and the soil has to be considered. In this situation, dynamic sinkage applies (Eq. 51). Parameter s is the wheel slip; r, the radius; ω, the angular speed and V_x, the speed of the vehicle.

$$z_d = \frac{1+s}{1-0.5s} z_0 \qquad \text{(Eq. 51)}$$

According to [1], the wheel slip has different values during breaking or during traction.

$$s = \begin{cases} \dfrac{\omega r - V_x}{V_x}, during\ breaking \\ \dfrac{\omega r - V_x}{\omega r}, during\ traction \end{cases} \qquad \text{(Eq. 52)}$$

Then, according to the model proposed in [15], the normal stress can be computed using (Eq. 55). The angles θ_m, θ_r and θ_f are the ones shown in Figure 10, θ_s is the static wheel-soil contact angle. Since the wheel is moving, a slightly different approach will be used for the calculus of θ_f, depending on z_d instead of z_0 as it was used in [16].

$$\sigma(\theta) = \begin{cases} \hat{k} r^{\hat{n}} (cos\theta - cos\theta_s)^{\hat{n}} d^{\hat{m}},\ for\ \theta_m \leq \theta \leq \theta_f \\ \hat{k} d^{\hat{m}} r^{\hat{n}} \left[cos\left(\theta_f - \frac{(\theta-\theta_r)\,(\theta_f-\theta_m)}{(\theta_m-\theta_r)}\right) - cos\theta_f \right]^{\hat{n}},\ for\ \theta_r \leq \theta \leq \theta_m \end{cases} \quad \text{(Eq. 53)}$$

$$\theta_m = \frac{\theta_r + \theta_f}{2} \quad \text{(Eq. 54)}$$

$$\theta_r = arccos\left(1 - \frac{z_e}{R}\right) \quad \text{(Eq. 55)}$$

$$\theta_f = arccos\left(1 - \frac{z_d}{R}\right) \quad \text{(Eq. 56)}$$

Given c, the cohesion modulus of the soil; ϕ, the friction angle; $j(\theta)$, the shear displacement along the wheel-soil interface [18], and κ, a shear deformation parameter, the longitudinal shear stress is:

$$\tau(\theta) = (c + \sigma(\theta) tan\phi)\left(1 - exp^{\frac{-j(\theta)}{\kappa}}\right) \quad \text{(Eq. 57)}$$

$$j(\theta) = r\left(\left(\theta_f - \theta\right) - (1 - s)\left(sin\ \theta_f - sin\ \theta\right)\right) \quad \text{(Eq. 58)}$$

Finally, applying Newton's second law, the drawbar pull (DP), the vertical force (F_z) and the required torque (T) can be obtained from (Eq. 59)(Eq. 61). I and m are the inertia and mass of the body, $a_x$and a_y are the acceleration values related to the wheel in the horizontal and vertical axes and α is the angular acceleration.

$$DP + Rb \int_{\theta_r}^{\theta_f} \{\tau(\theta) cos\theta - \sigma(\theta) sin\theta\} d\theta = m \cdot a_x \quad \text{(Eq. 59)}$$

$$bR \int_{\theta_r}^{\theta_f} [\tau(\theta)\sin(\theta) + \sigma(\theta)\cos(\theta)] d\theta - F_z = m \cdot a_y \quad \text{(Eq. 60)}$$

$$-bR^2 \int_{\theta_r}^{\theta_f} \tau(\theta) d\theta + T = I \cdot \alpha \quad \text{(Eq. 61)}$$

For calculation purposes, a_y will be considered negligible. For the calculus of the inertia of the wheel it will be assumed that the wheel is treatable as a cylinder. Then, (Eq. 62) holds.

$$I = \frac{1}{2} m r^2 \quad \text{(Eq. 62)}$$

4. WHEELED TERRESTRIAL ROBOTS

Unicycle robots are one of the simplest versions of terrestrial robots [19]. They consist of two independent driving wheels and, on occasion, some extra wheels to ensure stability. Their rotation centre is on the axis of the driving wheels. Due to that configuration, this kind of robots cannot move perpendicularly to their wheels. However, they can rotate over themselves. Some examples of unicycle robots are shown in Figure 11: PEA Bot [20], QB robot [21], Tibi and Dabo [22]

Figure 11.Some examples of unicycle robots

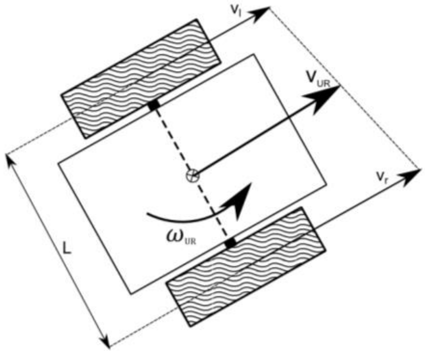

Figure 12. Unicycle robot.

Figure 12 illustrates the main parameters involved in the movement of a unicycle robot. Assuming non-sliding conditions, the velocity of the robot can be computed as shown in (Eq. 63).

$$v_{UR} = \frac{v_l + v_r}{2} = \frac{r(\dot{\phi}_l - \dot{\phi}_r)}{2} \qquad \text{(Eq. 63)}$$

where $\dot{\phi}_l$, $\dot{\phi}_r$ are the angular rotation and v_l, v_r, the linear velocity of the left and right wheels. L is the distance between driving wheels and r, the wheels' radius.

In order to calculate the rotation of the robot, it is necessary to determine the instant centre or instantaneous centre of zero velocity (IC). The IC on an imaginary axis of zero velocity, about which the body appears to rotate at a given instant. This axis is always perpendicular to the plane of motion. The graphical way to determine it is shown in Figure 13.

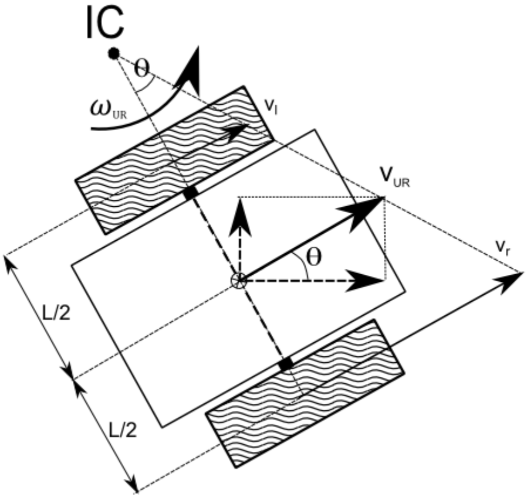

Figure 13. Example of the graphical obtention of the IC for a unicycle robot

If the velocity of the two wheels is different, the robot will rotate around the IC. Otherwise, the IC will be placed at an infinite distance, thus, the angular velocity of the robot will be null. Applying the similar triangles theorem, the rotation around the IC can be calculated using (Eq. 64).

$$\omega_{UR} = \frac{v_r - v_l}{L} = -\frac{r(\dot{\phi}_l + \dot{\phi}_r)}{L} \qquad \text{(Eq. 64)}$$

Assuming no sliding on the direction of the axis of the wheels, the position (x,y) can be obtained, and thus the position of the robot at every moment, as suggested in [19].

$$\dot{\theta} = \omega \qquad \text{(Eq. 65)}$$

$$\dot{x} = v_{UR} cos\theta \qquad \text{(Eq. 66)}$$

$$\dot{y} = v_{UR} sin\theta \qquad \text{(Eq. 67)}$$

$$\theta(t) = \int_0^t \dot{\theta}(\tau)d\tau + \theta_0 = \int_0^t \omega(\tau)d\tau + \theta_0 \qquad \text{(Eq. 68)}$$

$$x = \int_0^t \dot{x}(\tau)d\tau + x_0 = \int_0^t v(\tau)cos\theta d\tau + x_0 \qquad \text{(Eq. 69)}$$

$$y = \int_0^t \dot{y}(\tau)d\tau + y_0 = \int_0^t v(\tau)sin\theta d\tau + y_0 \qquad \text{(Eq. 70)}$$

Being $v(\tau)$ the desired velocity profile for the whole robot, $\dot{x}(\tau)$, $\dot{y}(\tau)$ the velocity on the x and y directions, respectively, x_0, y_0, θ_0 the initial values for x, y and θ.

5. PROGRAMS

The programs proposed on this chapter have been implemented using the Python programming language.

Two libraries have been used in order to program some mathematical issues, such as trigonometric functions or solver programs: SymPy [23] and NumPy [24]. SymPy is a Python library for symbolic mathematics. It is written entirely in Python and does not require any external libraries. NumPy is an extension to the Python language that includes a large library of high-level mathematical functions. Besides, another library, Matplotlib [25] has been used for graph generation.

Nevertheless, the algorithms are briefly explained in plain words, so that someone not familiar with Python can easily understand the code. Moreover, flow diagrams for each algorithm are provided in Appendix C.

A general idea of the relationship between the programs of this chapter and what they do is given in Figure 14 and Figure 15.

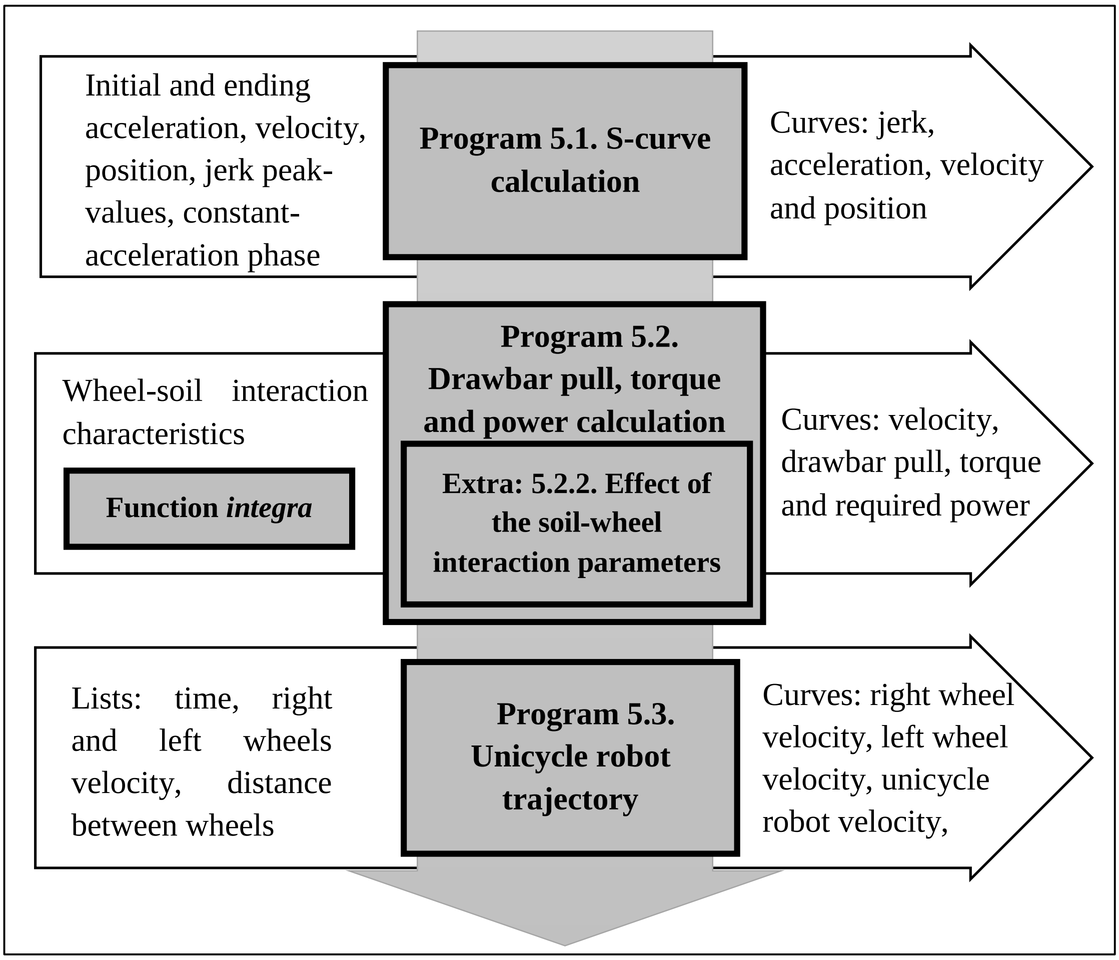
Initial and ending acceleration, velocity, position, jerk peak-values, constant-acceleration phase
Program 5.1. S-curve calculation
Curves: jerk, acceleration, velocity and position
Wheel-soil interaction characteristics
Function *integra*
Program 5.2. Drawbar pull, torque and power calculation
Extra: 5.2.2. Effect of the soil-wheel interaction parameters
Curves: velocity, drawbar pull, torque and required power
Lists: time, right and left wheels velocity, distance between wheels
Program 5.3. Unicycle robot trajectory
Curves: right wheel velocity, left wheel velocity, unicycle robot velocity,

Figure 14. General relationship between the proposed programs.

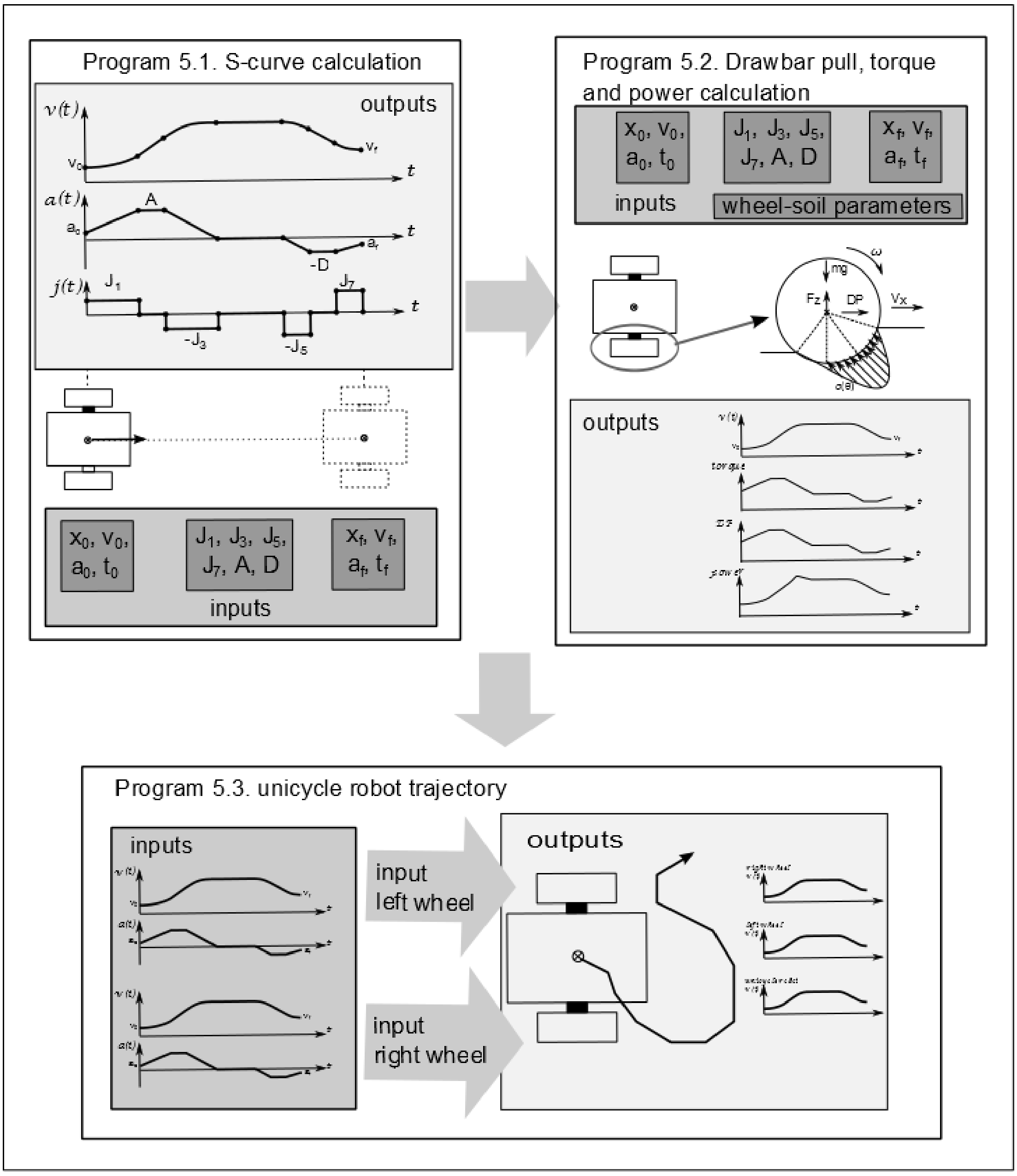

Figure 15. Visual diagram of the programs.

5.1. S-curve calculation

Input arguments: $a_0, a_f, v_0, v_f, x_0, x_f, A, D, J_1, J_3, J_5, J_7$

Output arguments: $t = [t_1, t_2, t_3, t_4, t_5, t_6,\ t_f]$

$x = [x_0, x_1, x_2, x_3, x_4, x_5, x_6, x_f]$

$v = [v_0, v_1, v_2, V, V, v_5, v_6, v_f]$

$a = [a_0, v_1, v_2, V, V, v_5, v_6, v_f]$

$j = [J_1, 0, -J_3, 0, -J_5, 0, J_7, 0]$

Jerk, acceleration, velocity and position curves

This program calculates and shows the jerk, acceleration, velocity and position profiles given their starting and desired final values and the time in which those have to be reached. In order to do this calculation, the jerk, acceleration and deceleration peak values are given.

With these inputs, the system of equations formed by the restrictions described in 2.2.2, proposed algorithm, is solved, thanks to the *solvers* module included in Sympy library. The system consists of 24 restrictions and 24 unknown variables. The points to be plotted are calculated one by one, stored in lists, and, finally, plotted. Regarding the code, both symmetrical and asymmetrical s-curves can be computed. Some examples are shown in Appendix C.1.1. Execution examples

Comments on the results

This program is supposed to be the first calculus to be done when planning the motion of one wheel of a robot when the starting and ending kinematic values and time are set. The curves should be computed for each of the wheels and taken as a reference to proceed with the rest of the planning, this is, how to obtain these curves as a result when the soil properties vary and what power should be provided to the engines that move the wheels.

Executing the program with different input values, some light can be shed on the effects of each parameter. The difference between the plots obtained from Example 1 and Example 2 (C.1.1. Execution examples) is only the value of jerks J_5 and J_7. Thus, the effect of jerk in the curves can be observed: the lower the peak values for jerk, the smoother the correspondent velocity profile and so the movement of the robot.

Moreover, a relation between the value of the jerk and the duration of the phases should be pointed. As shown in Figure 16, the lower the jerk value, the lower the duration of phases 2 (linear change of velocity) and 4 (constant velocity phase) and the higher the duration of phases 1 and 3. The value of the jerk value slightly affects to the velocity in the constant velocity phase: for a decrement of the jerk value from 4 to 0,3 m/s^3 (92,5%), the velocity at the constant velocity phase, which corresponds to the maximum achieved velocity in the studied profile, should be increased in a 13,67%.

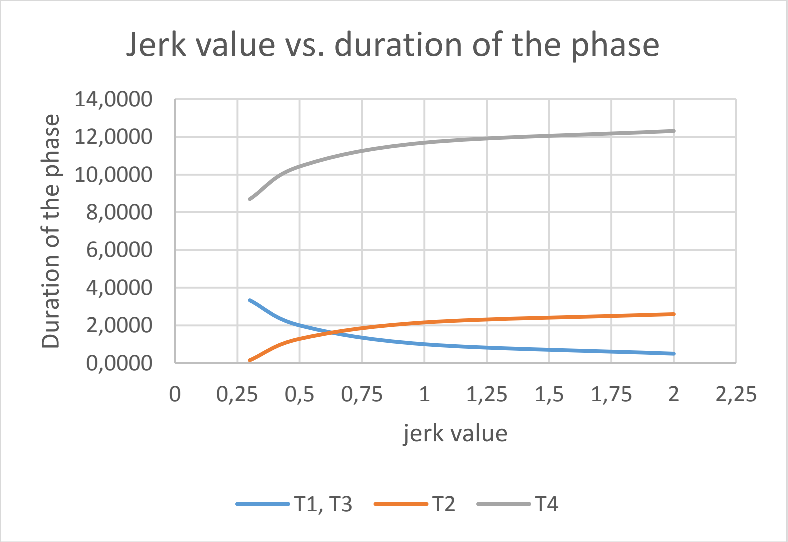

Figure 16. Peak value for Jerk (J_3, J_5) vs. duration of correspondent phases

Given these effects and considering a case where there is a need for a smooth movement, and considering only the kinematic parameters, it would be recommendable to set a target profile with the minimum possible jerk values.

5.2. Drawbar pull, torque and power calculation

Input arguments: $a_0, a_f, v_0, v_f, x_0, x_f, A, D, J_1, J_3, J_5, , J_7$
$coefk, coefn, coefm, c, psi, K, s, d, r, m, W, b, thetas, ze$
Output arguments: Curves: velocity, drawbar pull, required torque and required power

This program computes the equations from the model proposed in 3.4. Proposed modelThe aim is to obtain the drawbar pull, torque and power profiles required by the wheel to move following the desired velocity profile, which is calculated on the first stage of the program using the same code as in 5.1. A flow diagram of the program can be found in C.2.2.

Some terrain and wheel-soil input variables have been introduced with a different name: coefk ($\hat{k}$), coefn ($\hat{n}$), coefm ($\widehat{m}$), c, psi (ϕ), K (κ), thetas (θ_s), ze (z_e). The values used in [15] are taken as reference for these parameters. In order to use the program in a real case, some experiments should be conducted to find the values for the terrain in which the robot has to move. The slip value (s) is calculated using, from (Eq. 52), the line applicable during traction, since it is considered that the wheel is in motion in the case of study.

With the aim of plotting the drawbar pull, torque and required power, these are calculated at each study point and stored in lists, using the same method as in the program in 5.1.

One might notice that the calculus involved is quite complex for a Python program, since there are many variables and there is a need to use integrals. Although SymPy includes an integrating module, it does not work properly when a calculus using NumPy is involved in the function that has to be integrated. Therefore, a simple integrating function has been introduced in the program (see 5.2.1.). Besides, because the equations are long and involve many parameters and trigonometric functions, they have been programmed in independent functions in order to clarify the comprehension of the code.

5.2.1. Function integra

The function *integra* receives as inputs the function to be integrated, the variable, and the values A and B between which the function has to be integrated. The calculus is done using the mid-point method, which approximates the area under the curve in an interval using (Eq. 71). In order to obtain a precise result, the interval (A, B) is divided into smaller parts, being the result of the integration the sum of the areas of all of these.

$$\int_a^b f(x)dx = (b-a)\cdot f\left(\frac{a-b}{2}\right) \qquad \text{(Eq. 71)}$$

The higher the number of small intervals, the more precise the integration is, but also slower. To decide this parameter, an extra program (see C.2.3.) has been used. In it, a simple version of the calculus of the drawbar pull is implemented using the *integra* function. The integration interval is divided from 1 to 4,000 parts and afterwards, the results are plotted and printed. The execution of the program is slow, but speed is not its target.

To obtain a more precise program, a study of the error made (Eq. 72) by using the mid-point approximation is conducted taking as a reference value the result (*Real value*) of the integration obtained dividing the interval in 10,000 parts.

$$error = \left|\frac{Approximated\ value - Real\ value}{Real\ value}\right| 100 \qquad \text{(Eq. 72)}$$

Figure 17 shows the output graph, from which the conclusion that the output starts to converge when the interval is split in 500 parts or more can be reached.

A second execution of the program allows better specification of the number of divisions that should be used. In this case, the starting value is 400. The output graph (Figure 18) shows that the error diminishes around 900 parts of the interval and then oscillates. Nevertheless, given that the value of this error is of the order of 10^{-4}%, the decision is to take 1000 divisions in the execution of the program.

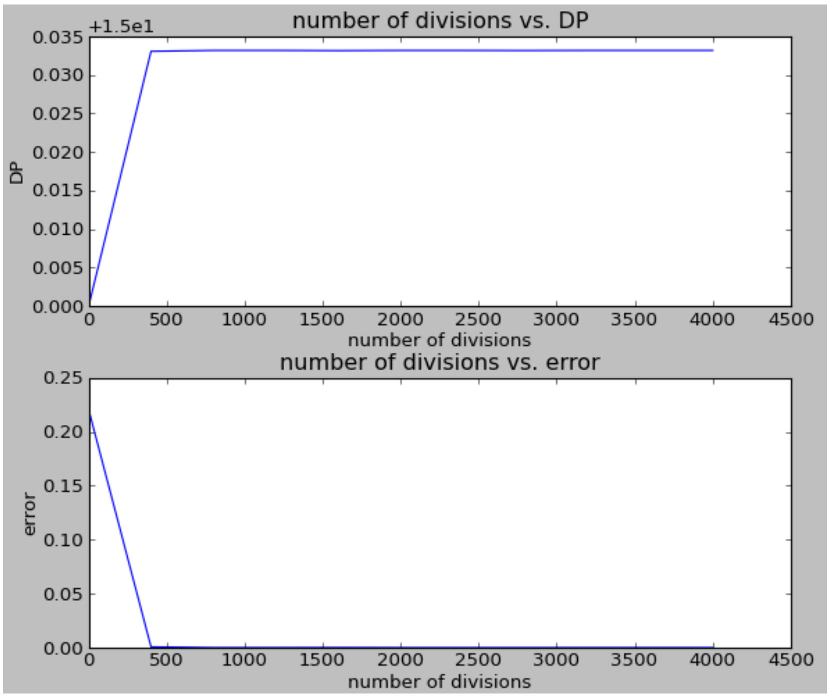

Figure 17. Output from program C.2.3. starting at 1 part division of the integration interval.

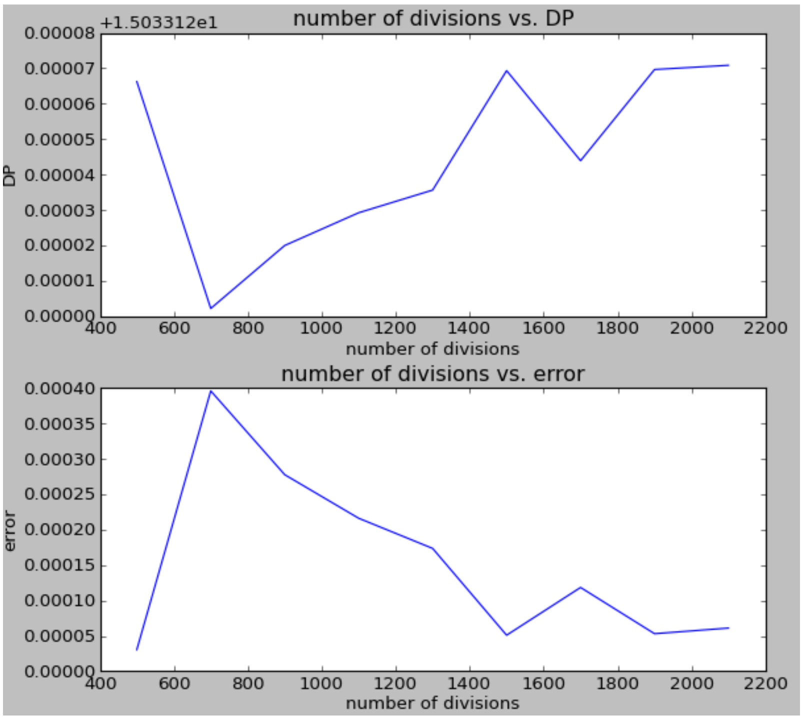

Figure 18. Output from program C.2.3. starting at 400 parts division of the integration interval.

5.2.2. Effect of the soil-wheel interaction parameters

With the aim of showing how each wheel and wheel-soil interaction parameter affects the DP and the required torque and power, some modifications on the program must be made. The programs calculate these values as in C.2.3 for a range of values of the parameter under study, store its input value and the output DP, torque and power in lists and, finally, plot the results.

One program has been implemented for each of the parameters, although they are all almost the same. The output graphs can be consulted in C.2.4.

5.2.3. Comments on the results

As stated before, the target of this program is to obtain the torque, DP and power required by one wheel to move following a specific velocity profile. This profile is calculated based on the desired starting and ending kinematic characteristics on the first stage of the program.

The effect of the soil and the dimensions of the wheels are under study in order to define how they affect these profiles, so this information can be taken into account when planning the motion of a robot -or even when designing the robot itself- with the aim of minimizing the power that it will require to move.

The execution examples in C.2.1. show the curves obtained by the program with different input values, while the execution of the program to calculate the effect of each parameter by itself can be consulted in C.2.4. Hereafter, the referenced figures can be found in annex C.2. under these two points.

Comparing Figure 33 with Figure 35, it becomes clear that higher slip values result in the need of supplying more DP and torque to the wheel, thus more power is needed to follow the desired velocity profile. This relation is further clarified in Figure 40.

Enlarging the wheels' dimensions, seems to increase the required torque, while diminishing the needed drawbar pull and power (see Figure 36). Figure 41 refutes this apparent relationship. For small diameter values, the DP and required torque do decrease. Nevertheless, starting from a certain diameter value (in this case around 0.5m), the required power increases, while the DP keeps reducing.

Still regarding to the wheel, a rise of the load on it results in higher DP, torque and power. This effect is exemplified in Figure 37 by an increase of the mass with respect to the one in Figure 33.

To study the influence of the type of soil on the performance of the wheel, the effect of the cohesion of the soil (parameter c) and parameter $\hat{k}$ have been analysed. From Figure 42, the conclusion that can be taken is that the more compact is the soil (thus, the lower the value of c), the higher is the needed DP and the lower the required torque and power are. Regarding $\hat{k}$, there is a certain interval of values for which higher $\hat{k}$ implies higher performance, but starting from a certain value, the required DP, torque and power decrease until they tend to a fixed value.

5.3. Unicycle robot trajectory

Input arguments: $vr[\,], vl[\,], t[\,], L$

Output arguments: Curves: right wheel velocity, left wheel velocity, unicycle robot velocity, trajectory in x-y plane

Starting from the velocity profiles for the right and left wheels of a unicycle robot, and given the distance between them (L), this program calculates the consequent (x, y) values in order to represent the trajectory followed by the robot in the x-y plane, using the equations from Chapter 3. This program should be taken as a starting point to further implement programs to plan the motion of the robot.

The input profiles are given as lists of values. Consequently, there is no need to integrate, since the expressions in (Eq. 68) - (Eq. 70) do not depend on τ. Anyhow, the equations have been respected so that it is easy to adapt the program to another kind of input profiles.

Some execution examples and the flow diagram of the program can be found in Appendix C.3.

Comments on the results

This program sheds some light on how the profile of each wheel of a unicycle robot affects the movement of the whole vehicle: when the right wheel is faster than the left one, the robot turns left (see execution examples in Appendix C.3.), when both wheels roll at the same velocity, it goes straight.

Figure 44 and Figure 45 are the outputs of the program for the same input profiles, but with different L-values, which means that the distance between the two wheels is different. It becomes clear from the figures that this parameter has great effect on the motion of the robot. It is necessary to point out that L only affects the motion of the robot when the two wheels roll at different velocities.

6. DISCUSSION

The main objective of this study was to obtain and program a way to plan the motion of a robot by defining an appropriate velocity profile and calculate the power, drawbar pull and torque profiles required to accomplish it, exemplifying the motion planning with the study of a two-wheeled robot.

In the past, trapezoidal velocity profiles have been widely used to control engines. Nowadays, the evolution of robots and their uses has led to the need of using smoother profiles, due to the demand of high precision and delicate movements. It has been shown that this can be achieved by minimizing the change of acceleration (jerk), and using profiles with the shape of an *S*, known as s-curves.

A model to obtain s-curve velocity profiles has been introduced and programmed, mostly based on the equations proposed in [2]. The main differences between the two algorithms is that the one proposed in this thesis aims to be more general, by considering the possibility that there is acceleration at the starting and ending states. Moreover, it allows the modelling of both symmetrical and asymmetrical profiles in a similar way that in [4], while there is no need to iterate in order to calculate the profiles, as it had to be done in [2].

On the other side, the proposed model needs the values of acceleration at its constant-value phases and jerk as an input. Consequently, it could be taken as a starting point to create a program optimizing these values regarding dynamic restrictions. For example, maximum jerk, maximum achievable acceleration or minimum power to be supplied. It could also be used as a reference to obtain even smoother profiles, by creating a new model with an s-curve acceleration profile.

To provide a good control of the movement of a robot, it is necessary to ensure that it will meet the desired velocity profile. For this reason, a way to prevent how the wheels will react on the soil becomes highly useful, in order to adapt the supplied torque and, therefore, control the speed of the vehicle.

The study of the interaction between wheels and soil is called terramechanics and, even though it has been widely investigated, most of it is still based on the studies carried by Bekker on 1969 [10] and Wong and Reece [11]. Besides, the used wheel-soil interaction models are obtained empirically. The wheel-soil interaction equations in this thesis are mostly based in [15], which slightly modifies these models to better represent the real behaviour of small vehicles.

Starting from the wheel-soil interaction equations and the desired velocity and acceleration profiles calculated in the first proposed model of this thesis, a model to obtain the required drawbar pull, torque and power has been proposed.

Many studies have been done about the effect of the parameters of the models of the soil on the behaviour of the wheels, such as [17] or [18], mainly focusing on the effect of the slip value. Nevertheless, up to the date, the author of this thesis has not been able to find any that included the calculation of the required drawbar pull, torque and power profiles so that a robot moves following a desired velocity profile. Moreover, most of the papers focused on this subject consider static and uniform velocity conditions, but not accelerated movements, which have been considered in the proposed model.

On the implementation of the resulting algorithm in a program, some modifications have been made in order to obtain the effect of the different soil parameters on the outputs and compare them with the ones in different articles and papers on the subject. It turned out that each study outcomes different curves, so none of them could be taken as a reference for the comparison. Therefore, some empirical experiments should be held in order to verify the model, as well as to define the values of the soil constants of the model in different soil types.

Briefly, the proposed model includes the effect of the soil in the movement of a wheel. Even though, to decide what should the movement of the wheel be at each instant, there can always be unpredictable events that can lead to poor position estimation, excessive sinkage or loose of traction. Therefore, to reduce the effect of propagating the error during rover traverses, a means to measure wheel sinkage and slippage would be recommendable, so the calculated drawbar pull, torque and power profiles could be actualized at every instant. In conclusion, the model should be taken as a reference for the motion planning of a robot, but complemented with a control loop.

Besides the validation and adaption of the wheel-soil interaction equations and parameters by means of empirical experiments, the model could be improved by adding the case where the wheel starts moving from static conditions. In that case, the relation between slip and velocity is different and should be studied separately.

Finally, the thesis has been focused on unicycle or two-wheeled robots with the aim of showing how the models could be implemented to control a real robot. A program to obtain the trajectory of the robot by getting as inputs the velocity profile for each wheel has been implemented.

In further studies, it would be useful to implement the opposite program: a program that calculates the velocity profile for each wheel starting from the velocity and trajectory that the robot should follow. Moreover, some experiments should be held with different two-wheeled robots in order to validate the model.

7. SUMMARY

This project aimed to obtain and program a model to plan the movement of a robot in terms of velocity, torque, drawbar pull and power, regarding its starting and ending kinematic states and taking into consideration the reaction between the wheels and the soil.

First, a model to define a general s-curve velocity profile has been proposed. Taking this model as a starting point, the interaction between wheel and soil is added in order to predict the drawbar pull, torque and power required by the wheel to move following the input velocity profile. Finally, a two-wheel system is studied in order to show how the models could be adapted into a mobile robot. The resulting models are programmed in Python language as an example of how they could be implemented.

The proposed s-curve model allows the generation of both symmetrical and asymmetrical profiles without need of an iterating process. The second model considers the effect of the soil in a moving wheel and the previously calculated velocity and acceleration profiles in order to obtain the required drawbar pull, torque and power profiles so that a robot can move following the desired velocity profile.

At this stage of the research, the models proposed in this thesis should be taken as a starting point to investigate more deeply the effect of different types of soil on the motion of a terrestrial robot. The aim should be to obtain a way to predict how the robot should behave at every instant of its trajectory while it calculates the characteristics of the soil as it moves and recalculates the curves.

8. REFERENCES

[1] F. Djohor, N. M'Sirdi and A. Naamane, "Wheel-Soil Interaction Model for All-Terrain Vehicles," in *2012 2nd International Conference on Communications, Computing and Control Applications (CCCA)*, Marseille, 2012.

[2] K. Erkorkmaz and Y. Altintas, "High speed CNC systems design. Part 1: Jerk limited trajectory generation and quintic spline interpolation," *International Journal os Machine Tools and Manufacture, vol. 41,* pp. 1323-1345, 2001.

[3] K. Doang Nguyen, I.-M. Chen and T.-C. Ng, "Planning Algorithms for S-curve Trajectories," in *IEE/ASME international Conference on Advanced Intelligent Mechatronics, pp. 1-6,* Zurich, 2007.

[4] C.-W. Ha, K.-H. Rew and K.-S. Kim, "A Complete Solution to Asymmetric S-curve Motion Profile: Theory & Experiments," in *International Conference on Control, Automation and Systems*, Seoul, 2008.

[5] F. Robillard, "Reaction of a wheel on the ground and involved parameters," Final Internship Report, University of Oulu, Oulu, 2012.

[6] S. Yong Jeong, Y. Jong Choi, R. Park and S. Gap Choi, "Jerk limited velocity profile generation for high speed industrial robot trajectories," in *16th IFAC World Congress*, Prague, 2005.

[7] J. Y. Wong, "Some thoughts on simulations in terramechanics," xTerramechanics -Integrated Simulation of Planetary Surface Missions", at the Keck Institute for Space Studies,California Institute of Technology, 1-3 August 2011. [Online]. Available: http://www.kiss.caltech.edu/workshops/xterramechanics2011b/presentations/wong.pdf. [Accessed 3 April 2013].

[8] G. Reina, "Methods for Wheel Slip and Sinkage Estimation in mobile Robots, Robot localization and Map Building," 2010. [Online]. Available: http://cdn.intechweb.org/pdfs/10589.pdf. [Accessed 8 April 2013].

[9] R. Irani and R. W. A. Bauer, "A dynamic terramechanic model for small lightweight vehicles with rigid wheels and grousers operating in sandy soil," *Journal of terramechanics,* no. 48, pp. 307-318, 2011.

[10] M. G. Bekker, Introduction to Terrain-Vehicle Systems, Ann Arbor: The University of Michigan Press, 1969, p. 846.

[11] J. Wong and A. Reece, "Prediction of rigid wheel performance based on the analysis of soil-wheel stresses. Part I: Performance of driven rigid wheels," *Journal of Terramechanics,* no. 4, pp. 81-98, 1967.

[12] G. Meiron-Griffith and M. Spenko, "An Empyrical Study of the Terramechanics of Small Unmanned Ground Vehicles," in *IEEE Aerospace Conference,* big Sky, 2010.

[13] L. Ding, H. Gao, D. Zongquan, K. Nagatani and K. Yoshida, "Experimental

study and analysis on driving wheel's performance for planetary exploration rovers moving in deformable soil," *Journal of Terramechanics*, no. 48, pp. 27-45, 2011.

[14] M. Grahn, "Prediction of sinkage and rolling resistance for off-the-road vehicles considering penetration velocity," *Journal of Terramechanics,* vol. 28, pp. 339-47, 1991.

[15] G. Meiron-Griffith and M. Spenko, "Application of a Diameter-Dependent Terramechanics Model to Small-Wheeled Unmanned Ground Vehicles Operating on Deformable Terrain," in *IIEE/RSJ International Conference on Intelligent Robots and Systems*, San Francisco, 2011.

[16] I. Shmulevich, U. Mussel and D. Wolf, "The effect of velocity on rigid wheel performance," *Journal of Terramechanics,* vol. 35, pp. 189-207, 1998.

[17] L. Ding, G. Haibo, Z. Deng, K. yoshida and K. Nagatani, "Slip ratio for lugged wheel of planetary rover in deformable soil: definition and estimation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. louis, 2009.

[18] O. A. Ani, H. Xu, Y.-P. Shen, S.-G. Liu and K. Xue, "Modeling and multiobjective optimization of traction performance for autonomous wheeled mobile robot in rough terrain," *Journal of Zhejiang University-SCIENCE C (Computers & Electronics),* vol. 14, no. 1, pp. 11-29, 2013.

[19] S. Lens, Locomotion d'un robot mobile, mémorie présenté en vue de l'obtention du grade d'ingénieur Civil Informaticien, Liège: Université de Liège, 2008.

[20] R. Needleman, "C-net.com," 5 1 2006. [Online]. Available: http://www.cnet.com/4831-11405_1-6412895-1.html. [Accessed 22 5 2013].

[21] B. Lohmann, M. Mini and L. Trazzi, "designboom," Designboom, [Online]. Available: http://www.designboom.com/technology/anybots-qb-telepresence-robot/. [Accessed 22 5 2013].

[22] "Tuvie-futuristic technology," [Online]. Available: http://www.tuvie.com/tibi-dabo-robots-for-public-spaces/. [Accessed 22 5 2013].

[23] S. D. Team, "SymPy.org," SymPy, [Online]. Available: http://sympy.org/en/index.html. [Accessed February-May 2013].

[24] "NumPy.org," NumPy, [Online]. Available: http://www.numpy.org/. [Accessed February-May 2013].

[25] J. D. Hunter, "Matplotlib: A 2D graphics environment," Matplotlib, [Online]. Available: http://matplotlib.org/. [Accessed February-June 2013].

[26] M. Lyasko, "Slip sinkage effect in soil-vehicle mechanics," *Journal of terramechanics,* vol. 47, pp. 21-31, 2010.

[27] I. C. Jong and B. G. Rogers, Engineering mechanics: Dynamics, United States of America: Saunders College Publishing, 1991.

[28] V. M. Arevalo, "Sinusoidal Velocity Profiles for Motion Control", report. Augen Opticos, S.A., México.

[29] T. H. Tran, N. M. kwok, S. Scheding and Q. P. Ha, "Dynamic Modelling of Wheel-Terrain Interaction of a UGV," in *IEEE Conference on Automation Science and Engineering*, Scottsdale, 2007.

[30] K. Gu and Y. Wei, "Dynamic Modeling and Sliding Mode Driving Control for Lunar Rover Slip," in *IEEE International Conference on Integration Technology*, Shenzhen, 2007.

[31] J. Iyer, M. Chapariha, M. Gougani, J. Jatskevich and A. Davoudi, "Torque sharing between V/F Controlled Vehicular Wheels under Slippery Ground Conditions," in *IEEE Transportation Electrification Conference and Expo (ITEC)*, 2012.

[32] J. A. Pytka, Dynamics of Wheel-Soil System, Boca Raton: CRC PRess, 2012.

[33] J. Otero Yugat, "Metodología de selección del servomotor y transmisión de un sistema de accionamiento mecánico por medio del análisis de su potencia transitoria," *Revista Técnica de Ingeniería Universidad Zulia*, vol. 33, no. 1, pp. 21-28, 2010.

[34] M. Rashidi and K. Seyfi, "Comparative studies on Bekker and Upadhyaya models for soil pressure-sinkage behaviour prediction," *American-Eurasian J. Agric. & Environ. Sci.*, vol. 3, no. 1, pp. 19-25, 2008.

[35] W. Petersen, *Volumetric contact model for planetary wheel/soil interaction*, Ontario: University of Waterloo, 2012.

[36] C. Guarino lo Bianco, "Optimal velocity planning for autonomous vehicles considering curvature constraints," in *IEEE International Conference on Robotics and Automation*, Rome, 2007.

[37] A. Alasty and H. Pendar, "Equations of Motion of a Single-Wheel Robot in a Rough Terrain," in *IEEE International Conference on Robotics and Automation*, Barcelona, 2005.

[38] "Hyperphysics," HyperPhysics, [Online]. Available: http://hyperphysics.phy-astr.gsu.edu/hbase/amom.html. [Accessed 5 april 2013].

[39] T. Boh, J. Billingsley, R. S. Bradbeer and P. Hodgson, "Terramechanics based traction control underwater wheeled robot," in *Oceans 2010 IEEE*, Sydney, 2010.

[40] D.-N. Kim, C. I-Ming and N. Teck-Chew, "Planning Algorithms for S-curve Trajectories," in *IEE/ASME international Conference on Advanced Intelligent Mechatronics, pp. 1-6*, Singapore, 2007.

[41] "hyperphysics," HyperPhysics , [Online]. Available: http://hyperphysics.phy-astr.gsu.edu/hbase/amom.html. [Accessed 5 april 2013].

9. APPENDICES

Appendix A. S-curve profiles: state of the art

A.1. Erkorkmaz and Altintas, 2001 [2]

Erkorkmaz and Altintas have proposed an algorithm using a fifth order resampling technique with jerk limited speed control to generate time-optimal trajectory, which provides continuous position, speed and acceleration profiles. This algorithm is meant to be used for speed profile only.

The procedure below imposes a smooth feed motion along the quantic spline toolpath by modulating the time duration between position reference points. As shown in Figure 19, the resulting acceleration profiles are trapezoidal, linear in regions 1, 3, 5 and 7 with prespecified slopes, this is, prespecified jerk values, and constant in regions 2 and 6; thus, jerk is zero in these regions. Displacement profiles are cubic for regions 1, 3, 5 and 7 and linear in 2, 4 and 6.

The proposed algorithm starts from the assumption that the different time values (t_0, …, t_7) are known, and so are the jerk profile and the initial conditions for displacement, feedrate and acceleration. The resulting profile is shown in Figure 19.

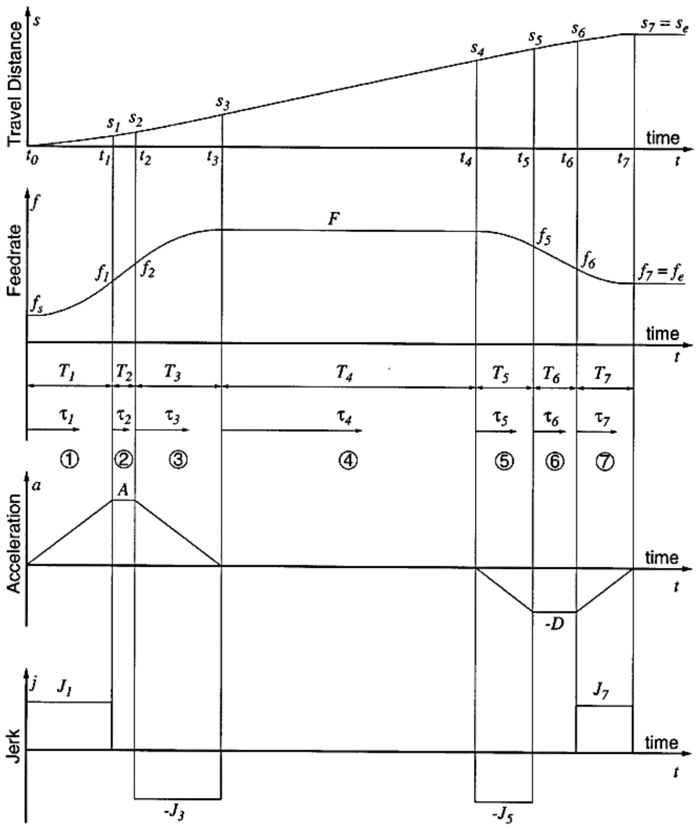

Figure 19. S-curve velocity / trapezoidal acceleration profile

A.1.1. Motion equations

$$j(\tau) = \begin{cases} J_1, & 0 \le t < t_1 \\ 0, & t_1 \le t < t_2 \\ -J_3, & t_2 \le t < t_3 \\ 0, & t_3 \le t < t_4 \\ -J_5, & t_4 \le t < t_5 \\ 0, & t_5 \le t < t_6 \\ J_7, & t_6 \le t < t_7 \end{cases} \qquad \text{(Eq. 73)}$$

$$a(\tau) = \begin{cases} J_1\tau_1, & 0 \le t < t_1 \\ A, & t_1 \le t < t_2 \\ A - J_3\tau_3, & t_2 \le t < t_3 \\ 0, & t_3 \le t < t_4 \\ -J_5\tau_5, & t_4 \le t < t_5 \\ -D, & t_5 \le t < t_6 \\ -D + J_7\tau_7, & t_6 \le t < t_7 \end{cases} \qquad \text{(Eq. 74)}$$

Where τ_k is the relative parameter that starts at the beginning of the kth phase.

$$f(\tau) = \begin{cases} f_s + \frac{1}{2}J_1{\tau_1}^2, \quad 0 \le t < t_1, \quad f_s\!: initial\ feedrate \\ f_1 + A\tau_2, \quad t_1 \le t < t_2, \quad f_1 = f_s + \frac{1}{2}J_1{T_1}^2 \\ f_2 + A\tau_3 - \frac{1}{2}J_3{\tau_3}^2, \quad t_2 \le t < t_3, \quad f_2 = f_1 + AT_2 \\ f_3, \quad t_3 \le t < t_4, \quad f_3 = f_2 + AT_3 - \frac{1}{2}J_3{T_3}^2 = F \\ f_4 - \frac{1}{2}J_5{\tau_5}^2, \quad t_4 \le t < t_5, \quad f_4 = f_3 \\ f_5 - D\tau_6, \quad t_5 \le t < t_6, \quad f_5 = f_4 - \frac{1}{2}J_5{T_5}^2 \\ f_6 - D\tau_7 + \frac{1}{2}J_7{\tau_7}^2, \quad t_6 \le t < t_7, \quad f_6 = f_5 - DT_6 \end{cases} \qquad \text{(Eq. 75)}$$

Where F is the feedrate to be achieved at the end of phase 3, f_k is the feedrate at the end of each phase and T_k is the duration of each phase.

$$x(\tau) = \begin{cases} f_s\tau_1 + \frac{1}{6}J_1{\tau_1}^3, 0 \le t < t_1 \\ x_1 + f_1\tau_2 + \frac{1}{2}A{\tau_2}^2,\ t_1 \le t < t_2, \quad x_1 = f_s\tau_1 + \frac{1}{6}J_1{T_1}^3 \\ x_2 + f_2\tau_3 + \frac{1}{2}A{\tau_3}^2 - \frac{1}{6}J_3{\tau_3}^3, t_2 \le t < t_3, \quad x_2 = x_1 + f_1T_2 + \frac{1}{2}A{T_2}^2 \\ x_3 + f_3\tau_4, \quad t_3 \le t < t_4, x_3 = x_2 + f_2T_3 + \frac{1}{2}A{T_3}^2 - \frac{1}{6}J_3{T_3}^3 \\ x_4 + f_4\tau_5 - \frac{1}{6}J_5{\tau_5}^3, t_4 \le t < t_5, \quad s_4 = s_3 + f_3T_4 \\ x_5 + f_5\tau_6 - \frac{1}{2}D{\tau_6}^2, t_5 \le t < t_6, \quad x_5 = x_4 + f_4T_5 - \frac{1}{6}J_5{T_5}^3 \\ x_6 + f_6\tau_7 - \frac{1}{2}D{\tau_7}^2 + \frac{1}{6}J_7{\tau_7}^3, t_6 \le t < t_7, \quad x_6 = x_5 + f_5T_6 - \frac{1}{2}D{T_6}^2 \end{cases} \qquad \text{(Eq. 76)}$$

Hence, the distance travelled during each phase can be expressed as:

$$l_k = \begin{cases} l_1 = x_1 + f_s T_1 + \frac{1}{6} J_1 {T_1}^3 \\ l_2 = x_2 - x_1 = f_1 T_2 + \frac{1}{2} A {T_2}^2 \\ l_3 = x_3 - x_2 = \ f_2 T_3 + \frac{1}{2} A {T_3}^2 - \frac{1}{6} J_3 {T_3}^3 \\ l_4 = x_4 - x_3 = f_3 T_4 \\ l_5 = x_5 - x = f_4 T_5 - \frac{1}{6} J_5 {T_5}^3 \\ l_6 = x_6 - x_5 = f_5 T_6 - \frac{1}{2} D {T_6}^2 \\ l_7 = x_7 - x_6 = \ f_6 T_7 - \frac{1}{2} D {T_7}^2 + \frac{1}{6} J_7 {T_7}^3 \end{cases} \qquad \text{(Eq. 77)}$$

Additionally, given the characteristics of the curves:

$$A = J_1 T_1 = J_3 T_3 \qquad \text{(Eq. 78)}$$

$$D = J_5 T_5 = J_7 T_7 \qquad \text{(Eq. 79)}$$

$$f_3 = F \rightarrow T_2 = \frac{1}{A}\left[F - f_s - \frac{1}{2} J_1 {T_1}^2 - A T_3 + \frac{1}{2} J_3 {T_3}^2\right] \qquad \text{(Eq. 80)}$$

$$\begin{aligned} f_7 = f_6 - D T_7 + \frac{1}{2} J_7 {T_7}^2 = f_e \rightarrow T_6 \\ = \frac{1}{D}\left[F - f_e - \frac{1}{2} J_5 {T_5}^2 - D T_7 + \frac{1}{2} J_7 {T_7}^2\right] \end{aligned} \qquad \text{(Eq. 81)}$$

$$x_7 = x_6 + f_6 T_7 - \frac{1}{2} D {T_7}^2 + \frac{1}{6} J_7 {T_7}^3 = L \qquad \text{(Eq. 82)}$$

A.1.2. initialization

Inputs

T_s: control loop sampling period
L: total distance of travel
N_i: total number of interpolation steps
f_s, F, f_e: initial, desired and final feedrates
A, D: desired acceleration and deceleration magnitudes
J: desired jerk magnitude

Conditions

To determine if there is acceleration or deceleration in the beginning of the motion (Eq. 83) is used.

$$\begin{cases} A = sgn(F - f_s) \cdot |A|\,, & J_1 = J_3 = sgn(A) \cdot |J| \\ D = sgn(F - f_e) \cdot |D|\,, & J_5 = J_7 = sgn(D) \cdot |J| \end{cases} \qquad \text{(Eq. 83)}$$

The total number of interpolation steps is limited by the existence of acceleration and deceleration stages. If both of them hold, then, the number of interpolation steps has to be at least 4 ($N_i \geq 4$), in order to allow at least one step for phases 1, 3, 5 and 7.

From the specified initial conditions and the motion equations, the following equations can be set:

$$J \leq min\left(\frac{|A|}{T_s}, \frac{|D|}{T_s}\right), \qquad |A| > 0 \; or \; |D| > 0 \qquad \text{(Eq. 84)}$$

$$J = min\left(\frac{|A|}{T}, \frac{|D|}{T}\right), \qquad A = 0 \; and \; D = 0 \qquad \text{(Eq. 85)}$$

$$T_2 = \frac{F - f_s}{A} - \frac{A}{J_1} \geq 0, \qquad A \neq 0 \qquad \text{(Eq. 86)}$$

$$A = sgn(A) \cdot \sqrt{J_1(F - f_s)}; \quad T_2 = 0, \qquad A = 0 \qquad \text{(Eq. 87)}$$

$$T_6 = \frac{F - f_e}{D} - \frac{D}{J_5} \geq 0, \qquad D \neq 0 \qquad \text{(Eq. 88)}$$

$$D = sgn(D) \cdot \sqrt{J_5(F - f_e)}; \quad T_6 = 0, \qquad D = 0 \qquad \text{(Eq. 89)}$$

L characteristic curve:

$$L = \left(\frac{1}{2A} + \frac{1}{2D}\right)F^2 + \left(\frac{A}{2J_1} + \frac{D}{2J_5} + T_4\right)F + \left(\frac{Af_s}{2J_1} + \frac{Df_e}{2J_5} - \frac{f_s^2}{2A} - \frac{f_e^2}{2D}\right), \qquad \text{(Eq. 90)}$$

$$\mathrm{J_1 = J_3 \; and \; J_5 = J_7}$$

$$T_4 = \frac{1}{F}\left[L - \left\{\left(\frac{1}{2A} + \frac{1}{2D}\right)F^2 + \left(\frac{A}{2J_1} + \frac{D}{2J_5}\right)F + \left(\frac{Af_s}{2J_1} + \frac{Df_e}{2J_5} - \frac{f_s^2}{2A} - \frac{f_e^2}{2D}\right)\right\}\right] \geq 0, \qquad T_4 \geq 0 \qquad \text{(Eq. 91)}$$

$$F = \frac{-\beta + \sqrt{\beta^2 - 4\alpha\gamma}}{2\alpha}, \qquad T_4 = 0 \tag{Eq. 92}$$

being $\alpha = \frac{1}{2A} + \frac{1}{2D}$, $\beta = \frac{A}{2J_1} + \frac{D}{2J_5}$ and $\gamma = \frac{Af_s}{2J_1} + \frac{Df_e}{2J_5} - \frac{f_s^2}{2A} - \frac{f_e^2}{2D} - L$. If the equation possesses complex roots, then $fs = fe = 0$.

Finally, path generation initialization is reinvoked to use the new maximum federate and J, A, D and F are readjusted to obtain a realizable case.

Number of interpolation steps for each phase.

$$N_1 = round\left(\frac{l_1}{\Delta_s}\right), N_3 = round\left(\frac{l_3}{\Delta_s}\right), N_5 = round\left(\frac{l_5}{\Delta_s}\right), N_7 = round\left(\frac{l_7}{\Delta_s}\right) \tag{Eq. 93}$$

If any of those turned out to be 0, they would be set to 1.

The total numbers of steps for the acceleration and deceleration stages are:

$$N_{acc} = round\left(\frac{l_1 + l_2 + l_3}{\Delta_s}\right), \qquad N_{dec} = round\left(\frac{l_5 + l_6 + l_7}{\Delta_s}\right) \tag{Eq. 94}$$

Number of steps for jerkless acceleration and deceleration stages:

$$N_2 = N_{acc} - (N_1 + N_3), \qquad N_6 = N_{dec} - (N_5 + N_7) \tag{Eq. 95}$$

Number of constant federate steps:

$$N_4 = N - (N_{acc} + N_{dec}) \tag{Eq. 96}$$

Finally, the travel lengths are quantified as:

$$l'_k = N_k \cdot \Delta_s, k = 1, \dots, 7 \tag{Eq. 97}$$

Lastly, acceleration and jerk values are readjusted to maintain the specified feedrates for the new travel lengths.

A.1.3. Continuously executed part

Once the initialization is completed, at each step of interpolation the following equations are executed:

$$s(\tau_k) = \frac{1}{6} j_{0k}\tau_k^3 + \frac{1}{2} a_{0k}\tau_k^2 + f_{0k}\tau_k + s_{0k}, \qquad 0 \le \tau_k \le T_k \tag{Eq. 98}$$

Where j_{0k}, a_{0k}, f_{0k}, s_{0k}have been calculated in the initialization phase, j_{0k} is the jerk value at the beginning of the kth phase, a_{0k} is the acceleration value at the beginning of the kth phase, $f_{0k}$is the federate value at the beginning of the kth phase and s_{0k} is the displacement value at the beginning of the kth phase.

The total distance travelled from the beginning of the kth phase until the nth interpolation step in this phase is:

$$s_{kn}(\tau_{kn}) = n\Delta s = \frac{1}{6} j_{0k}\tau_{kn}^{3} + \frac{1}{2} a_{0k}\tau_{kn}^{2} + f_{0k}\tau_{kn} + s_{0k}, \quad 0 \leq \tau_k \leq T_k \qquad \text{(Eq. 99)}$$

$$T_{kn}^{i} = \tau_{kn} - \tau_{k,n-1} \qquad \text{(Eq. 100)}$$

A.2. Yong Jeong et al., 2005 [6]

Yong Jeong et al. propose a readily implementable algorithm to be used in generating velocity profiles for point-to-point motion trajectories and speed profiles for path motion trajectories. It includes only the low-order polynomial equations that can be evaluated analytically. The main difference with the algorithm proposed by Erkokmaz and Altintas [2] is that the one proposed by Yong Jeong et al. does not need iterative steps in order to decide the coefficients of jerk limited speed profile.

A.2.1. Time-fixed profile motion equations

The motion equations used in this paper are the ones proposed by Erkorkmaz and Altintas [2]. The paper points out the difference between AFP (acceleration-first-profile) and DFP (deceleration-first-profile). To decide if the profile is AFP or DFP, one can use:

1. AFP:

$$\frac{x_f - x_0}{T} \geq \frac{v_0 + v_f}{2} \qquad \text{(Eq. 101)}$$

2. If Q(t) is a DFP whose constraint set is $\{\, -x0, -xf, -v0, -vf, V, A, D, J\}$:

$$Q(t) = -X(t), \qquad \dot{Q(t)} = -\dot{X(t)}, \qquad \ddot{Q(t)} = -\ddot{X(t)}$$

3. If $x_f{}' - x_0{}' \geq x_f - x_0$, there is an AFP solution whose constraint set is $\{\, x0', xf', v0, vf, V, A, D, J\}$.
4. If there is not an AFP solution for a given constraint set, there is a DFP solution under it.

Using this, DFP problems can be converted into AFP.

Formulation (algorithm)

Boundary conditions: {x$_0$, x$_f$, v$_0$, v$_f$}
Limit constraints: {V, A, D, J, T}

Any profile satisfying the boundary conditions and the limit constraints can be a solution of the problem on condition that its travelling period is T. Therefore, it is necessary to implement some extra conditions in order to choose a solution. In this paper, a jerk-minimization criterion is proposed because smaller jerks provide smoother acceleration profiles.

The effective jerk-limit is defined as: $J_e = \gamma J$

1) Decide whether the profile is AFP or not by using (Eq. 101).
2) If it is DFP, transform the problem into AFP.
3) Evaluating the simultaneous equations of the displacement condition and the peak velocity condition, find $y, \bar{y}\ and\ \gamma.$

$$y = T_1 + T_2 + T_3 \qquad \text{(Eq. 102)}$$

$$\hat{y} = T_4 \qquad \text{(Eq. 103)}$$

$$\bar{y} = T_5 + T_6 + T_7 \qquad \text{(Eq. 104)}$$

$$L = 0.5(v_0 + v_p)x + 0.5(v_p + v_f)\bar{y} + v_p\hat{y} \qquad \text{(Eq. 105)}$$

$$0 \le y, \bar{y}, T - y - \bar{y} \le T \qquad \text{(Eq. 106)}$$

$$0 < \gamma \le 1 \qquad \text{(Eq. 107)}$$

4) Readjust acceleration, deceleration and jerk limits if needed:
   a. $A \leftarrow 0.5Jy, if\ y < \frac{2A}{\gamma J}\ (no\ const.acc.)$
   b. $D \leftarrow 0.5J\bar{y}, if\ \bar{y} < \frac{2D}{\gamma J}\ (no\ const.dec.)$
   c. $J = J_e \leftarrow \gamma J$
5) Determinate each time interval with the adjusted A, D, J:
   $T_1 = \frac{A}{J},\ T_2 = y - 2T_1,\ T_3 = T_1, T_4 = \hat{y}, T_5 = \frac{D}{J}, T_6 = \bar{y} - 2T_5, T_7 = T_5$

Then, several different cases are introduced in the paper, by adapting the equations above depending on each instance.

A.3. Kim, I-Ming and Teck-Chew, 2007 [3]

Kim et al. [3] generalize the model of polynomial s-curve motion profiles in a recursive form, proposing a general algorithm to design s-curve trajectories with jerk bounded and time-optimal consideration.

The approach suggested is to define the s-curve model as piecewise polynomials of 2^n-1 segments, being n the order of the curve model. Thus, for a 3rd order s-curve model, the number of segments to be connected is 7, while for a 4th order it is 15.

Unlike in the other algorithms proposed, in this case the order of each curve is not defined, but a general behaving algorithm is introduced. Therefore, the same formulae can be used for a 3rd order curve or for a 5th or 7th one.

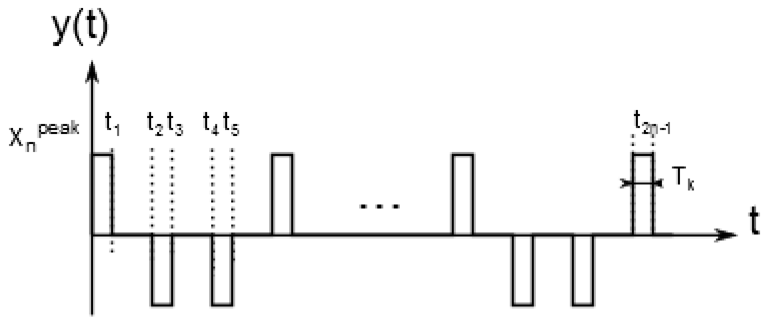

Figure 20. nth order s-curve model

A.3.1. Problem definition

Inputs: peak values of kinematic features (X_1^{peak}, X_2^{peak}, … , X_n^{peak}).
Aim: design a polynomial s-curve trajectory while optimizing the time of motion.

A.3.2. Algorithm

```
for p=n to 1 {
        $T_p = 0$                                                          (Eq. 108)
}
for p=n to 1{
        $X_0 = \frac{X_n}{2^n}\prod_{i=1}^{n}\left[\left(\sum_{j=0}^{i-1} 2^j T_{n+1-i+j}\right) + T_{n+1-i}\right]$      (Eq. 109)
    for q=1 to (p-1){
        $X_q^{max} = \frac{X_n}{2^{n-q}}\prod_{i=1}^{n-q}\left[\left(\sum_{j=0}^{i-1} 2^j T_{n+1-i+j}\right) + T_{n+1-i}\right]$      (Eq. 110)
        if $X_q^{max} > X_q^{peak}$, recalculate $T_p$ from
            $X_q^{peak} = \frac{X_n}{2^{n-q}}\prod_{i=1}^{n-q}\left[\left(\sum_{j=0}^{i-1} 2^j T_{n+1-i+j}\right) + T_{n+1-i}\right]$      (Eq. 111)
}
}
```

All of the time periods are set to 0 initially. Then, T_p is calculated by the peak value of X, X_0. This T_p is used to calculate the maximum value of the kinematic features, X_q^{max}, and then it is compared with the input peak value. T_p is recalculated until no peak input is exceeded.

A.4. Ha, Rew and Kim, 2008 [4]

Ha, Rew and Kim propose an asymmetric S-curve motion profile to ease manipulation of jerks during the arrival time in order to reduce the residual vibration. A design parameter called jerk ratio is introduced to scale down the jerks during the deceleration period. Because of this, motion parameters are remarkably simplified in analytic forms. As the jerk ratio increases, the residual vibration decreases, although the motion profile is lengthened.

A.4.1. Formulation

Given values: V_{max}, A_{max}, δ_{target}

Constraints: Velocity: $v(t=t_0)= v(t=t_7)=0$,
$-V_{max} \leq v \leq V_{max}$
Position: $x(t=t_0)= 0$
$x(t=t_7)= \delta_{target}$
Acceleration: $-A_{max} \leq a \leq A_{max}$

A.4.2. Motion equations

Table 2. Equations from the s-curve model proposed in [4].

Period	Equations			
	Acceleration	Velocity	Position	
$[t_0, t_1]$	$a_1 = J\Delta t_j$	$v_1 = \frac{1}{2}J\Delta t_j{}^2$	$\Delta x_1 = \frac{1}{6}J\Delta t_j{}^3$	(Eq. 112)
$[t_1, t_2]$	$a_2 = J\Delta t_j$	$v_2 = J\Delta t_j\left(\frac{1}{2}\Delta t_j + \Delta t_a\right)$	$\Delta x_2 = \frac{1}{2}J\Delta t_j\Delta t_a\left(\Delta t_j + \Delta t_a\right)$	(Eq. 113)
$[t_2, t_3]$	$a_3 = 0$	$v_3 = J\Delta t_j\left(\Delta t_j + \Delta t_a\right)$	$\Delta x_3 = J\Delta t_j{}^2\left(\frac{5}{6}\Delta t_j + \Delta t_a\right)$	(Eq. 114)
$[t_3, t_4]$	$a_4 = 0$	$v_4 = J\Delta t_j\left(\Delta t_j + \Delta t_a\right)$	$\Delta x_4 = J\Delta t_j\left(\Delta t_j + \Delta t_a\right)\Delta t_v$	(Eq. 115)
$[t_4, t_5]$	$a_5 = -\frac{J\Delta t_j}{\gamma}$	$v_5 = J\Delta t_j\left(\frac{1}{2}\Delta t_j + \Delta t_a\right)$	$\Delta x_5 = J\gamma\Delta t_j{}^2\left(\frac{5}{6}\Delta t_j + \Delta t_a\right)$	(Eq. 116)
$[t_5, t_6]$	$a_6 = -\frac{J\Delta t_j}{\gamma}$	$v_6 = \frac{1}{2}J\Delta t_j{}^2$	$\Delta x_6 = \frac{1}{2}J\gamma\Delta t_j\Delta t_a\left(\Delta t_j + \Delta t_a\right)$	(Eq. 117)
$[t_6, t_7]$	$a_7 = 0$	$v_7 = 0$	$\Delta x_7 = \frac{1}{6}J\gamma\Delta t_j{}^3$	(Eq. 118)

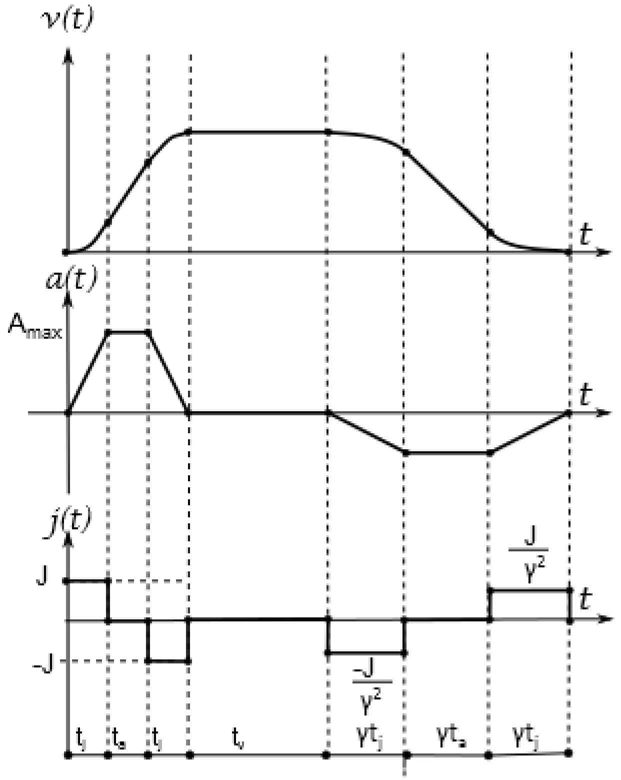

Figure 21. Asymetrical s-curve velocity profile

A.4.3. Algorithm

1) Choose β to fix the jerk level and γ≥1 for the smooth arrival motion. Calculate $\delta_s^*$and δ_l^*.

$$\delta_s^* = \delta_x|_{(\Delta t_j=\Delta t_j{}^*)} = (1+\gamma)\beta^2\frac{V_{max}^2}{A_{max}} \qquad \text{(Eq. 119)}$$

$$\delta_l^* = \delta_x|_{(\Delta t_a=\Delta t_a{}^*)} = (1+\gamma)\frac{(1+\beta)}{2}\frac{V_{max}^2}{A_{max}} \qquad \text{(Eq. 120)}$$

2) <u>If</u> $\delta_{target} < \delta_s^*$, apply parameters $\Delta t_a = \Delta t_v = 0$ and

$$\Delta t_j = \sqrt[3]{\frac{\Delta t_j{}^*}{(1+\gamma)A_{max}}\delta_{target}} \qquad \text{(Eq. 121)}$$

3) <u>If</u> $\delta_s^* \leq \delta_{target} \leq \delta_l^*$, apply the profiles with parameters $\Delta t_j = \Delta t_j{}^*$, $\Delta t_v = 0$ and

$$\Delta t_a = -\frac{3\beta V_{max}}{2A_{max}}\sqrt{\left(\frac{\beta V_{max}}{2A_{max}}\right)^2 + \frac{2\delta_{target}}{(1+\gamma)A_{max}}} \qquad \text{(Eq. 122)}$$

4) <u>If</u> $\delta_{target} > \delta_l^*$, apply the profiles with parameters $\Delta t_j = \Delta t_j{}^*$, $\Delta t_a = \Delta t_a{}^*$ and

$$\Delta t_v = \frac{\delta_{target} - \delta_l^*}{V_{max}} \qquad \text{(Eq. 123)}$$

Appendix B. Pressure-sinkage: state of the art

B.1. Meiron-Griffith and Spenko [12]

Meiron-Griffith and Spenko [12] propose the following equations to take into account the effect of the diameter of the wheels. This model is valid for all wheel diameters.

$$\sigma = \hat{k} z^{\hat{n}} D^{\hat{m}} \qquad \text{(Eq. 124)}$$

Where D is the wheel diameter and $\hat{k}$, $\hat{n}$, and $\hat{m}$ are fitting constants empirically determined for each soil type.

Then, the sinkage equations results in:

$$z_0 = \frac{3W}{b(3-\hat{n})\hat{k}D^{\hat{m}+0.5}}^{\frac{2}{2\hat{n}+1}} \qquad \text{(Eq. 125)}$$

And finally, the compaction resistance is:

$$R_c = b\hat{k}D^{\hat{m}} \frac{{z_0}^{\hat{n}+1}}{\hat{n}+1} \qquad \text{(Eq. 126)}$$

The equations above are then used to compute the drawbar pull. However, this modification on Bekker equations does not include the semi-elliptical distribution of normal stress that exists under a wheel.

B.2. Meiron-Griffith and Spenko, 2011 [15]

In this paper, Meiron-Griffith and Spenko [15] modify the model proposed previously in order to add the semi-elliptical distribution beneath a wheel. the distribution of the stresses is shown in Figure 22.

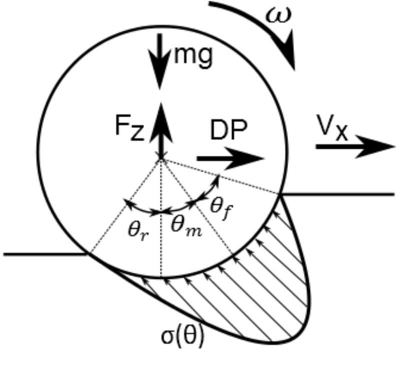

Figure 22. Semi-elliptical distribution of normal stress beneath a wheel

Static sinkage:

$$z_0 = \frac{3W}{b(3-\hat{n})\hat{k}D^{\hat{m}+0.5}}^{\frac{2}{2\hat{n}+1}} \qquad \text{(Eq. 127)}$$

Dynamic sinkage, in the form proposed by [26]

$$z_d = K_{ss} z_0 \tag{Eq. 128}$$

$$K_{ss} = \frac{1+i}{1-0.5i} \tag{Eq. 129}$$

$$i = \frac{\omega r - v_b}{\omega r} \tag{Eq. 130}$$

Being ω the wheel angular velocity; i, the wheel slip; r, the wheel radius, and v_b, the vehicle's speed.

The normal stress can be calculated as a function of θ:

$$\sigma(\theta) = \begin{cases} \hat{k} r^{\hat{n}} (cos\theta - cos\theta_s)^{\hat{n}} D^{\hat{m}}, \ for\ \theta_m \le \theta_s \le \theta_f \\ \hat{k} D^{\hat{m}} r^{\hat{n}} \left[cos\left(\theta_f - \frac{(\theta-\theta_r)\,(\theta_f-\theta_m)}{(\theta_m-\theta_r)}\right) - cos\theta_f \right]^{\hat{n}}, \ for\ \theta_r \le \theta \le \theta_m \end{cases} \tag{Eq. 131}$$

$$\theta_m = \frac{\theta_r + \theta_f}{2} \tag{Eq. 132}$$

Where θ is the arbitrary angle along the stress arc; θ_s, the static wheel-soil contact angle; θ_r, the rear wheel-soil contact angle; θ_f, the forward wheel-soil contact angle, and r, the wheel radius.

Longitudinal shear stress (Janosi-Hanamoto equation):

$$\tau = \tau_{max}\left(1 - exp^{\frac{-j}{\kappa}}\right) \tag{Eq. 133}$$

Being $\tau_{max}$the shear strength of the soil; j, the longitudinal soil deformation, and κ, the shear deformation parameter.

As a function of θ, shear stress is:

$$\tau(\theta) = (c + \sigma(\theta)tan\phi)\left(1 - exp^{\frac{-j(\theta)}{\kappa}}\right) \tag{Eq. 134}$$

Finally, shear and normal stresses are combined to estimate the total force acting on the wheel and consequently, the drawbar pull:

$$F = rb\int_{\theta_r}^{\theta_f} \{\tau(\theta)cos\theta - \sigma(\theta)sin\theta\} d\theta \tag{Eq. 135}$$

B.3. Shmulevich, Mussel and Wolf, 1998 [16]

Shmulevich et al. develop a soil-wheel interaction simulation model to study the effect of velocity on an off-road wheel performance.

The soil-wheel mathematical model used is that for a moving wheel in steady-state condition, this is, constant speed and zero acceleration. For the mentioned model, the formulation is based on the equilibrium equations of the wheel. The effect of the soil bulldozing wave pushed before the wheel is neglected.

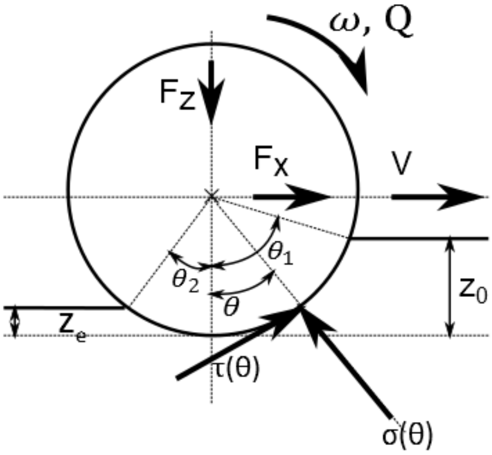

Figure 23. Rigid Wheel soil-wheel interaction

B.3.1. Equilibrium equations

- z direction:

$$bR\int_{\theta_2}^{\theta_1}[\tau(\theta)\sin(\theta)+\sigma(\theta)\cos(\theta)]d\theta - F_z = 0 \qquad \text{(Eq. 136)}$$

- x direction:

$$bR\int_{\theta_2}^{\theta_1}[\tau(\theta)\cos(\theta)+\sigma(\theta)\sin(\theta)]d\theta - F_x = 0 \qquad \text{(Eq. 137)}$$

- Moment equation:

$$bR^2\int_{\theta_2}^{\theta_1}\tau(\theta)d\theta - Q = 0 \qquad \text{(Eq. 138)}$$

Where the entry and exit angles can be calculated as:

$$\theta_1 = arc\cos\left(1-\frac{z_0}{R}\right) \qquad \text{(Eq. 139)}$$

$$\theta_2 = -arc\cos\left(1-\frac{z_e}{R}\right) \qquad \text{(Eq. 140)}$$

Velocity in the normal direction:

$$V_N = V\,sin\theta = \omega R(1-s)sin\theta \qquad \text{(Eq. 141)}$$

Velocity in the tangential direction:

$$V_L = \omega R - V\,cos\theta = \omega R[1-(1-s)cos\theta] \qquad \text{(Eq. 142)}$$

Deformation in the normal and tangential directions:

$$j_N = \int_0^t V_N dt = R(1-s)(-cos\theta_1 + cos\theta) \quad \text{(Eq. 143)}$$

$$j_L = \int_0^t V_L dt = R[(\theta_1 - \theta) - (1-s)(sin\theta_1 - sin\theta)] \quad \text{(Eq. 144)}$$

The presupposition of the equality of the speed of a wheel element at the contact area and the velocity of a soil element is correct in the normal direction whenever there is contact between the soil and the wheel. Regarding the tangential direction, it is right only if there is enough friction force to prevent slippage.

For a loose soil, which is the case of off-road terrains, Shmulevich et al. [16] decide to take a similar approach to the one used by Wong and Reece, although replacing the speed or penetration with speed in the normal direction of the wheel.

Soil stress in the normal direction is:

$$\sigma = (k_1 + k_2 b)\left(\frac{j_N}{b}\right)^n \left(\frac{V_{NS}}{u_0}\right)^m \quad \text{(Eq. 145)}$$

$$V_{NS} = \max\left(\frac{dj_N}{dt}, minu_0\right) = \max(V_N, minu_0) \quad \text{(Eq. 146)}$$

On non-elastic soils, the wheel-contact area, the soil stress at the rear sector and the rebound deformation are equal to zero. On the other hand, for elastic soils these values can be calculated using:

$$\sigma = (k_1 + k_2 b)\left(\frac{j_{N_{max}}}{b}\right)^n \left(\frac{minu_0}{u_0}\right)^m - \left(j_{N_{max}} - j_N\right)k_e \quad \text{(Eq. 147)}$$

$$j_{N_{max}} = z_0(1-s) \quad \text{(Eq. 148)}$$

Slip:

$$s = 1 - V(\omega R) \quad \text{(Eq. 149)}$$

Rebound deformation:

$$z_e = \frac{1}{k_e}(k_1 + k_2 b)\left(\frac{z_0(1-s)}{b}\right)^n \left(\frac{minu_0}{u_0}\right)^m \quad \text{(Eq. 150)}$$

Tangential stress can only be calculated where there is enough friction force to prevent slippage.

$$\tau = \tau_m\left[1 - \exp\left(-\frac{j_L}{k}\right)\right] \quad \text{(Eq. 151)}$$

$$\tau_m = (c + \sigma tan\phi) + C_d\left[1 - exp\left(\frac{-\alpha dj_L}{dt}\right)\right] \quad \text{(Eq. 152)}$$

$$\mu = \mu_0 exp(-C_m V_s + \mu_d) \quad \text{(Eq. 153)}$$

Finally, the minimum value for shear stress will be chosen for each point:

$$\tau = \min(\tau, \mu, \sigma) \quad \text{(Eq. 154)}$$

Tractive efficiency:

$$TE = \frac{F_x V}{Q\omega} \qquad \text{(Eq. 155)}$$

It has been shown that the results of the simulation model correspond to the experimental data, improving the prediction given by Wong's models, especially with slip values around zero. The values used in the experiment are shown in Table 3.

Table 3. Values used in the experiment

Soil	φ (deg)	c (N/m^2)	k_1 (N/m^2)	k_2 (N/m^3)	n	m	u_0 (m/s)	minu$_0$ (m/s)
Compact soil	33.3	691	1.382e^5	6.803e^5	0.47	0.30	0.03	0.0015
Loose sand	31.1	829	0	5.442e^5	1.15	0.30	0.03	0.0100
Sand loam	33.3	691	0	8.660e^5×b^{n-1}	0.36	0.12	1.00	0

B.3.2. Effect of velocity on wheel performances

The experiments show that as velocity increments, so do the maximum normal and tangential stresses, while the contact zone decreases, and so does the angular zone. Moreover, it is proved that an increase in relative velocity causes a lower relative wheel sinkage. Overall, the conclusion is that higher relative velocity results in better wheel performance in terms of sinkage, net tractive ratio and tractive efficiency.

B.3.3. Effect of the type of soil on wheel performances

The free rolling wheel force ratio and the sinkage are lower on compact soil than on loose sand, while these type of soils show bigger values for net tractive ratio and maximum tractive efficiency.

B.4. Djohor, M'Sirdi and Naamane [1]

Assuming a mechanical system in which both the wheel and the ground are deformable, one can consider that the wheel-soil contact surface has the shape of a plane tangent to the surface of the wheel, where the contact surface is rectangle shaped.

B.4.1. Wheel-soil relative motion

Assuming free-slip rolling, the slip values can be calculated as shown below:

Longitudinal slip:

$$s = \begin{cases} \dfrac{V_x - V_r}{V_x}, during\ breaking \\ \dfrac{V_x - V_r}{V_r}, during\ traction \end{cases} \qquad \text{(Eq. 156)}$$

Longitudinal slip rate:

$$S_s = |s| \tag{Eq. 157}$$

Lateral slip rate:

$$S_\alpha = \begin{cases} |\tan(\alpha)| \; during\ breaking \\ (1 - S_s)|\tan(\alpha)| \; during\ traction \end{cases} \tag{Eq. 158}$$

Slip angle:

$$\alpha = arctan\left(\frac{V_y}{V_x}\right) \tag{Eq. 159}$$

Slip velocity of the contact surface:

$$V_s = \sqrt{(V_x - V_r)^2 + {V_y}^2} \tag{Eq. 160}$$

Slip velocity's angle of direction from the longitudinal axle:

$$\beta = arcsin\left(\frac{V_y}{V_s}\right) \tag{Eq. 161}$$

B.4.2. Wheel-soil behaviour laws

When modelling a soil-wheel system, there are three main model characterizations, depending on whether the soil and the wheel are considered rigid or deformable. When considering both of them deformable, the suggested approach is to decompose the contact surface in two parts:

1) Rigid wheel. The soil is deformed both vertically due to the vehicle's load and tangentially owing to tangential forces.
2) Rigid soil. The wheel gets both vertical and tangential reactions from the soil.

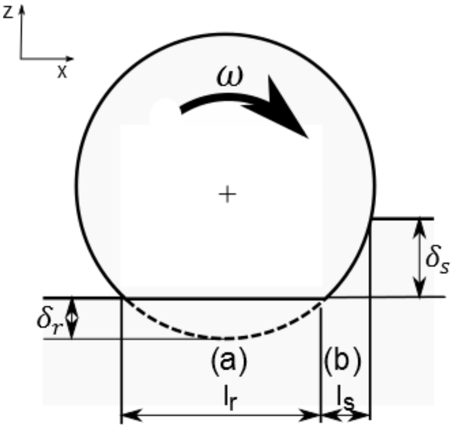

Figure 24. approximated shape of the contact surface

The assumption of independence between soil's deformation in areas (a) and (b) is taken.

Vertical deformation of the soil

$$\delta_s \equiv z_0 = {\frac{3F_z}{\omega(3-n)\left(\frac{k_c}{\omega} + k_\phi\right)\sqrt{D}}}^{\frac{2}{2n+1}} \tag{Eq. 162}$$

Vertical deformation of the wheel

$$\delta_r = \frac{F_z}{k_z} \tag{Eq. 163}$$

Length of area (a) $$l_r = \sqrt{8r\delta_r} \qquad \text{(Eq. 164)}$$

Length of area (b) $$l_s = \sqrt{2r(\delta_s + \delta_r)} - \sqrt{2r\delta_r} \qquad \text{(Eq. 165)}$$

B.4.3. Contact force expression

Longitudinal forces $$F_\xi = F_\xi{}^{(a)} + F_\xi{}^{(b)} \qquad \text{(Eq. 166)}$$

Lateral forces $$F_\eta = F_\eta{}^{(a)} + F_\eta{}^{(b)} \qquad \text{(Eq. 167)}$$

The sign of the forces is determined by:

$$F_x = -sign(s)\, F_\xi \qquad \text{(Eq. 168)}$$

$$F_y = -sign(\alpha)\, F_\eta \qquad \text{(Eq. 169)}$$

Depending on the values of the slip rate, the forces introduced above can be calculated from the following equations:

$$If\ \sqrt{S_s{}^2 + S_\alpha{}^2} \leq \frac{K}{l_s} \qquad \begin{cases} F_\xi{}^{(a)} = \dfrac{l_s}{2K}(Ac + F_z\tan(\phi))S_s \\ F_\eta{}^{(a)} = \dfrac{l_s}{2K}(Ac + F_z\tan(\phi))S_\alpha \end{cases} \qquad \text{(Eq. 170)}$$

$$If\ \sqrt{S_s{}^2 + S_\alpha{}^2} > \frac{K}{l_s} \qquad \begin{cases} F_\xi{}^{(a)} = [Ac + F_z\tan(\phi)]\left[1 - \dfrac{l_p}{l_s} + \dfrac{S_s l_p{}^2}{2Kl_s}\right] \\ F_\eta{}^{(a)} = [Ac + F_z\tan(\phi)]\left[1 - \dfrac{l_p}{l_s} + \dfrac{S_\alpha l_p{}^2}{2Kl_s}\right] \end{cases} \qquad \text{(Eq. 171)}$$

Where $$l_p = \frac{K}{\sqrt{S_s{}^2 + S_\alpha{}^2}} \qquad \text{(Eq. 172)}$$

$$If\ S_s < S_{sc}\ and\ S_\alpha < S_{\alpha c} \qquad \begin{cases} F_\xi{}^{(b)} = C_s S_s l_n{}^2 + \mu_{cx}F_z(1 - 3l_n{}^2 + 2l_n{}^3) \\ F_\eta{}^{(b)} = C_\alpha S_\alpha l_n{}^2 + \mu_{cy}F_z(1 - 3l_n{}^2 + 2l_n{}^3) \end{cases} \qquad \text{(Eq. 173)}$$

Where $$C_s = \frac{k_x \omega l_r{}^2}{2} \qquad \text{(Eq. 174)}$$

$$C_\alpha = \frac{k_y \omega l_r{}^2}{2} \qquad \text{(Eq. 175)}$$

$$l_n = 1 - \frac{1}{3\mu_c F_z}\sqrt{(C_s S_s)^2 + (C_\alpha S_\alpha)^2} \qquad \text{(Eq. 176)}$$

If $S_s \geq S_{sc}\ and\ S_\alpha \geq S_{\alpha c}$ $$\begin{cases} F_\xi{}^{(b)} = \mu_{cx}F_z \\ F_\eta{}^{(b)} = \mu_{cy}F_z \end{cases} \qquad \text{(Eq. 177)}$$

S_{sc} and $S_{\alpha c}$ are the critical values of the longitudinal slip rate beyond which the elastic constraints and deformations do not hold.

$$S_{sc} = \frac{3\mu_c F_z}{C_s} \qquad \text{(Eq. 178)}$$

$$S_{\alpha c} = \frac{C_s}{C_\alpha}\sqrt{{S_{sc}}^2 - {S_s}^2} \tag{Eq. 179}$$

B.4.4. Resistive forces expression

The vehicle load causes a sinkage effect on the vehicle. Due to compaction, the wheel receives a rolling resistance force:

$$\widetilde{F_x} = \omega\left(\frac{\delta_s^{n+1}}{n+1}\right)\left(\frac{k_c}{\omega} + k_\phi\right) \tag{Eq. 180}$$

B.5. Irani, Bauer and Warkentin, 2011 [9]

With the purpose of improving its tractive effort, radial paddles are used on wheels. The grousers cause oscillations in the dynamic profiles of certain dynamic parameters, such as sinkage or drawbar pull, which are not included in Bekker and Wong models. Moreover, the existence of grousers causes the shear stress to act at some distance of the wheel.

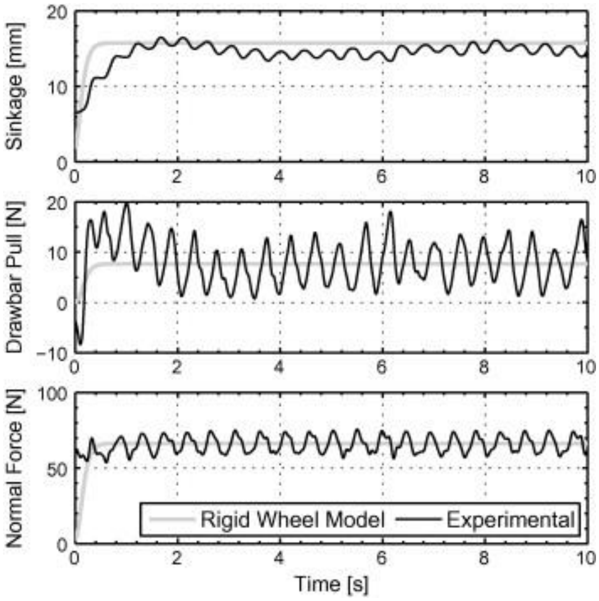

Figure 25. Experimental data of a rigid wheel with 16, 10 mm long grousers, operating at 0.25 slip and a 66 N normal load overlayed with a typical terramechanic model for a rigid wheel. [9]

To include this effect, two main solutions have been used: finite element methods and discrete element methods (DEM). The main problem this solutions present is that they demand high computational loads. Moreover, the use of DEM is still in development. In this paper, another solution is proposed, based on the expansion of the traditional terramechanics models.

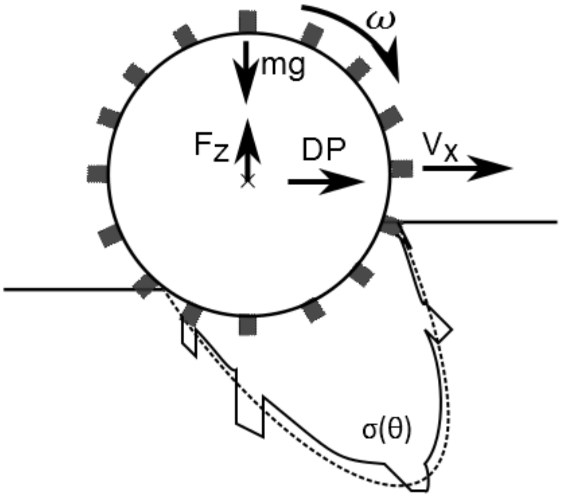

Figure 26. Proposed normal stress distribution for a wheel with grousers

The authors of this paper propose the following equation for the pressure-sinkage:

$$p(z) = \left(ck_c' + \gamma b k_\phi{}'\right)\left(\frac{z}{b}\right)^n + A_g sin\left(\omega_g t + \Phi\right) \tag{Eq. 181}$$

$$\omega_g = \frac{\omega}{n_g} \tag{Eq. 182}$$

The suggested formulation for the stress is:

$$\sigma_p = \gamma z N_\phi + q N_\phi + 2c\sqrt{N_\phi} \tag{Eq. 183}$$

Where N_ϕ is the flow value:

$$N_\phi = tan^2\left(45º + \frac{\phi}{2}\right) \tag{Eq. 184}$$

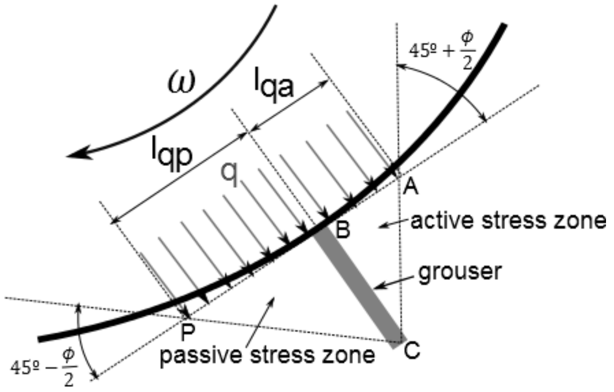

Figure 27. Active and passive stress zones in a wheel with grousers

It is considered that the amplitude term is affected mainly by two factors, A_σ, related to the active and passive stresses, and A_γ, related to the change in the local soil density around the wheel and grouser caused by the soil deformation due to the wheel.

$$A = A_\sigma + A_\gamma \qquad \text{(Eq. 185)}$$

$$A_\sigma = k_g' \bar{\sigma}_p \qquad \text{(Eq. 186)}$$

$$A_\gamma = k_a{}' l_c d_\gamma \qquad \text{(Eq. 187)}$$

Comparing the simulated results and the experimental ones, in regard of normal force, drawbar pull and sinkage, it appears that the proposed model improves the traditional terramechanics one when there are important dynamic effects due to grousers on the wheels.

Appendix C. Programs

This Appendix is a supplement for Chapter 5, providing examples of the execution of the proposed programs and flow diagrams. The values of the variables should not be taken as a reference, since they have no relationship to any real robot.

C.1. S-Curve calculation

C.1.1. Execution examples

Example 1
Table 4 shows the values of the input and output of this execution, while in Figure 28 the output graphs can be seen.

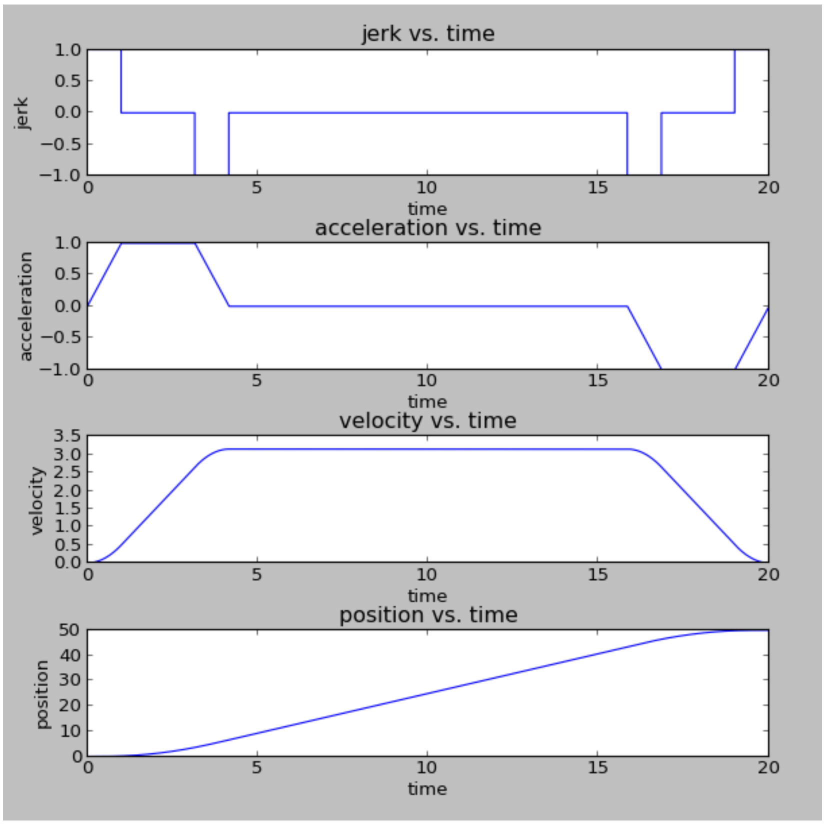

Figure 28. Output plot for Example 1

Table 4. Input and output values for program C.1 in Example 1

Input	a_0=0 $a_f = 0$ v_0=0 $v_f = 0$ x_0=0 x_f=50 $A = 1$ $D = 1$ J_1=1 J_3=1 $J_5 = 1$ J_7=1

Output	time= [0, 1.0000, 3.1557, 4.1557, 15.844, 16.844, 19.000, 20]
	position= [0, 0.16667, 3.5681, 6.5571, 43.443, 46.432, 49.833, 50]
	velocity= [0, 0.50000, 2.6557, 3.1557, 3.1557, 2.6557, 0.50000, 0]
	acceleration= [0, 1, 1, 0, 0, -1, -1, 0]
	jerk= [1, 0, -1, 0, -1, 0, 1, 0]

Example 2

In this case, the only change with respect to Example 1 is that jerk values for phase 1 and 3 are set higher than the ones for phases 5 and 7. Thus, the curve is no longer symmetric. The input and output values are shown in Table 5, while output plots of the program are shown in Figure 29.

Table 5. Input and output values for program C.1 in Example 2

Input	a_0=0 $a_f = 0$ v_0=0 $v_f = 0$ x_0=0 x_f=50 $A = 1$ $D = 1$ J_1=2 J_3=2 $J_5 = 1$ J_7=1

Output	time= [0, 0.50000, 3.0950, 3.5950, 15.905, 16.905, 19.000, 20]
	position= [0, 0.041667, 4.0575, 5.5633, 43.663, 46.591, 49.833, 50]
	velocity= [0, 0.25000, 2.8450, 3.0950, 3.0950, 2.5950, 0.50000, 0]
	acceleration= [0, 1, 1, 0, 0, -1, -1, 0]
	jerk= [2, 0, -2, 0, -1, 0, 1, 0]

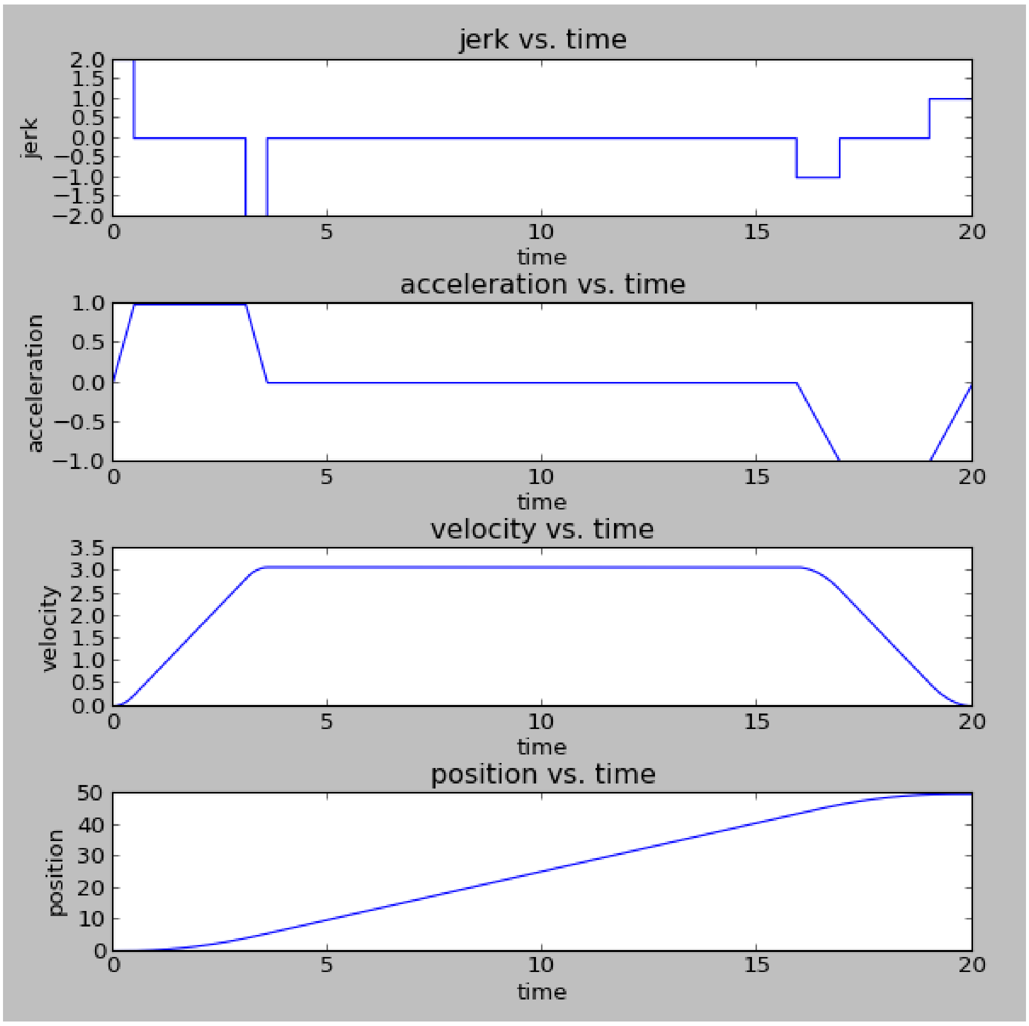

Figure 29. Output plot for Example 2.

Example 3
This example modifies the jerk values for phases 5 and 7, setting them in a lower value, while 1 and 3 are kept as in Example 2. In Table 6 the values of input and output variables are given. Figure 30 shows the output plots.

Table 6. Input and output values for program C.1 in Example 3

Input	a_0=0 $a_f = 0$ v_0=0 $v_f = 0$ x_0=0 x_f=50 $A = 1$ $D = 1$ J_1=2 J_3=2 $J_5 = 0.5$ J_7=0.5

Output	time= [0, 0.50000, 3.2195, 3.7195, 14.781, 16.781, 18.000, 20]
	position= [0, 0.041667, 4.4193, 5.9873, 41.598, 47.370, 49.333, 50]
	velocity= [0, 0.25000, 2.9695, 3.2195, 3.2195, 2.2195, 1.0000, 0]
	acceleration= [0, 1, 1, 0, 0, -1, -1, 0]
	jerk= [2, 0, -2, 0, -0.5, 0, 0.5, 0]

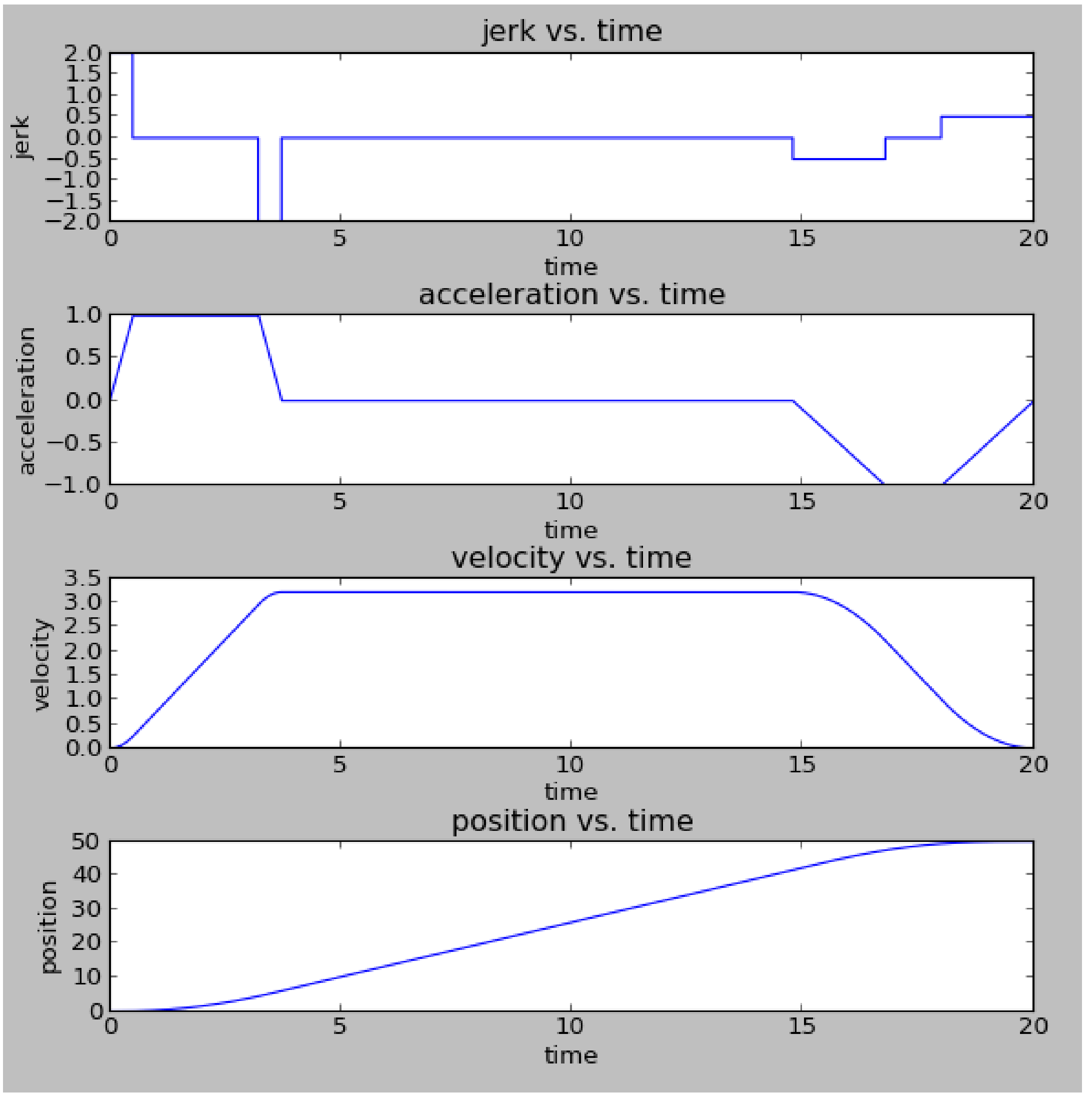

Figure 30. Output plot for Example 3.

Example 4

This execution example shows how the program works when the initial value for velocity and the desired ending acceleration are not zero. Besides, jerk values are set different for phases 1,3 and 5,7. The output graphs are the ones in Figure 31, while the input and output values can be fount in Table 7.

Table 7. Input and output values for program C.1 in Example 4

Input	a_0=0 $a_f = -0.5$ v_0=1 $v_f = 0$ x_0=0 x_f=50 $A = 1$ $D = 1$ J_1=1 J_3=1 $J_5 = 0.5$ J_7=0.5

Output	time= [0, 1.0000, 1.8861, 2.8861, 15.864, 17.864, 19.000, 20]
	position= [0, 1.1667, 2.8885, 5.6080, 43.064, 48.169, 49.667, 50]
	velocity= [1, 1.5000, 2.3861, 2.8861, 2.8861, 1.8861, 0.75000, 0]
	acceleration= [0, 1, 1, 0, 0, -1, -1, -0.5]
	jerk= [1, 0, -1, 0, -0.5, 0, 0.5, 0]

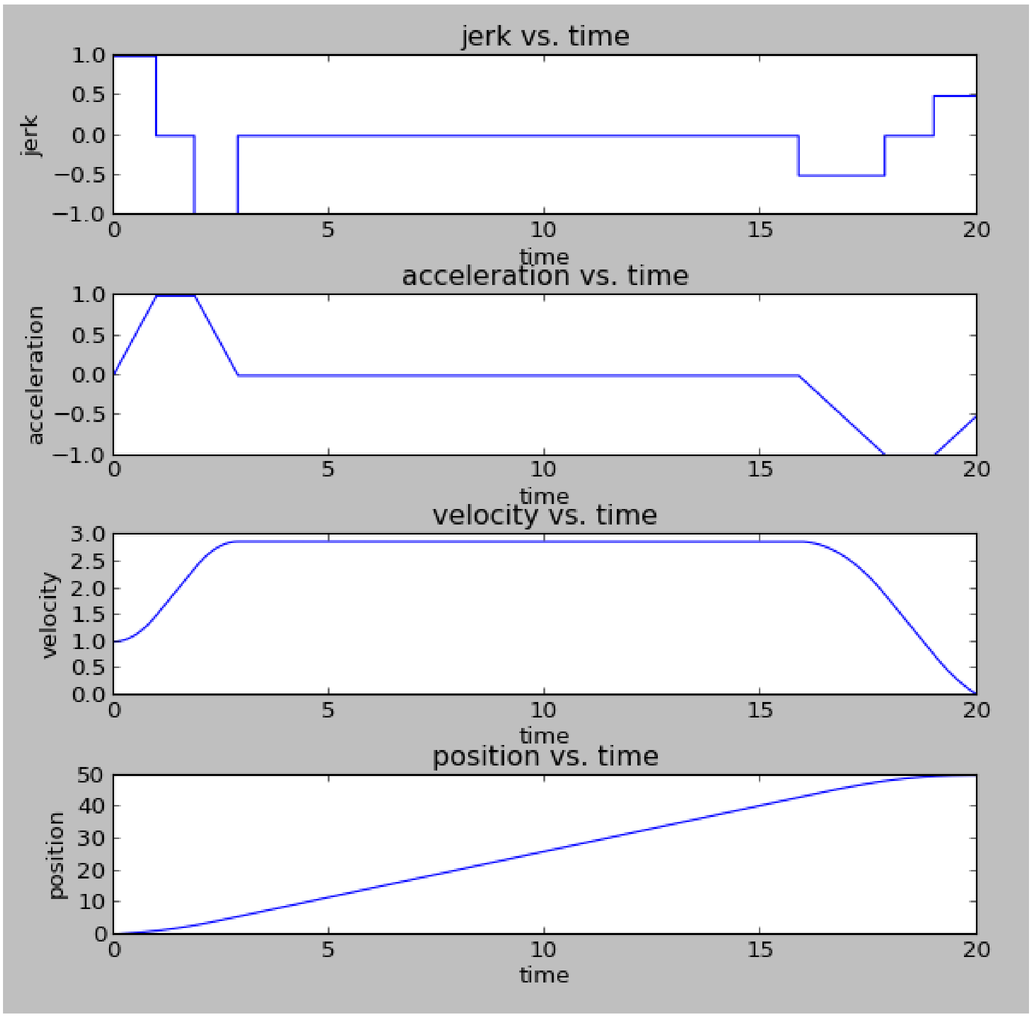

Figure 31. Output plot for Example 4.

C.1.2. Flow diagram

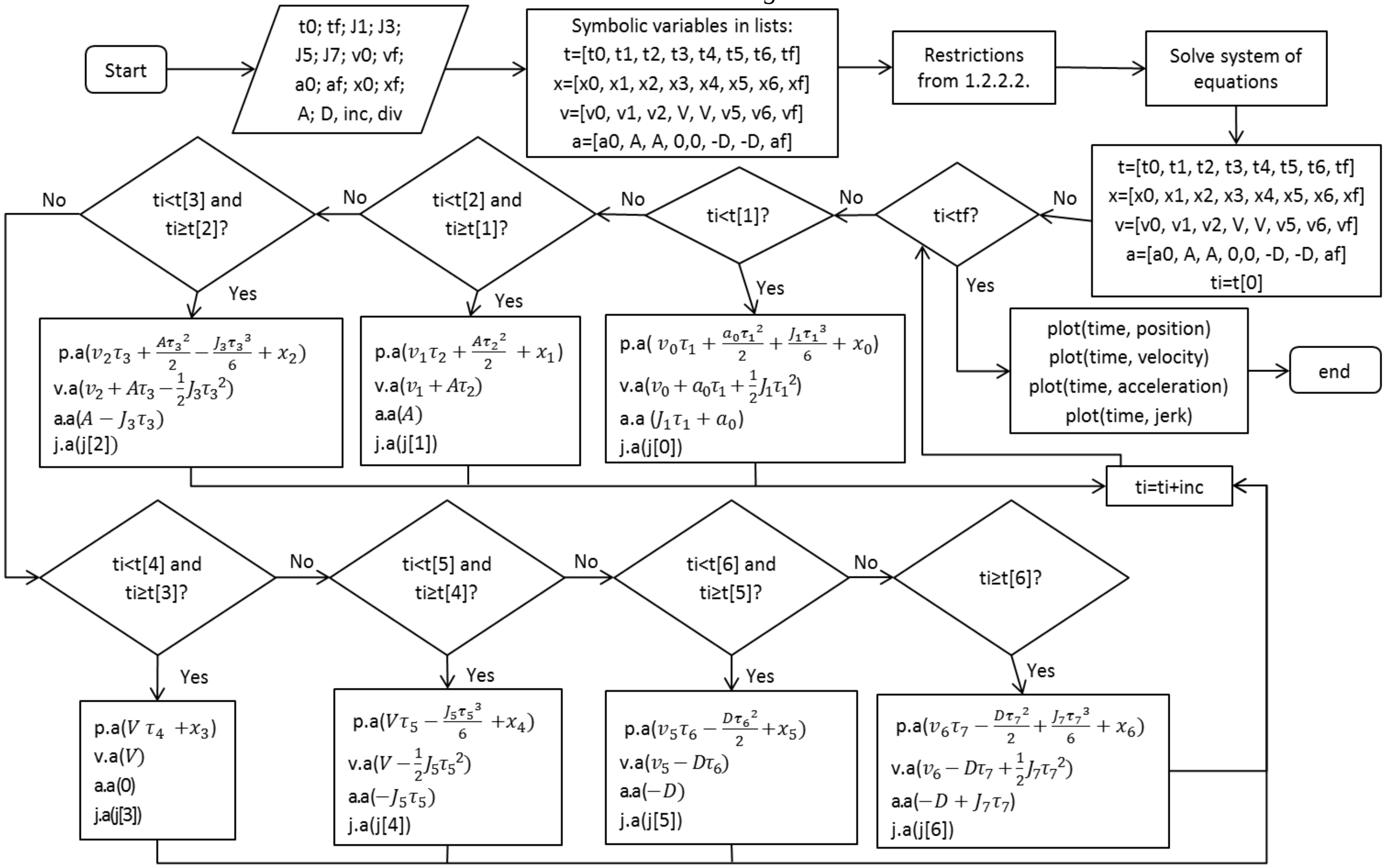
Start
t0; tf; J1; J3;
J5; J7; v0; vf;
a0; af; x0; xf;
A; D, inc, div
Symbolic variables in lists:
t=[t0, t1, t2, t3, t4, t5, t6, tf]
x=[x0, x1, x2, x3, x4, x5, x6, xf]
v=[v0, v1, v2, V, V, v5, v6, vf]
a=[a0, A, A, 0,0, -D, -D, af]
Restrictions from 1.2.2.2.
Solve system of equations
t=[t0, t1, t2, t3, t4, t5, t6, tf]
x=[x0, x1, x2, x3, x4, x5, x6, xf]
v=[v0, v1, v2, V, V, v5, v6, vf]
a=[a0, A, A, 0,0, -D, -D, af]
ti=t[0]
No
ti<tf?
Yes
plot(time, position)
plot(time, velocity)
plot(time, acceleration)
plot(time, jerk)
end
No
ti<t[1]?
Yes
p.a($v_0\tau_1 + \frac{a_0\tau_1^2}{2} + \frac{J_1\tau_1^3}{6} + x_0$)
v.a($v_0 + a_0\tau_1 + \frac{1}{2}J_1\tau_1^2$)
a.a ($J_1\tau_1 + a_0$)
j.a(j[0])
No
ti<t[2] and ti≥t[1]?
Yes
p.a($v_1\tau_2 + \frac{A\tau_2^2}{2} + x_1$)
v.a($v_1 + A\tau_2$)
a.a(A)
j.a(j[1])
No
ti<t[3] and ti≥t[2]?
Yes
p.a($v_2\tau_3 + \frac{A\tau_3^2}{2} - \frac{J_3\tau_3^3}{6} + x_2$)
v.a($v_2 + A\tau_3 - \frac{1}{2}J_3\tau_3^2$)
a.a($A - J_3\tau_3$)
j.a(j[2])
No
ti=ti+inc
ti<t[4] and ti≥t[3]?
Yes
p.a($V\ \tau_4\ + x_3$)
v.a(V)
a.a(0)
j.a(j[3])
No
ti<t[5] and ti≥t[4]?
Yes
p.a($V\tau_5 - \frac{J_5\tau_5^3}{6} + x_4$)
v.a($V - \frac{1}{2}J_5\tau_5^2$)
a.a($-J_5\tau_5$)
j.a(j[4])
No
ti<t[6] and ti≥t[5]?
Yes
p.a($v_5\tau_6 - \frac{D\tau_6^2}{2} + x_5$)
v.a($v_5 - D\tau_6$)
a.a($-D$)
j.a(j[5])
No
ti≥t[6]?
Yes
p.a($v_6\tau_7 - \frac{D\tau_7^2}{2} + \frac{J_7\tau_7^3}{6} + x_6$)
v.a($v_6 - D\tau_7 + \frac{1}{2}J_7\tau_7^2$)
a.a($-D + J_7\tau_7$)
j.a(j[6])

Figure 32. Flow diagram for program 5.1. S-curve calculation. p.a, v.a, a.a and j.a hold for position.append, velocity.append, acceleration.append and jerk.append, respectively.

C.2. Drawbar-pull, torque and power calculation

C.2.1. Execution Examples

Example 1

In this execution example, the values related to s-curve calculation are the same used in Example 1 of C.1.1. The parameters related to soil-wheel interaction are taken from [15], although the units are changed in order to fit the rest of the program. The input and output of the program is given in Table 8.

Table 8. Input and output values for program C.2 in Example 1

Input	a_0=0 $a_f = 0$ v_0=0 $v_f = 0$ x_0=0 x_f=50
	$A = 1$ $D = 1$ J_1=1 J_3=1 $J_5 = 1$ J_7=1
	div=1000 coefk=212.58*10^3 coefn=0.82
	coefm=-0.364 K=0.025 c=3000 s=0.1 psi=27°
	d=0.2 b=0.1 m=40

Output	See Figure 33.

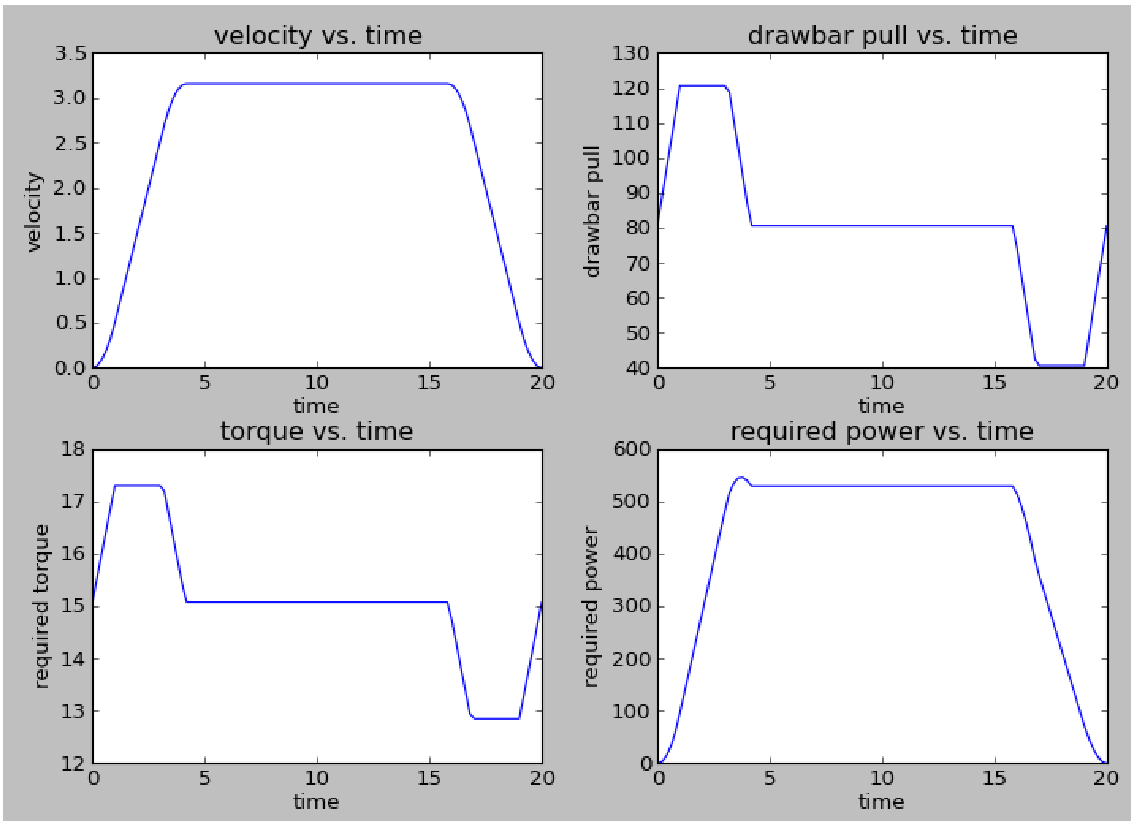

Figure 33. Output plot for Example 1.

Example 2

In this case, with reference to Example 1, a different s-curve profile is given, in the form of non-zero final acceleration value and different jerk values for phases 1,3 and phases 5,7. The input and output values of this execution are given in Table 9.

Table 9. Input and output values for program C.2 in Example 2

Input	a_0=0 $a_f = -0.5$ v_0=1 $v_f = 0$ x_0=0 x_f=50
	$A = 1$ $D = 1$ J_1=1 J_3=1 $J_5 = 0.5$ J_7=0.5
	div=1000 coefk=212.58*10^3 coefn=0.82
	coefm=-0.364 K=0.025 c=3000 s=0.1 psi=27°
	d=0.2 b=0.1 m=40

Output	See Figure 34.

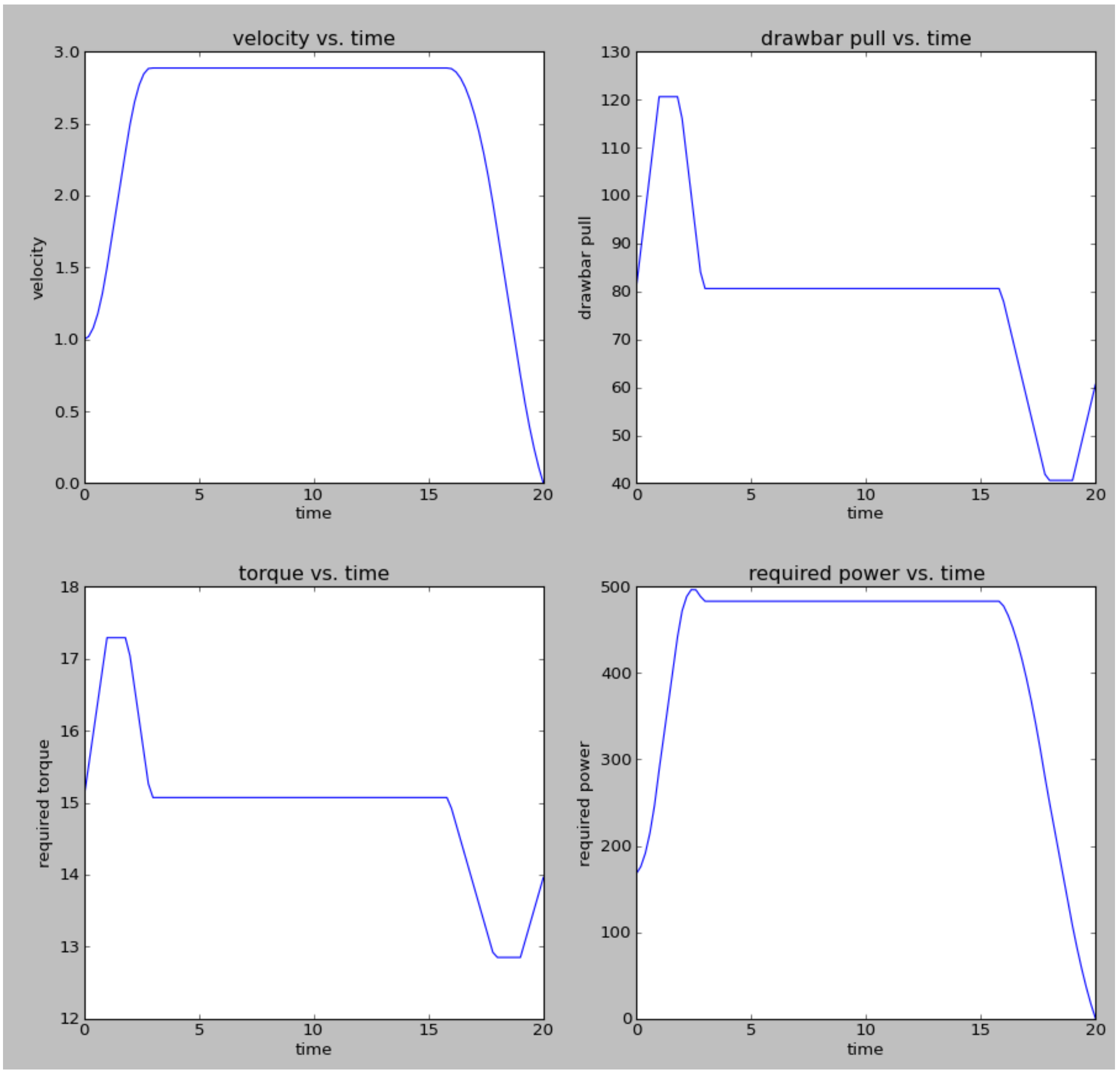

Figure 34. Output plot for Example 2.

Example 3

The difference between this execution and the one in Example 1is the value for the slip, which is set to 0.4 in this case. The whole set of inputs and outputs is shown in Table 10.

Table 10. Input and output values for program C.2 in Example 3

Input	a_0=0 $a_f = 0$ v_0=0 $v_f = 0$ x_0=0 x_f=50
	$A = 1$ $D = 1$ J_1=1 J_3=1 $J_5 = 1$ J_7=1
	div=1000 coefk=212.58*10^3 coefn=0.82
	coefm=-0.364 K=0.025 c=3000 s=0.4 psi=27°
	d=0.2 b=0.1 m=40

Output	See Figure 35. Output plot for Example 3..

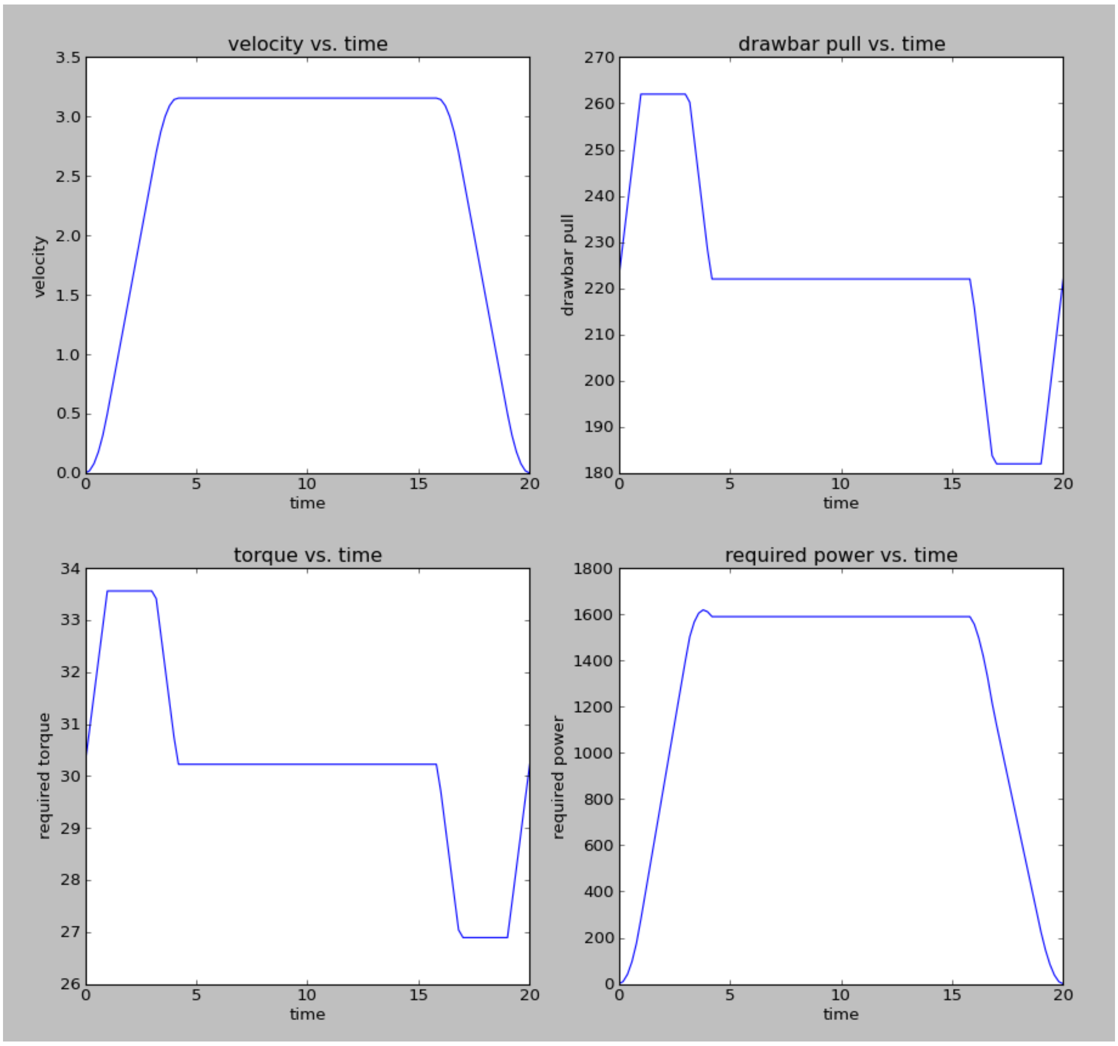

Figure 35. Output plot for Example 3.

Example 4

This execution has different wheel-dimension parameters than the one in Example 1. The wheel diameter has been changed from 0.2m to 0.4m and the wheel width has been modified from 0.1m to 0.2m. Table 11 shows the input and output values for this example.

Table 11. Input and output values for program C.2 in Example 4

Input	a_0=0 $a_f = 0$ v_0=0 $v_f = 0$ x_0=0 x_f=50
	$A = 1$ $D = 1$ J_1=1 J_3=1 $J_5 = 1$ J_7=1
	div=1000 coefk=212.58*10^3 coefn=0.82
	coefm=-0.364 K=0.025 c=3000 s=0.1 psi=27°
	d=0.4 b=0.2 m=40

Output	See Figure 36.

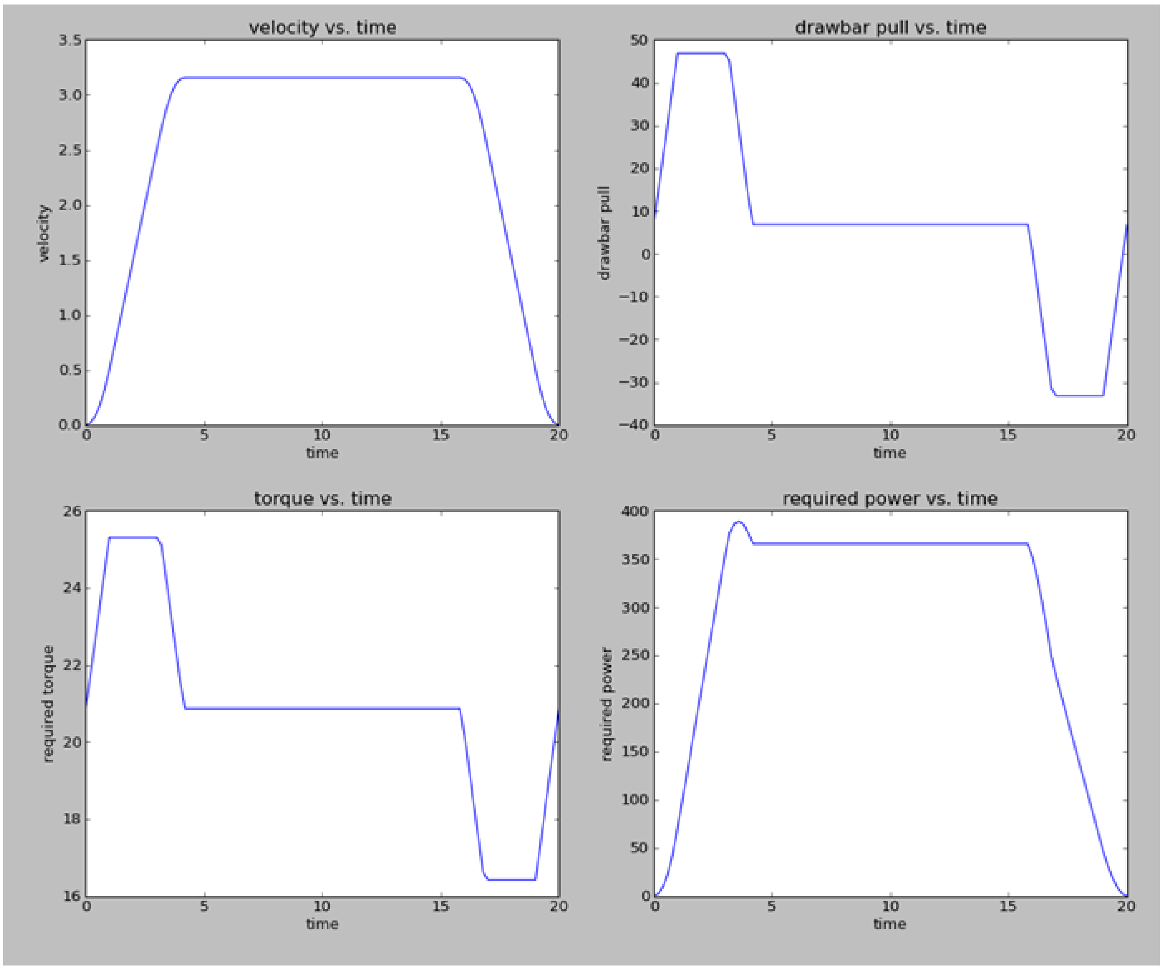

Figure 36. Output plot for Example 4.

Example 5

In this example, the parameter that has been changed from Example 1 is the mass of the load on the wheel, which on the previous execution examples was 40kg and in this one is 80kg. Table 12 includes the values of all of the inputs and outputs of this execution.

Table 12. Input and output values for program C.2 in Example 5

Input	a_0=0 $a_f = 0$ v_0=0 $v_f = 0$ x_0=0 x_f=50
	$A = 1$ $D = 1$ J_1=1 J_3=1 $J_5 = 1$ J_7=1
	div=1000 coefk=212.58*10^3 coefn=0.82
	coefm=-0.364 K=0.025 c=3000 s=0.1 psi=27°
	d=0.2 b=0.1 m=80

Output	See Figure 37.

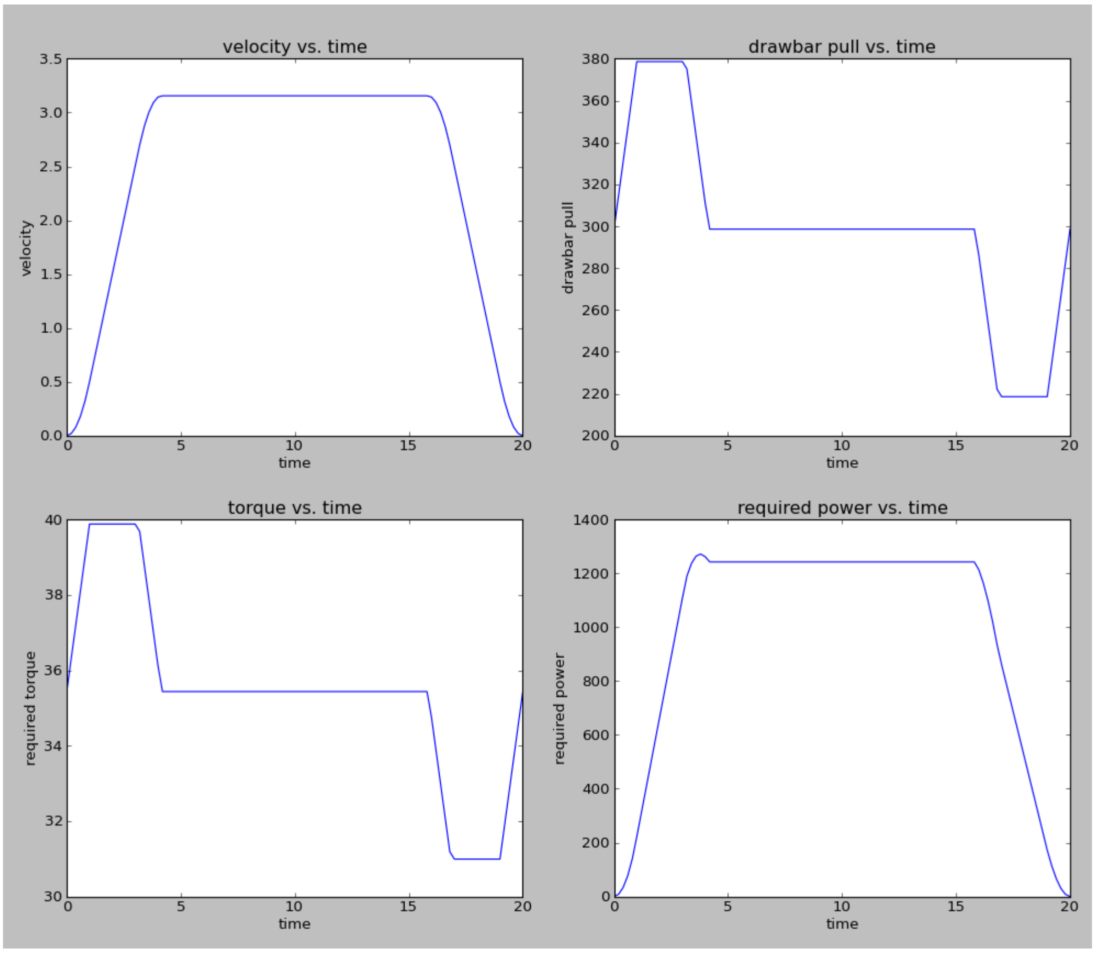

Figure 37. Output plot for Example 5

C.2.2. Flow diagram

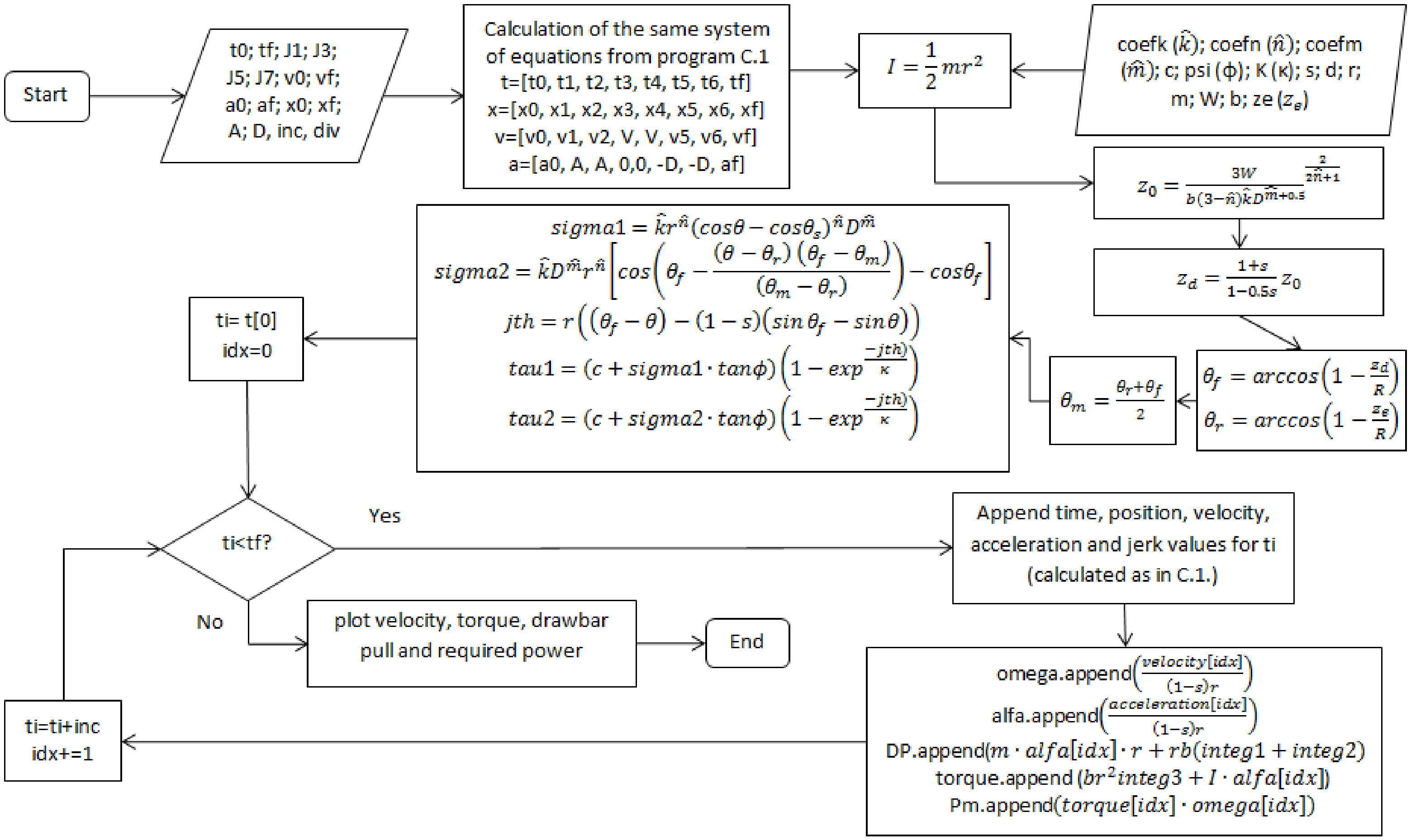
Start
t0; tf; J1; J3; J5; J7; v0; vf; a0; af; x0; xf; A; D, inc, div
Calculation of the same system of equations from program C.1
t=[t0, t1, t2, t3, t4, t5, t6, tf]
x=[x0, x1, x2, x3, x4, x5, x6, xf]
v=[v0, v1, v2, V, V, v5, v6, vf]
a=[a0, A, A, 0,0, -D, -D, af]
$I = \frac{1}{2}mr^2$
coefk ($\hat{k}$); coefn ($\hat{n}$); coefm ($\hat{m}$); c; psi (ϕ); K (κ); s; d; r; m; W; b; ze (z_e)
$z_0 = \frac{3W}{b(3-\hat{n})\hat{k}D^{\hat{m}+0.5}}^{\frac{2}{2\hat{n}+1}}$
$z_d = \frac{1+s}{1-0.5s} z_0$
$\theta_f = arccos\left(1 - \frac{z_d}{R}\right)$
$\theta_r = arccos\left(1 - \frac{z_e}{R}\right)$
$\theta_m = \frac{\theta_r+\theta_f}{2}$
$sigma1 = \hat{k}r^{\hat{n}}(cos\theta - cos\theta_s)^{\hat{n}}D^{\hat{m}}$
$sigma2 = \hat{k}D^{\hat{m}}r^{\hat{n}}\left[cos\left(\theta_f - \frac{(\theta-\theta_r)(\theta_f-\theta_m)}{(\theta_m-\theta_r)}\right) - cos\theta_f\right]$
$jth = r\left((\theta_f - \theta) - (1-s)(sin\,\theta_f - sin\theta)\right)$
$tau1 = (c + sigma1 \cdot tan\phi)\left(1 - exp^{\frac{-jth)}{\kappa}}\right)$
$tau2 = (c + sigma2 \cdot tan\phi)\left(1 - exp^{\frac{-jth)}{\kappa}}\right)$
ti= t[0]
idx=0
ti<tf?
Yes
No
Append time, position, velocity, acceleration and jerk values for ti (calculated as in C.1.)
omega.append$\left(\frac{velocity[idx]}{(1-s)r}\right)$
alfa.append$\left(\frac{acceleration[idx]}{(1-s)r}\right)$
DP.append($m \cdot alfa[idx] \cdot r + rb(integ1 + integ2)$
torque.append ($br^2 integ3 + I \cdot alfa[idx]$)
Pm.append($torque[idx] \cdot omega[idx]$)
ti=ti+inc
idx+=1
plot velocity, torque, drawbar pull and required power
End

Figure 38. Flow diagram for program 5.2. Drawbar pull, torque and power calculation.

C.2.3. Integra function: flow diagram

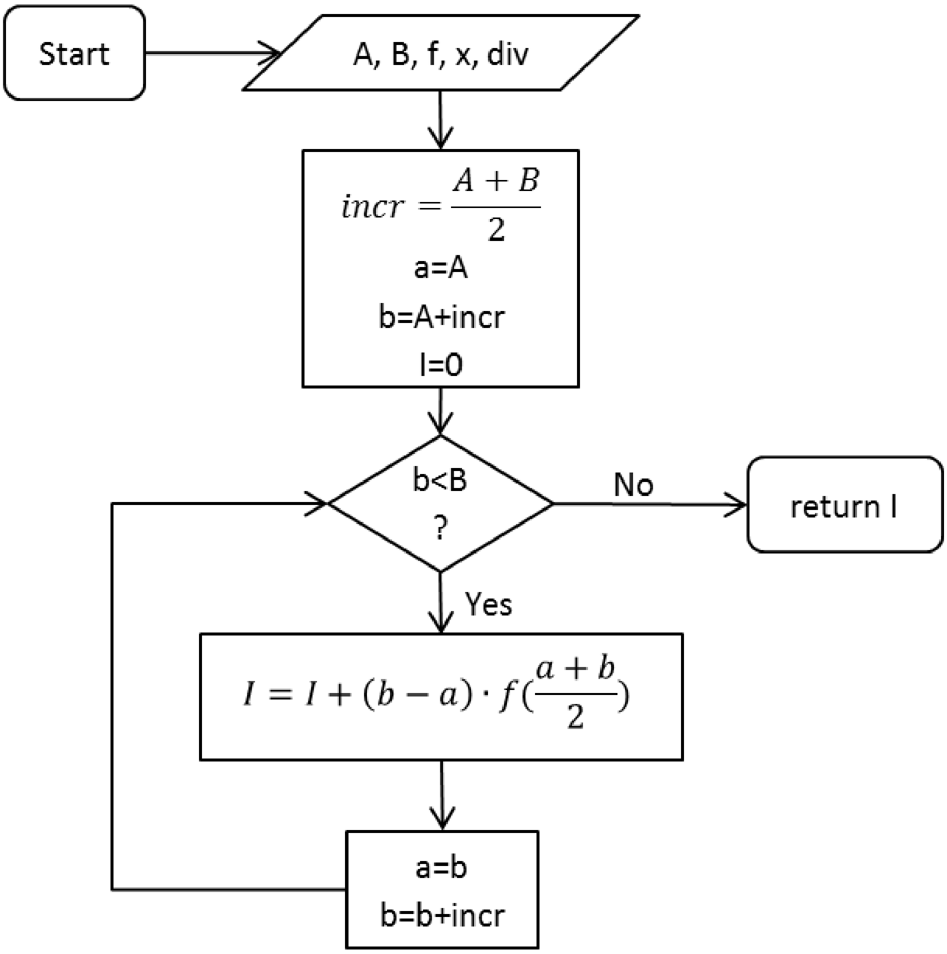

Figure 39. Flow diagram for function *integra*

C.2.4.Effect of wheel and wheel-soil interaction parameters

In order to plot the effect of the different parameters involved in the wheel-soil interaction system, slight modifications on the program in 5.2. Drawbar pull, torque and power calculation have been necessary, mainly changing the variable under study into a Python list and adding a loop so the calculation of DP, torque and power is done for each of the values.

Execution examples

Effect of slip value

In order to study the effect of the slip value in the output values for required drawbar pull, torque and power, these have been calculated for s between 0 and 0.74. the output graph is shown in Figure 40.

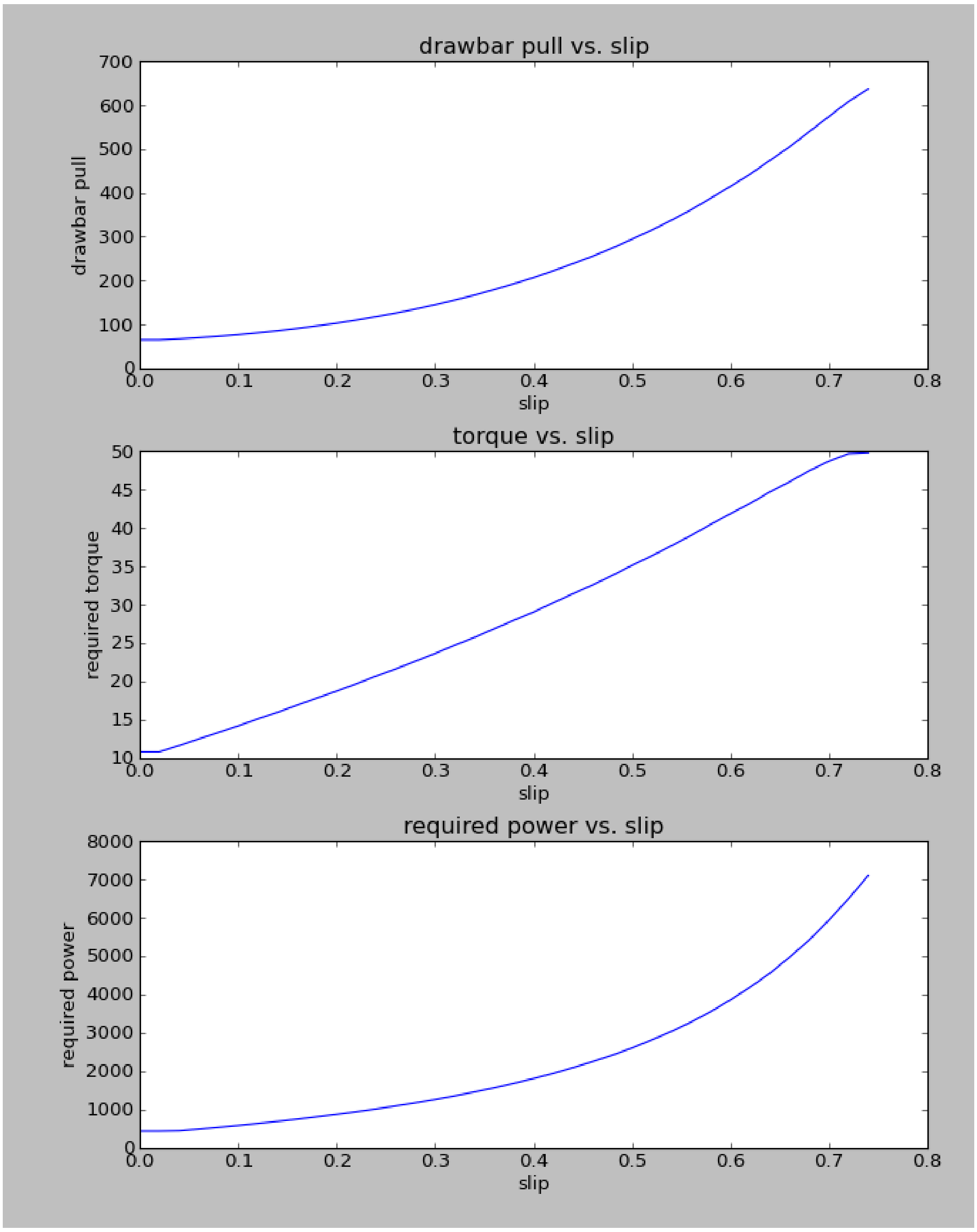

Figure 40. Effect of the value of the slip parameter on the drawbar pull, required torque and required power.

Effect of wheel diameter

With the aim of computing the effect of the diameter of the wheel on the drawbar pull, torque and power, these have been calculated for d between 0.1 and 1.6. Figure 41 shows the output plots of the program.

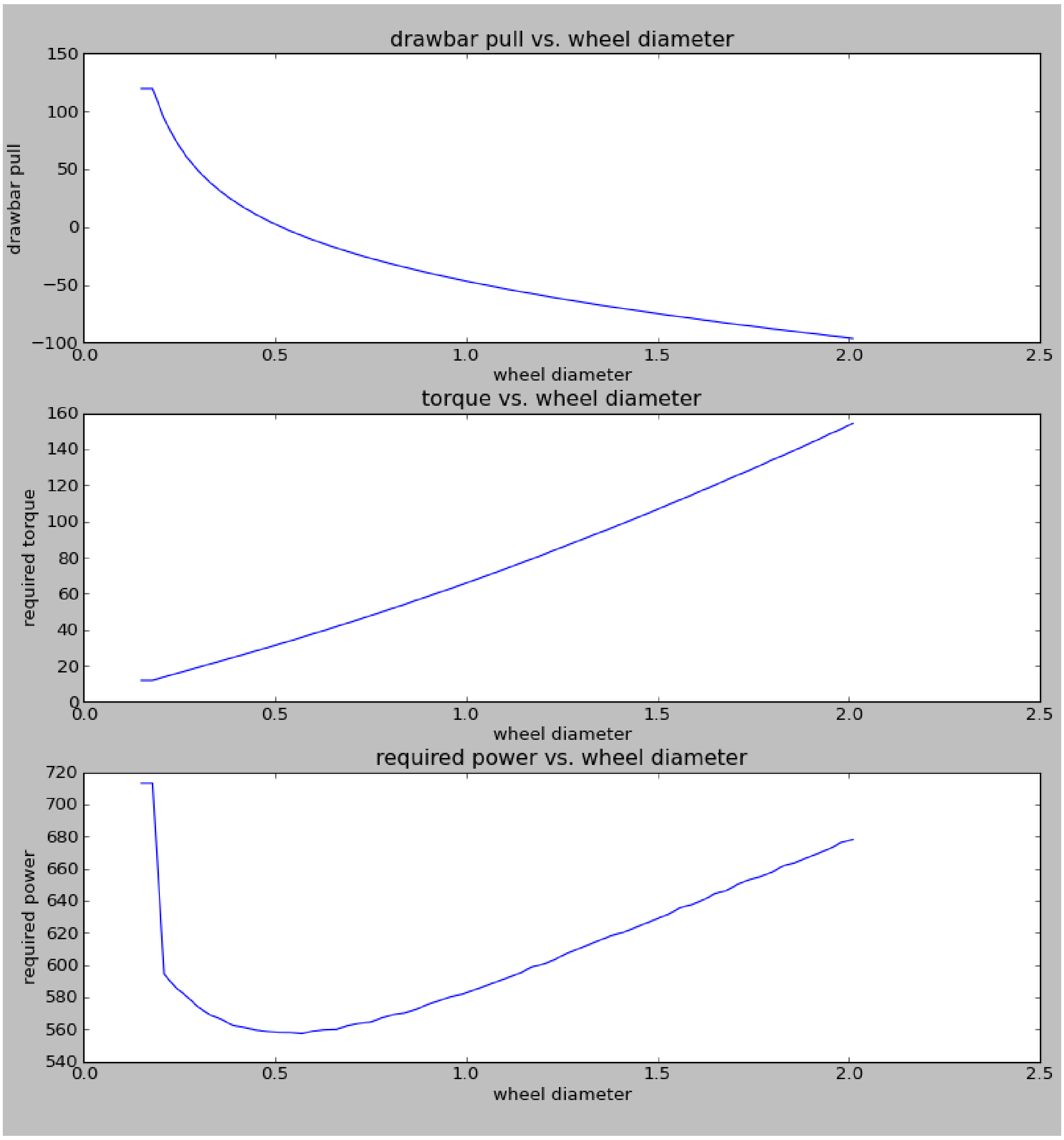

Figure 41. Effect of the value of the diameter of the wheel on the drawbar pull, required torque and required power.

Effect of parameter c

Parameter c stands for the cohesion of the soil. Figure 42 helps analyse its effect on the performance of the wheel.

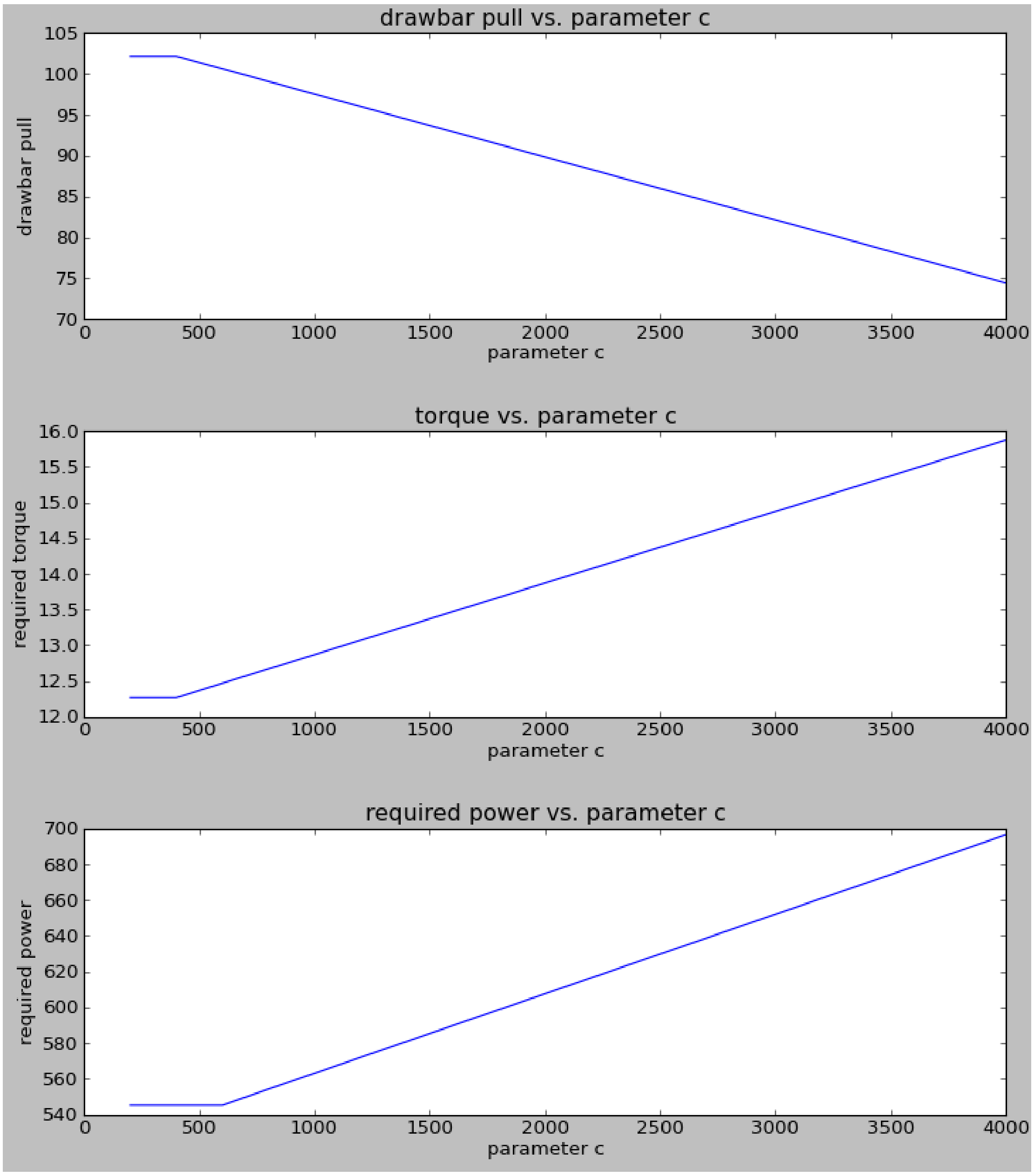

Figure 42. Effect of parameter c on the required drawbar pull, torque and power.

Effect of parameter $\hat{k}$

Parameter $\hat{k}$ is one of the constants of the proposed model that affect the most on the required drawbar pull, torque and power. Its effects on the calculation are shown in Figure 43.

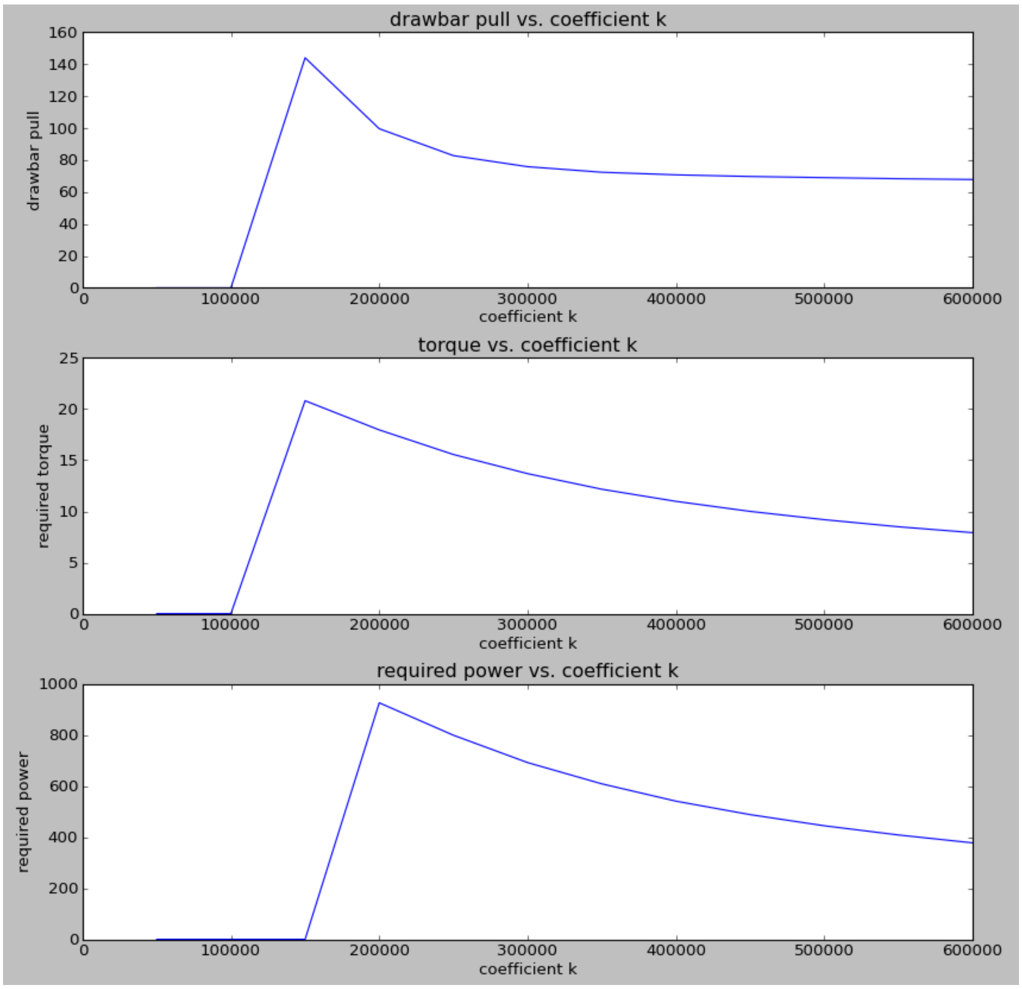

Figure 43. Effect of $\hat{k}$ on the required drawbar pull, torque and power.

C.3. Unicycle robot trajectory

C.3.1. Execution Examples

Example 1

Inputs

vr=[0,1,2,3,4,4.5,4.7,5,5,5,5,5,5,5,5,5,5,4,3,2,1,1,1,1,0.5,0.25,0]
vl=[0,1,2,3,4,4,4,3.5,3.5,3.5,3.5,4,5,5,5,5,5,5,5,4,2,1,2,2,1,0,0]
t=[0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23, 24,25,26]
L=1

Outputs See Figure 44

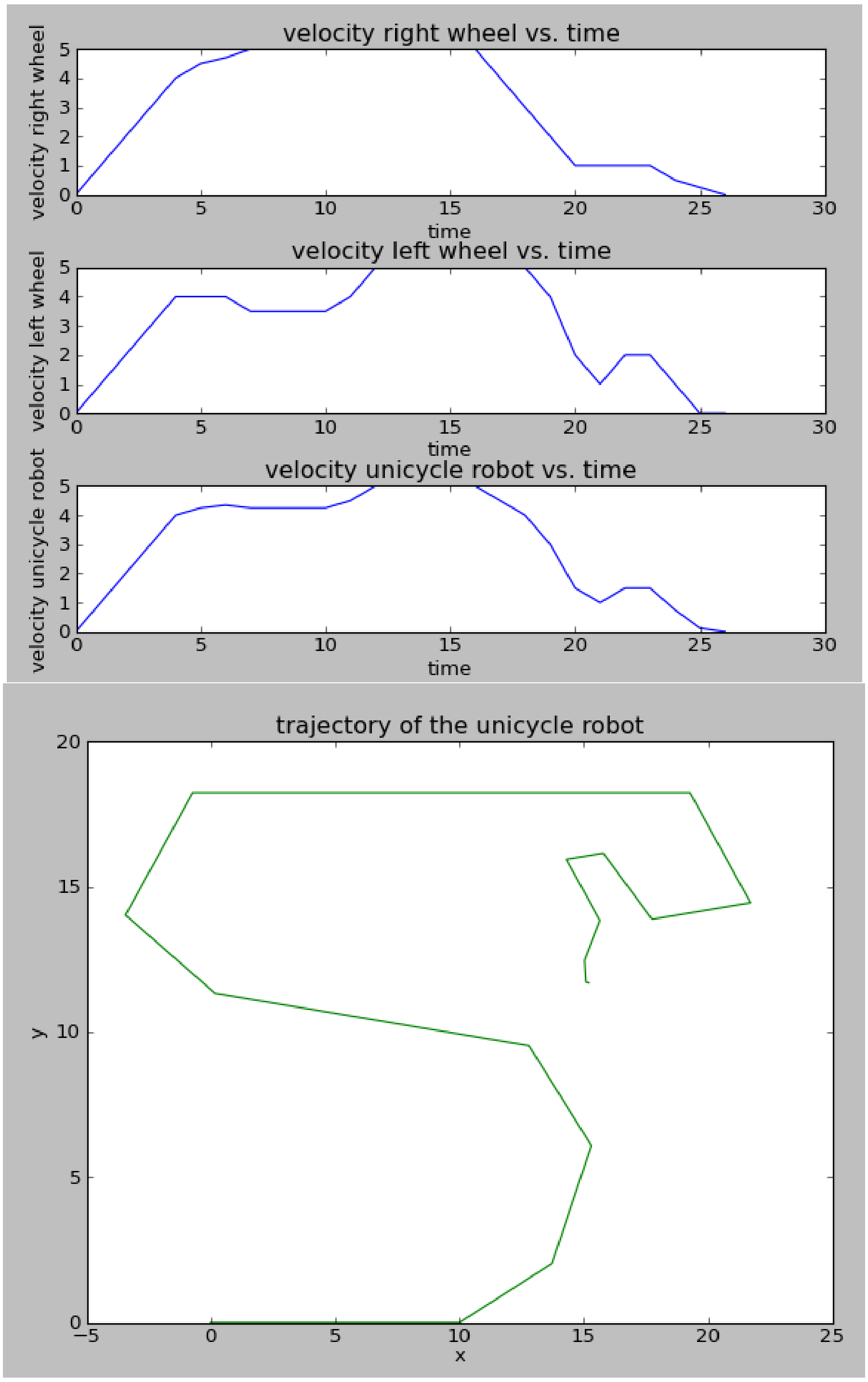

Figure 44. Output graphs for Example 1.

Example 2

The only difference between Example 1 and Example 2 is the value of L, which is the distance between the wheels of the robot. The input velocity profiles for the wheels are the same.

Inputs vr=[0,1,2,3,4,4.5,4.7,5,5,5,5,5,5,5,5,5,5,5,4,3,2,1,1,1,1,0.5,0.25,0]
vl=[0,1,2,3,4,4,4,3.5,3.5,3.5,3.5,4,5,5,5,5,5,5,5,5,4,2,1,2,2,1,0,0]
t=[0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,
24,25,26]
L=0.5

Outputs See Figure 45

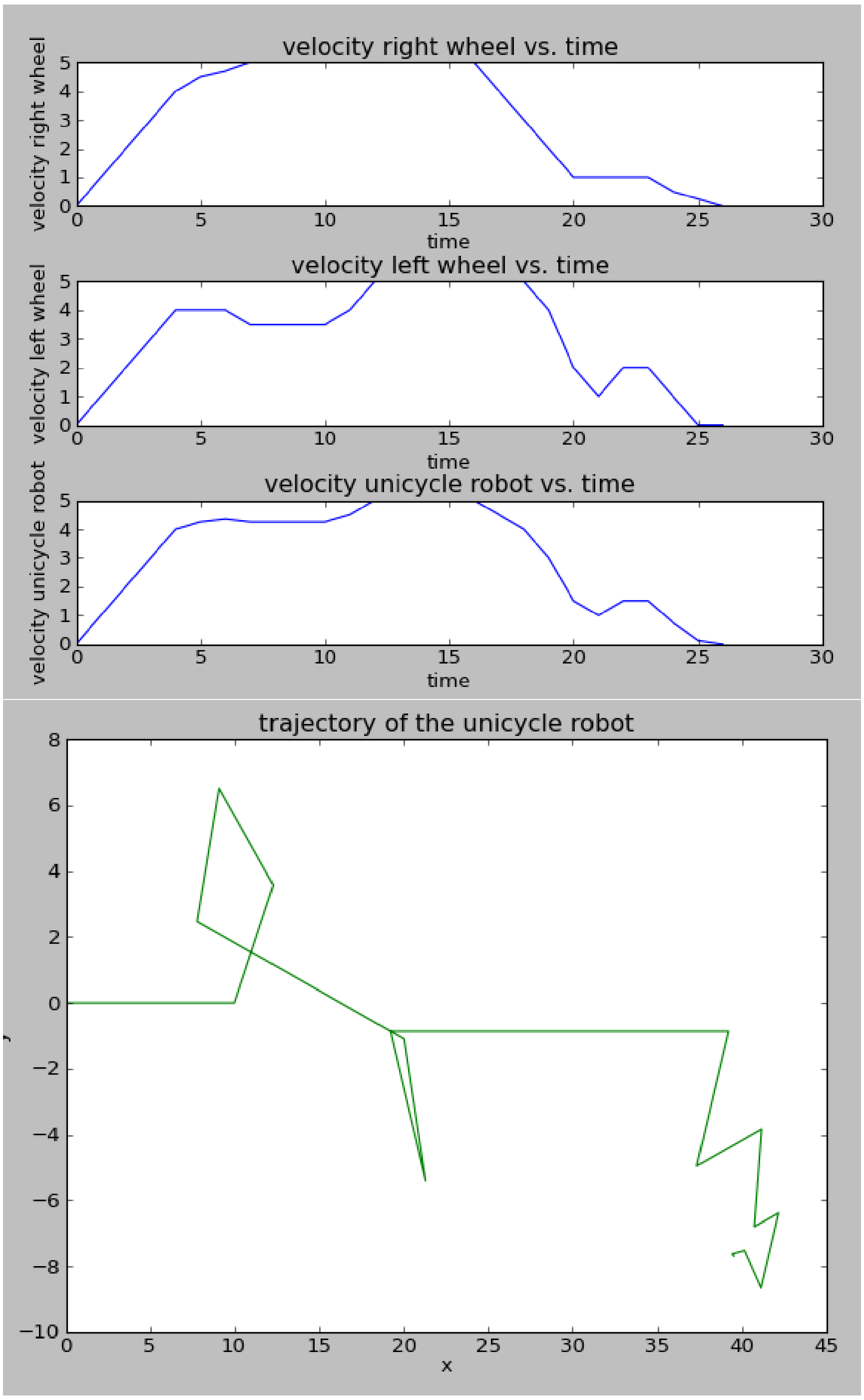

Figure 45. Output graphs for Example 2.

C.3.2. Flow diagram

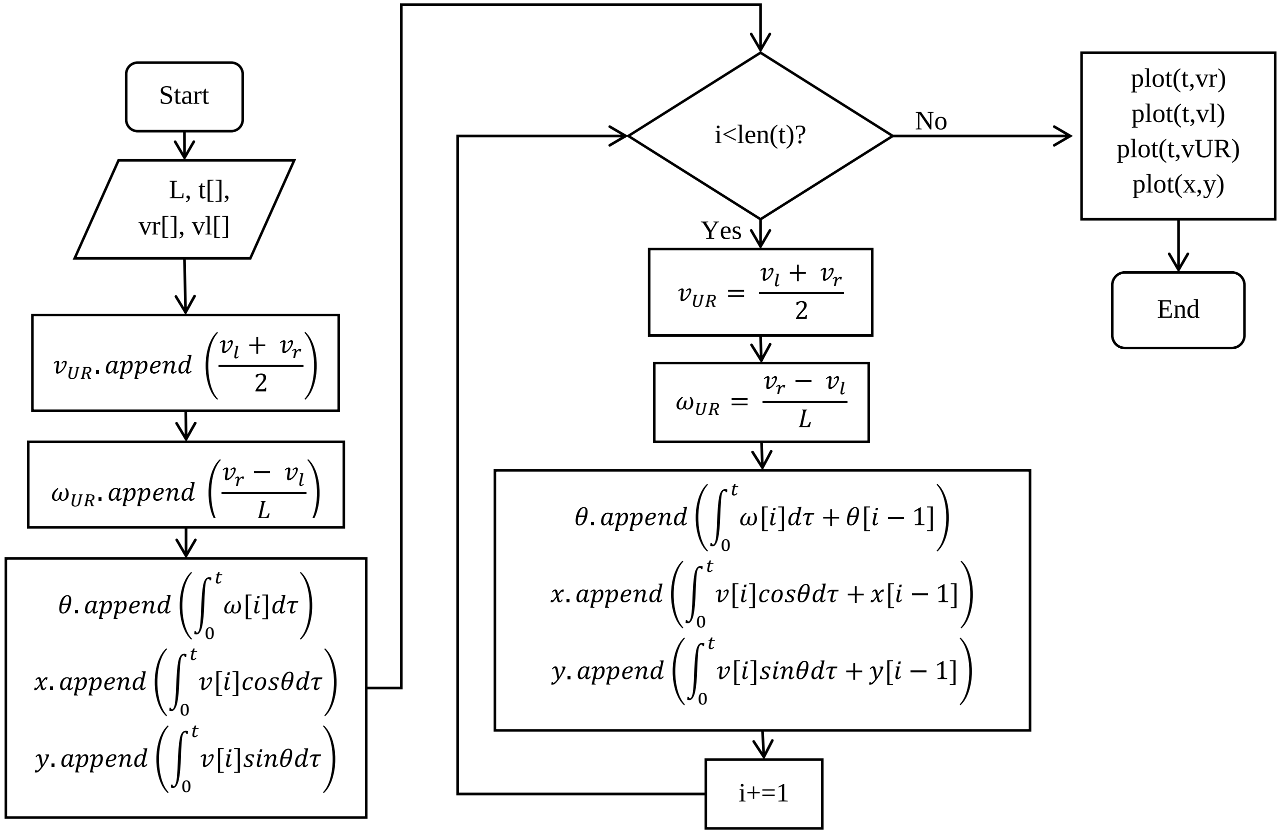
Start
L, t[], vr[], vl[]
$v_{UR}.append\left(\frac{v_l + v_r}{2}\right)$
$\omega_{UR}.append\left(\frac{v_r - v_l}{L}\right)$
$\theta.append\left(\int_0^t \omega[i]d\tau\right)$
$x.append\left(\int_0^t v[i]cos\theta d\tau\right)$
$y.append\left(\int_0^t v[i]sin\theta d\tau\right)$
i<len(t)?
No
Yes
$v_{UR} = \frac{v_l + v_r}{2}$
$\omega_{UR} = \frac{v_r - v_l}{L}$
$\theta.append\left(\int_0^t \omega[i]d\tau + \theta[i-1]\right)$
$x.append\left(\int_0^t v[i]cos\theta d\tau + x[i-1]\right)$
$y.append\left(\int_0^t v[i]sin\theta d\tau + y[i-1]\right)$
i+=1
plot(t,vr)
plot(t,vl)
plot(t,vUR)
plot(x,y)
End

Figure 46. Flow diagram for program 5.3. Unicycle robot trajectory.