

1

Mobile Robot Kinematics

Mobile Robot Kinematics

Motion planning and control are fundamental components of robotic autonomy¹. For example, in order for an autonomous car to accomplish an objective (e.g. move from point A to B) it first needs to plan a trajectory and determine what control inputs (e.g. throttle and steering) will enable it to follow the trajectory. Both of these components require an understanding of the physical behavior of the robot in order to develop reasonable/actionable plans and controls. In the context of motion planning and control, a robot's physical behavior is generally characterized by its *dynamics* and *kinematics*.

Definition 1.0.1 (Dynamics). *A robot's dynamics describe the relationship between forces acting on the robot and changes to the robot's physical state.*

In other words, dynamics can be thought of as the result of Newton's Second Law ($F = ma$) in the context of a particular robot. For example, the dynamics of an autonomous car would describe the relationship between acceleration and forces induced by the tires, gravity, aerodynamics, and so on.

Definition 1.0.2 (Kinematics). *A robot's kinematics describe additional restrictions (constraints) on the robot's motion that are not induced by forces.*

The most trivial example is that the rate of change of the robot's position must equal its velocity. More generally a robot's kinematics describe limitations on its motion that are a function of the robot's physical state or geometry. For example a robotic arm with multiple joints is kinematically constrained since the rigid connections at each joint only allow rotation about a single axis.

From the preceding descriptions it should be noted that a robot's dynamics and kinematics describe limitations on its motion in different ways². Not only is it important to identify and describe a robot's dynamics and kinematics, but a roboticist should also ask:

1. Do I need to consider *all* of the dynamics/kinematics? Are they all important to the robot's task?

¹ R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza. *Introduction to Autonomous Mobile Robots*. MIT Press, 2011

² One simple heuristic for determining how a particular constraint/relationship should be classified is to remember that dynamics are affected by changing the robot's mass, while kinematics are *not*.

2. Can any of the dynamics/kinematics be simplified/approximated to make the motion planning and control task easier?

The combination of dynamics and kinematics make up a *model* of the physical behavior of the robot, and depending on the robot's task some models may be more appropriate than others. In particular the complexity of the model often needs to be balanced with its accuracy/relevance for the task at hand.

To illustrate this, consider again the autonomous car example. The most accurate model would leverage the car's dynamics, and would include engine dynamics, suspension dynamics, tire dynamics, and so on. In particular, incorporating tire dynamics is critical to understand the phenomenon of drifting, which may be important for motion planning and control in autonomous racing applications. However in other applications it may be more appropriate to simplify the model by replacing the tire *dynamics* with a simple *kinematic* constraint that the tires cannot move laterally (i.e. a "no side slip" constraint).

In fact, in the context of motion planning and control for robotics, models built entirely from kinematics can be very useful (and much simpler). For this reason this chapter specifically focuses on robot kinematics, and in particular:

1. How to express the configuration of a robot in terms of *general coordinates*
2. How to mathematically express kinematic constraints in terms of general coordinates
3. How to identify different types of kinematic constraints, namely *holonomic* and *nonholonomic* constraints
4. Examples of kinematic models, specifically for wheeled robots

1.1 Generalized Coordinates

A robot's physical state (also commonly referred to as its "configuration") can usually be represented (i.e. quantified) in different ways. The particular choice of representation defines a finite set of numbers known as *generalized coordinates*.

Definition 1.1.1 (Generalized Coordinates). *Generalized coordinates refer to a set of coordinates that can completely specify the unique position of your robot.*

For example, the wheel rolling on a plane in Figure 1.1 can be represented by three parameters, x, y , and θ , where (x, y) indicates the position at which the wheel touches the ground, and θ indicates the direction the wheel is traveling in the general frame. This set of parameters (x, y, θ) that define the wheel's configuration are generalized coordinates for this system. Note that in practice people often use "configuration" and "generalized coordinates" interchangeably, even though the specific choice of generalized coordinates are not necessarily the only possible representation of the robot's configuration.

The generalized coordinates are mathematically expressed by the vector $\xi \in \mathbb{R}^n$, where n is the number of generalized coordinates used to describe the

robot's configuration. A robot's motion through time (i.e. its trajectory) is then expressed by the function

$$\xi(t) : \mathbb{R} \rightarrow \mathbb{R}^n,$$

where t denotes time. In the case of the wheel in Figure 1.1 the generalized coordinate vector would be $\xi = [x \ y \ \theta]^\top$.

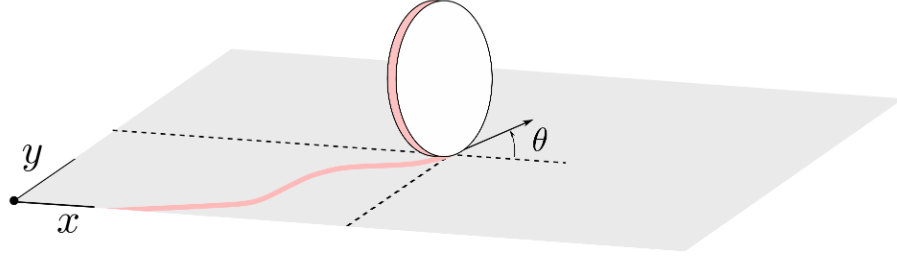


Figure 1.1: Generalized coordinates for a wheel rolling without slipping on a plane.

1.2 Kinematic Constraints

Once a set of generalized coordinates ξ has been identified, they can be used to mathematically define kinematic constraints that define a robot's motion. A more formal definition of general kinematic constraints is first presented:

Definition 1.2.1 (Kinematic Constraints). *Let the generalized coordinates of a robot be denoted as $\xi = [\xi_1, \dots, \xi_n]^\top$. Constraints that depend on these generalized coordinates and their velocities are called kinematic constraints and are expressed as*

$$a_i(\xi, \dot{\xi}) = 0, \quad i = 1, \dots, k < n \quad (1.1)$$

where $\dot{\xi} = \frac{d\xi}{dt}$ are the velocities.

Kinematic constraints in robotics applications are often linear with respect to the generalized velocities. Constraints of this kind are referred to as *Pfaffian constraints* and are expressed as

$$a_i^\top(\xi) \dot{\xi} = 0, \quad i = 1, \dots, k < n \quad (1.2)$$

where $a_i(\xi) \in \mathbb{R}^n$. For notational simplicity these constraints can be compactly expressed in matrix form as

$$A^\top(\xi) \dot{\xi} = 0, \quad (1.3)$$

where $A(\xi) \in \mathbb{R}^{n \times k}$.

Example 1.2.1 (Pendulum). Figure 1.2 shows a simple pendulum that is assumed to rotate about a fixed pivot point. Let the position of the mass be given by the Cartesian coordinates (x, y) , which can be used as the generalized coordinates for this system (i.e. $\xi = [x, y]^\top$). Since the rod connecting the pivot point

to the mass is assumed to be rigid this implies a kinematic constraint. Assuming the pivot point is at the origin $(0,0)$ this constraint can be expressed as

$$a_1(\xi) = x^2 + y^2 - L^2 = 0, \quad (1.4)$$

where L is the length of the rod. Note that while this does not appear to be a Pfaffian constraint, it can be equivalently expressed as one. In particular, consider the derivative of the expression with respect to time, which yields the Pfaffian constraint

$$\frac{da_1(\xi)}{dt} = \frac{da_1(\xi)}{d\xi} \dot{\xi} = 2x\dot{x} + 2y\dot{y} = 0, \quad (1.5)$$

In this particular case a more natural choice of coordinates would simply be $\xi = [\theta]$, which also fully specifies the system's configuration and eliminates the need to enforce additional kinematic constraints. In fact, it can be noted that since $x = L \sin \theta$ and $y = -L \cos \theta$ that the above kinematic constraint is trivially satisfied for all θ .

Example 1.2.2 (No-Slip Wheel). Consider again the wheel illustrated in Figure 1.1 with generalized coordinates $\xi = [x, y, \theta]^\top$, and assume that there is a no-slip condition between the wheel and the plane it rolls on. This no-slip condition means that the velocity component of the wheel in the lateral direction is always zero. Since the heading of the wheel is given by the unit vector $e_v = [\cos \theta, \sin \theta]^\top$, the lateral direction can be described by the perpendicular unit vector $e_{v,\perp} = [\sin \theta, -\cos \theta]^\top$.

Since the velocity vector is $v = [\dot{x}, \dot{y}]^\top$, the no-slip kinematic constraint can be expressed by the inner product $v \cdot e_{v,\perp} = 0$, which is equivalently expressed as

$$a_1(\xi, \dot{\xi}) = \dot{x} \sin \theta - \dot{y} \cos \theta = 0. \quad (1.6)$$

Note that this constraint is linear in the generalized velocities (\dot{x}, \dot{y}) and therefore is a Pfaffian constraint.

1.3 Holonomic and Nonholonomic Constraints

It is useful to further classify different types of kinematic constraints based on how they restrict the motion of the system. In particular, the most common classifications for kinematic constraints are *holonomic* or *nonholonomic*.

1.3.1 Holonomic Constraints

Holonomic constraints are kinematic constraints that can be expressed as a function of *only* the generalized coordinates (without dependence on generalized velocities). In robotics applications, holonomic constraints generally arise due to mechanical interconnections, such as rigid links and joints of a robotic arm.

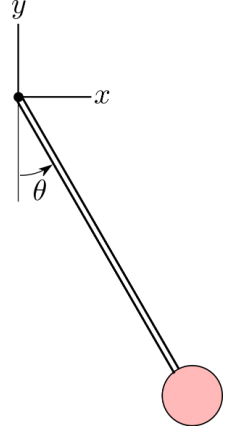


Figure 1.2: Generalized coordinates for a simple pendulum.

Definition 1.3.1 (Holonomic Constraints). *Constraints that can be expressed in the form*

$$h_i(\xi) = 0, \quad i = 1, \dots, k < n \quad (1.7)$$

are called holonomic.

Additionally, a *holonomic system* is a system that is only subject to holonomic constraints. Note that these constraints can *always* be equivalently expressed as Pfaffian constraints of the form (1.2) by differentiating the expression

$$\frac{dh_i(\xi)}{dt} = \frac{dh_i(\xi)}{d\xi} \dot{\xi} = a_i^\top(\xi) \dot{\xi} = 0, \quad i = 1, \dots, k < n \quad (1.8)$$

However, it is important to note that not all Pfaffian constraints are holonomic. A Pfaffian constraint is only holonomic if it is *integrable* to the form (1.7).

Holonomic constraints are a unique subclass of kinematic constraints that *restrict the accessible configurations of the system*. In fact, the space of accessible configurations for a system with n generalized coordinates under k holonomic constraints will have dimension $n - k$.

Examples: Consider again the pendulum from Example 1.2.1, where the kinematic constraint (1.4) can be expressed as $h_i(\xi) = 0$ (equivalently where the Pfaffian constraint (1.5) is integrable into the form $h_i(\xi) = 0$). This constraint restricts the pendulum mass to lie on a circle of radius L , which is a one dimensional subset ($n - k = 2 - 1 = 1$).

Alternatively, consider the wheel from Example 1.2.2, where the kinematic constraint (1.6) *cannot* be integrated to yield a constraint of the form $h_i(\xi) = 0$. In contrast to the pendulum, this system has no restriction on what configuration it can be in as it can potentially move to any point (x, y) .

1.3.2 Nonholonomic Constraints

While holonomic constraints are kinematic constraints which restrict the accessible configurations of the system, not all kinematic constraints are holonomic. In particular, it is possible to have kinematic constraints that *do not* restrict accessible configurations, but rather restrict the motion *between* configurations. These constraints are referred to as *nonholonomic* constraints.

Definition 1.3.2 (Nonholonomic Constraints). *Constraints that can be described in Pfaffian form, but cannot be integrated to $h_i(\xi) = 0$ form are called nonholonomic.*

Additionally, a *nonholonomic system* is a system that is subject to at least one nonholonomic constraint. The restriction of instantaneous motion that is induced by a nonholonomic constraint can be interpreted by considering the Pfaffian form $a_i(\xi)^\top \dot{\xi} = 0$. It is clear that for any coordinate ξ , this constraint limits the motion $(\dot{\xi})$ to lie in the null space of $a_i(\xi)^\top$.

Examples: Consider again the wheel example from Example 1.2.2 which has a nonholonomic constraint

$$a_i(\xi)^\top \dot{\xi} = \begin{bmatrix} \sin \theta & -\cos \theta & 0 \end{bmatrix} \dot{\xi} = 0.$$

The null space of $a_i(\xi)^\top$ in this case is spanned by the vectors $[\cos \theta, \sin \theta, 0]$ and $[0, 0, 1]$ which suggests that any potential motion must be made up of a linear combination of these vectors. Intuitively this would be expected because $[\cos \theta, \sin \theta, 0]$ is the unit vector in the direction of rolling, and $[0, 0, 1]$ would correspond to the wheel spinning but not rolling.

1.4 Kinematic Models

Once an appropriate set of generalized coordinates $\xi \in \mathbb{R}^n$ and all relevant kinematic constraints have been identified for a particular robot the next step is to develop a kinematic model. In particular these kinematic models will consist of a set of differential equations of the form $\dot{\xi}(t) = G(\xi(t))u(t)$, where $u(t)$ is referred to as a system input or *control*. Given a particular input $u(t)$ and an initial condition $\xi(0)$ this model will define a trajectory of the system.

Definition 1.4.1 (Kinematic Model). *Given a generalized coordinate vector $\xi \in \mathbb{R}^n$ and k Pfaffian kinematic constraints $A^\top(\xi)\dot{\xi} = 0$, a kinematic model can be defined as $\dot{\xi} = G(\xi)u$ where the column space of $G(\xi) \in \mathbb{R}^{n \times n-k}$ spans the null space of $A^\top(\xi)$. Additionally, for any input u the solutions to the kinematic model are guaranteed to satisfy the Pfaffian constraints.*

Consider k Pfaffian constraints written in matrix form as $A^\top(\xi)\dot{\xi} = 0$ (which can be a combination of holonomic and nonholonomic constraints). As was noted earlier these constraints imply that a generalized velocity $\dot{\xi}$ is only admissible at a configuration ξ if it lies in the $n - k$ dimensional null space of the matrix $A^\top(\xi)$. A new matrix, $G(\xi) \in \mathbb{R}^{n \times n-k}$ can therefore be defined such that the columns of $G(\xi)$ span the null space of $A^\top(\xi)$. In other words, for each column g_i of G it holds that $A^\top(\xi)g_i = 0$. To ensure that the generalized velocity $\dot{\xi}$ satisfies the kinematics constraints it is therefore sufficient to require that $\dot{\xi} = G(\xi)u$ where $u \in \mathbb{R}^{n-k}$ can be *any* vector. To explicitly show why this is true, consider any vector u and write $\dot{\xi} = G(\xi)u = \sum_{i=1}^{n-k} g_i(\xi)u_i$. When this is substituted into the Pfaffian constraints the expression becomes

$$\begin{aligned} A^\top(\xi)\dot{\xi} &= A^\top(\xi) \left(\sum_{i=1}^{n-k} g_i(\xi)u_i \right), \\ &= \sum_{i=1}^{n-k} A^\top(\xi)g_i(\xi)u_i, \\ &= 0, \end{aligned}$$

which shows that the kinematic constraints are satisfied.

Examples: Consider again the wheel example from Example 1.2.2 which has a single nonholonomic constraint

$$a_i(\xi)^\top \dot{\xi} = \begin{bmatrix} \sin \theta & -\cos \theta & 0 \end{bmatrix} \dot{\xi} = 0,$$

where $\xi = [x, y, \theta]^\top$. The null space of $a_i(\xi)^\top$ in this case is spanned by the vectors $[\cos \theta, \sin \theta, 0]$ and $[0, 0, 1]$ and therefore the kinematic model is given by

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}. \quad (1.9)$$

Note that in many cases the control inputs u_1 and u_2 also have an intuitive physical meaning. In this problem u_1 is the speed at which the wheel is moving, and u_2 is the angular rate at which it rotates.

1.5 Kinematic Models of Wheeled Robots

Robots come in all shapes, sizes, and configurations and with varying forms of mobility. However, wheeled robots are perhaps the most widely used because of their high mobility and simple design. For this reason several standard kinematic models for different wheeled robot configurations will now be given.

1.5.1 Unicycle Model

The unicycle model of a robot is the simplest kinematic model, and assumes that the robot can be approximated by a single wheel. In this case the kinematic constraints are exactly the same as the wheel rolling on a plane discussed previously in Example 1.2.2. A simplified diagram showing the generalized coordinates of this model is given in Figure 1.3, and the kinematic model is the same as (1.9):

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}, \quad (1.10)$$

where v is the forward speed of the unicycle and ω is the rate of rotation.

The advantage of the unicycle model lies in its simplicity and its ability to capture one of the most fundamental behaviors of wheeled robots. Such a model might be suited for higher level motion planning tasks, such as planning geometric paths to get a robot from point A to point B. Often times such a model might be complemented with models of higher fidelity (e.g. dynamics models) for performing lower level tasks such as control or for refining motion plans created by the unicycle model.

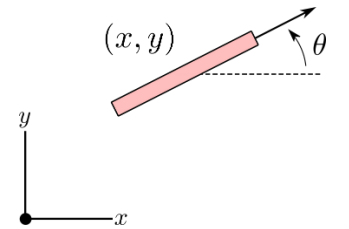


Figure 1.3: Generalized coordinates for a unicycle.

1.5.2 Differential Drive Model

The differential drive model is a slight variation on the unicycle model (see Section 1.5.1) that does not lump all of the wheels together. Instead, this model assumes two wheels are fixed on a rear shared axle, with a passive wheel that induces no kinematic constraints in the front. As shown in Figure 1.4 this model has the same generalized coordinates as the unicycle model ($\xi = [x, y, \theta]^\top$) but also includes some geometry of the robot by assuming the width of the rear axle is denoted by L .

Same as for the unicycle model, this model assumes the wheels roll without slipping. The derivation of the kinematic constraints is therefore similar to Example 1.2.2. In particular, the heading of each wheel is always given by $e_v = [\cos \theta, \sin \theta]^\top$, the lateral direction is given by $e_{v,\perp} = [\sin \theta, -\cos \theta]^\top$, and thus the two no-slip kinematic constraints can be expressed as

$$\dot{p}_l \cdot e_{v,\perp} = 0, \quad \dot{p}_r \cdot e_{v,\perp} = 0,$$

where \dot{p}_l and \dot{p}_r are the left and right wheel velocity vectors, respectively. The next step is to determine how to express \dot{p}_l and \dot{p}_r as functions of the generalized coordinates and generalized velocities. From the geometry of the robot it can be seen that

$$p_l = [x - \frac{L}{2} \sin \theta, y + \frac{L}{2} \cos \theta], \quad p_r = [x + \frac{L}{2} \sin \theta, y - \frac{L}{2} \cos \theta],$$

where p_l and p_r are the positions of the left and right wheels. By taking the derivative with respect to time the velocities are given by

$$\dot{p}_l = [\dot{x} - \dot{\theta} \frac{L}{2} \cos \theta, \dot{y} - \dot{\theta} \frac{L}{2} \sin \theta], \quad \dot{p}_r = [\dot{x} + \dot{\theta} \frac{L}{2} \cos \theta, \dot{y} + \dot{\theta} \frac{L}{2} \sin \theta].$$

It turns out that after some algebraic manipulation the no-slip kinematic constraints simply become:

$$\dot{p}_l \cdot e_{v,\perp} = \dot{p}_r \cdot e_{v,\perp} = \dot{x} \sin \theta - \dot{y} \cos \theta = 0,$$

which means having the no-slip kinematic constraint on both wheels is actually redundant! This also makes intuitive sense because the wheels are rigidly connected together, so if one wheel cannot move laterally then the other must not be able to. Additionally, it is noted that this nonholonomic constraint is identical to the one for the unicycle and so the kinematic model is also identical to (1.10). However the difference is that the *control inputs* can now be expressed in a more realistic form with respect to the actual geometry of the robot.

In particular, instead of the inputs being the forward speed v and body rotation rate ω as in (1.10), the inputs will be chosen to be the left and right wheel rotation rates, ω_l and ω_r . A relationship between these sets of inputs can be derived by exploiting the geometry of the robot and the no-slip wheel assumption. In particular, since the position $p = [x, y]$ can be written as $p = \frac{1}{2}(p_l + p_r)$ the

velocity vector $\dot{p} = \frac{1}{2}(\dot{p}_l + \dot{p}_r)$. From the no-slip wheel assumption the speed can be expressed as $v = e_v \cdot \dot{p}$, which can be simplified to

$$\begin{aligned} v &= e_v \cdot \dot{p}, \\ &= e_v \cdot \frac{1}{2}(\dot{p}_l + \dot{p}_r), \\ &= \frac{1}{2}(e_v \cdot \dot{p}_l + e_v \cdot \dot{p}_r), \\ &= \frac{1}{2}(v_l + v_r), \\ &= \frac{r}{2}(\omega_l + \omega_r), \end{aligned}$$

where r is the radius of the wheel and v_l and v_r are the speeds of the left and right wheels.

Additionally, the no-slip condition on each individual wheel can be expressed as $v_l = e_v \cdot \dot{p}_l$ and $v_r = e_v \cdot \dot{p}_r$ which can be expanded to

$$\begin{aligned} \dot{x} \cos \theta + \dot{y} \sin \theta - \dot{\theta} \frac{L}{2} &= v_l, \\ \dot{x} \cos \theta + \dot{y} \sin \theta + \dot{\theta} \frac{L}{2} &= v_r. \end{aligned}$$

Noting that $\dot{x} \cos \theta + \dot{y} \sin \theta = v$ these expressions can be written as $\frac{L}{2}\dot{\theta} = v_r - v$ and $\frac{L}{2}\dot{\theta} = v - v_l$. Finally, combining these expressions yields

$$\begin{aligned} L\dot{\theta} &= v_r - v_l, \\ &= r(\omega_r - \omega_l). \end{aligned}$$

In summary, a one-to-one mapping between the inputs is given by

$$v = \frac{r}{2}(\omega_l + \omega_r), \quad \omega = \frac{r}{L}(\omega_r - \omega_l).$$

Finally, the differential drive kinematic model is given by

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{r}{2} \cos \theta & \frac{r}{2} \cos \theta \\ \frac{r}{2} \sin \theta & \frac{r}{2} \sin \theta \\ \frac{r}{L} & -\frac{r}{L} \end{bmatrix} \begin{bmatrix} \omega_r \\ \omega_l \end{bmatrix}. \quad (1.11)$$

Overall, the complexity of this model over the unicycle model has not increased. However, by leveraging the geometry of the robot the inputs to this model may be more intuitive for motion planning and control tasks since the actual control mechanism is generally a motor attached to the wheel axles.

1.6 Dynamic Models

As was discussed in the introduction, mobile robot kinematic models are useful for describing fundamental physical behavior in a simple way, but they do not *completely* capture all real world influences on the robot's motion. The unicycle

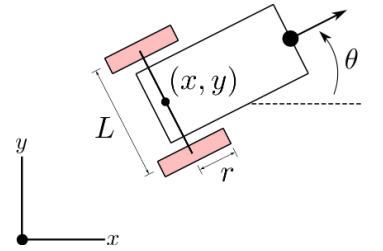


Figure 1.4: Generalized coordinates for a differential drive robot.

and differential drive models are examples of kinematic models that are approximations of the true system behavior. In particular they both make the no-slip wheel assumption, which directly lead to the kinematic constraints. Additionally, the choice of the inputs for the kinematic models ignores other important dynamics of the robot. In the unicycle model it is assumed the velocity is the input, but in practice directly commanding a desired velocity is not always straightforward since the amount of force required to change velocities varies with the mass of the robot ($F = ma$). In the differential drive model the inputs are the rotational rates of the wheels, but again in practice the amount of torque output required by the motor to change the rotation rate can vary depending on the robot's mass as well as other motor dynamics.

One common extension to kinematic models to incorporate *dynamics* is to simply add integrators to replace the input variables. The most common example of this is to replace a velocity input v with an acceleration input a and add the integrator $\dot{v} = a$. The force that generates the acceleration can then be considered as the input by using the dynamics equation $\dot{v} = \frac{1}{m}F$ where m is the mass of the robot. Similarly, a rotation rate input ω could be replaced by a rotational acceleration input. For example, the unicycle model (1.10) could be extended with integrator states to be

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{v} \\ \dot{\theta} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} v \cos \theta \\ v \sin \theta \\ a \\ \omega \\ \alpha \end{bmatrix}. \quad (1.12)$$

where a is linear acceleration in the forward direction and α is the angular acceleration (which of course could also be written with forces and torques as inputs).

In summary, factors to consider when deciding whether a certain kinematic model is sufficient, or if additional kinematics/dynamics are needed, include the robot's configuration/geometry and the task at hand (e.g. planning, control, etc.).