Modelling of Ground Vehicles

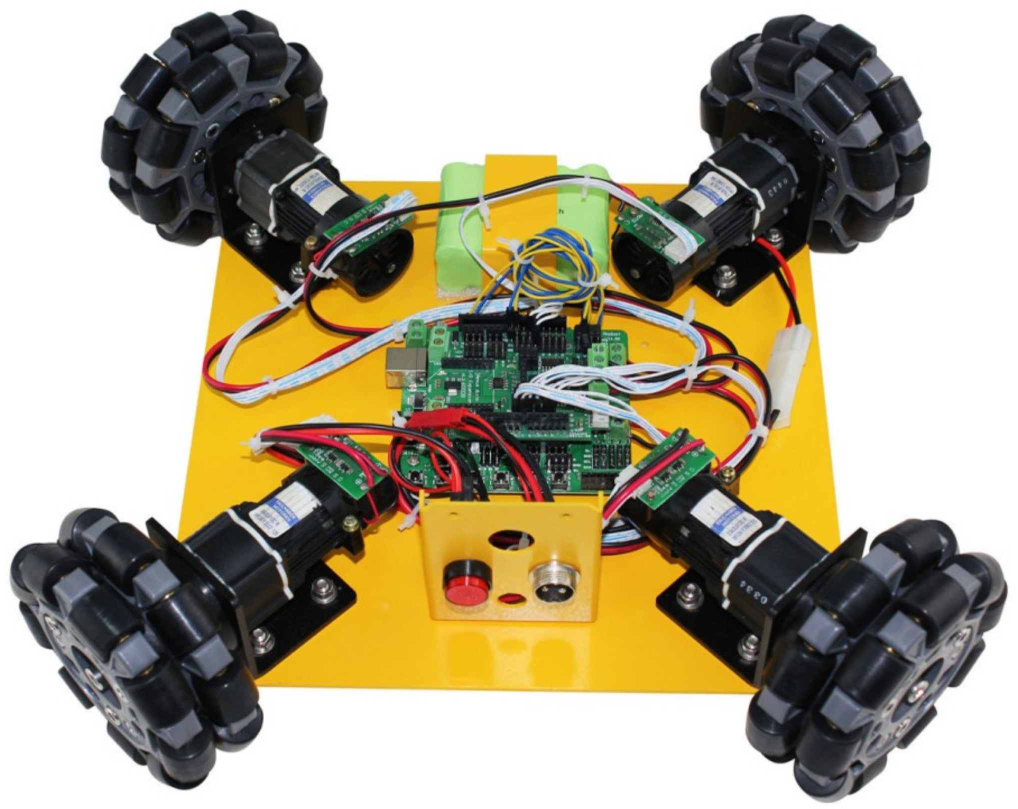
TSFS12: Autonomous Vehicles –planning, control, and learning systems

Lecture 3: Jan Åslund <jan.aslund@liu.se>

LINKÖPINGS UNIVERSITET

Modelling of Ground Vehicles

Today I will consider different types of ground vehicles



One important question: In which directions can the different vehicles move?

Mathematical Modelling

Mathematical Modelling of Systems

In the basic automatic control course linear state-space models in the form

$$\begin{cases} \dot{\mathbf{q}}(t) = A\mathbf{q}(t) + B\mathbf{u}(t) \\ \mathbf{y}(t) = C\mathbf{q}(t) + D\mathbf{u}(t) \end{cases}$$

were considered. The state-vector is denoted $\mathbf{q}$, $\mathbf{u}$ is the control vector, and $\mathbf{y}$ is the output vector.

In this course, we will consider more general non-linear models

$$\begin{cases} \dot{\mathbf{q}}(t) = \mathbf{f}(\mathbf{q}(t), \mathbf{u}(t)) \\ \mathbf{y}(t) = \mathbf{h}(\mathbf{q}(t), \mathbf{u}(t)) \end{cases}$$

where $\mathbf{f}$ and $\mathbf{h}$ are non-linear functions.

Non-linear state space models

I will derive models in the form $\dot{\mathbf{q}} = \mathbf{f}(\mathbf{q}, \mathbf{u})$ for different kinds ground vehicles, where $\mathbf{q}$ is the state vector and $\mathbf{u}$ is the input vector.

First, I will describe how a system of the type $\dot{\mathbf{q}} = \mathbf{f}(\mathbf{q}, \mathbf{u})$ can be interpreted geometrically, and I will use the model

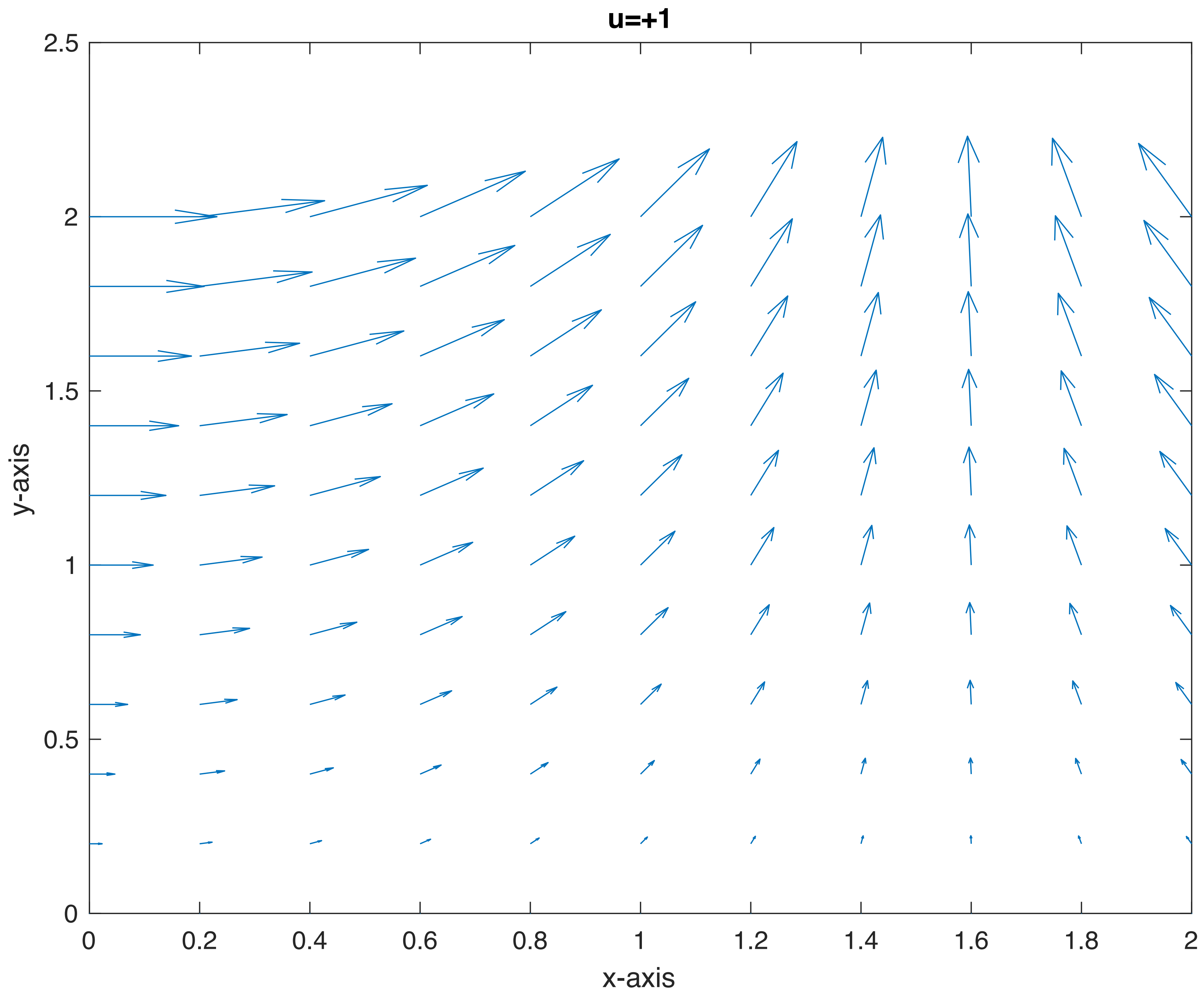
$$\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \begin{pmatrix} y \cos x \\ uy \sin x \end{pmatrix}$$

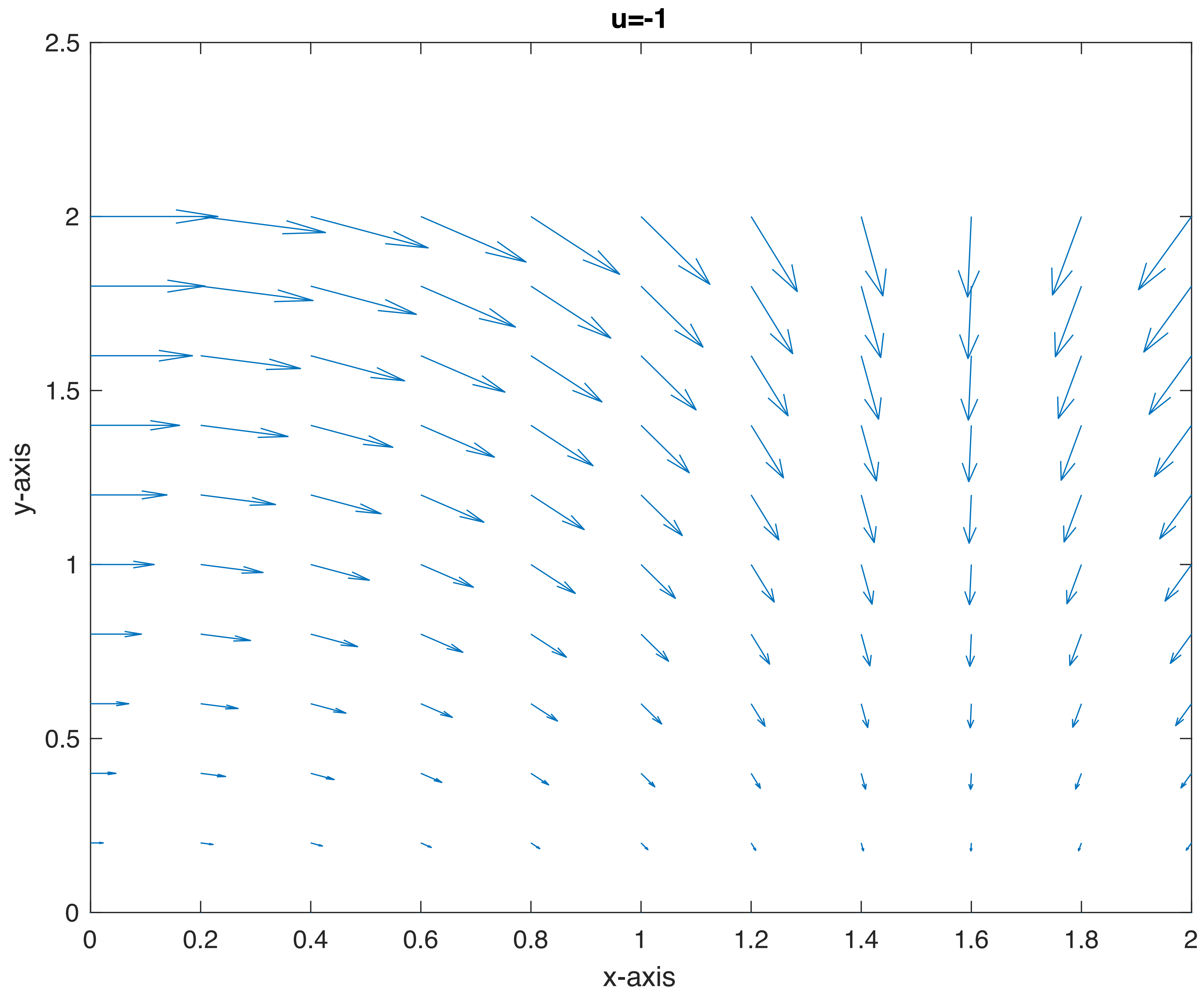
for illustration.

The state vector is $\mathbf{q} = \begin{pmatrix} x \\ y \end{pmatrix}$ and $\mathbf{f}(\mathbf{q}, \mathbf{u}) = \begin{pmatrix} y \cos x \\ uy \sin x \end{pmatrix}$ in this example

For every point (x, y) and control signal u, the right-hand side is a vector

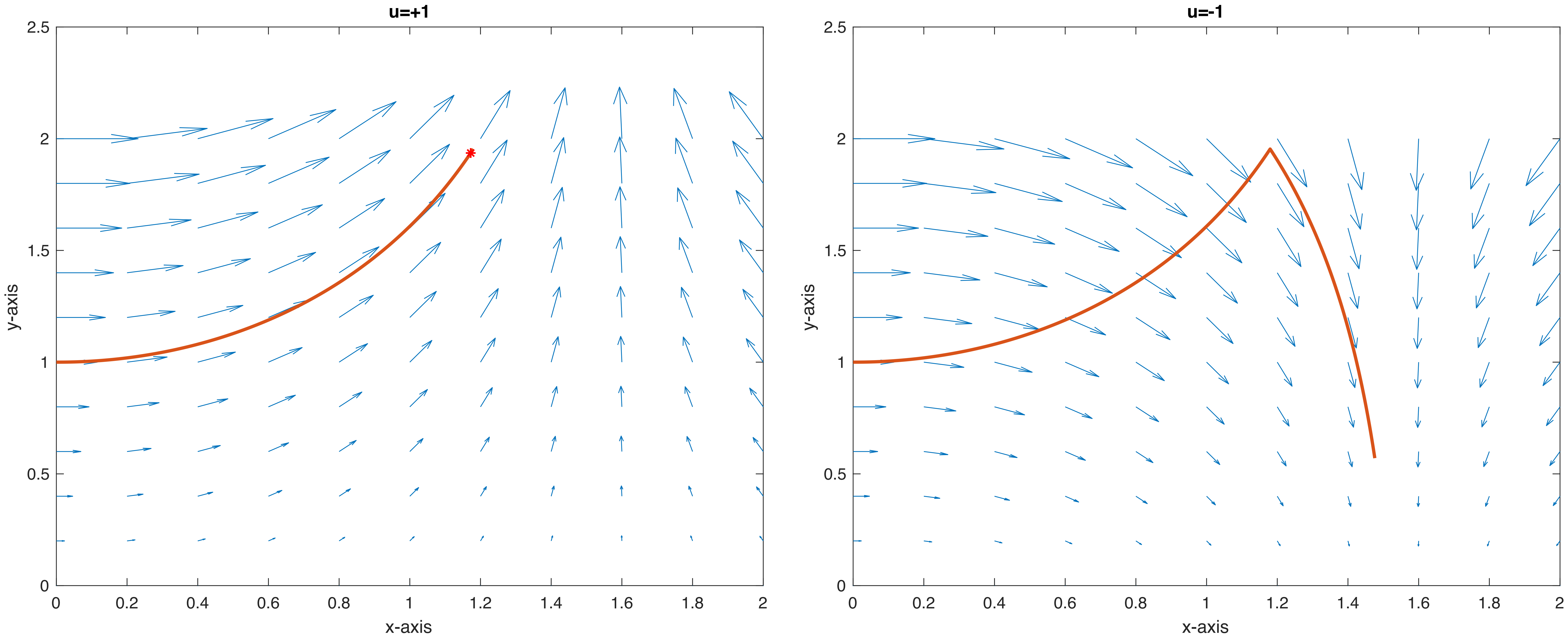
$$\mathbf{f}(\mathbf{q}, \mathbf{u}) = \begin{pmatrix} y\cos x \\ uy\sin x \end{pmatrix}$$





The vector fields for the two values $u = \pm 1$

The left-hand side of $\dot{\mathbf{q}} = f(\mathbf{q}, \mathbf{u})$, can be interpreted as the velocity vector of a particle and the solution is the trajectory of the particle moving in the vector field.



A particle moving first in the vector field corresponding to $u = +1$ and then the control signal switches to $u = -1$

Double integrators

Another type models often considered in control theory are double integrators

$$\begin{cases} \ddot{\mathbf{q}}(t) = \mathbf{f}(\mathbf{q}(t), \mathbf{u}(t)) \\ \mathbf{y}(t) = \mathbf{h}(\mathbf{q}(t), \mathbf{u}(t)) \end{cases}$$

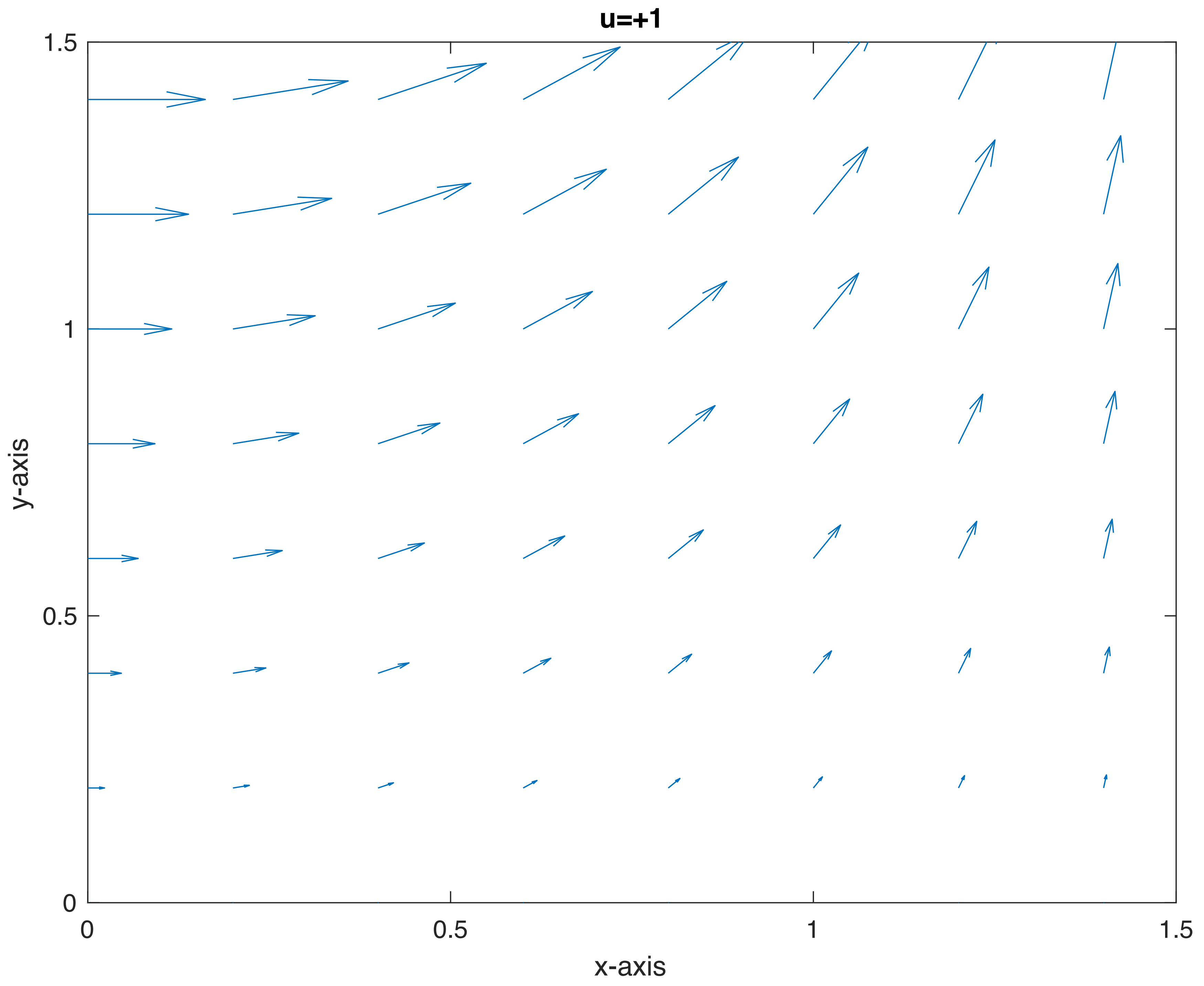
A typical application is dynamic systems where the left-hand side $\ddot{\mathbf{q}}$ is the acceleration and the right-hand side $\mathbf{f}(\mathbf{q}, \mathbf{u})$ is representing the force acting on the system.

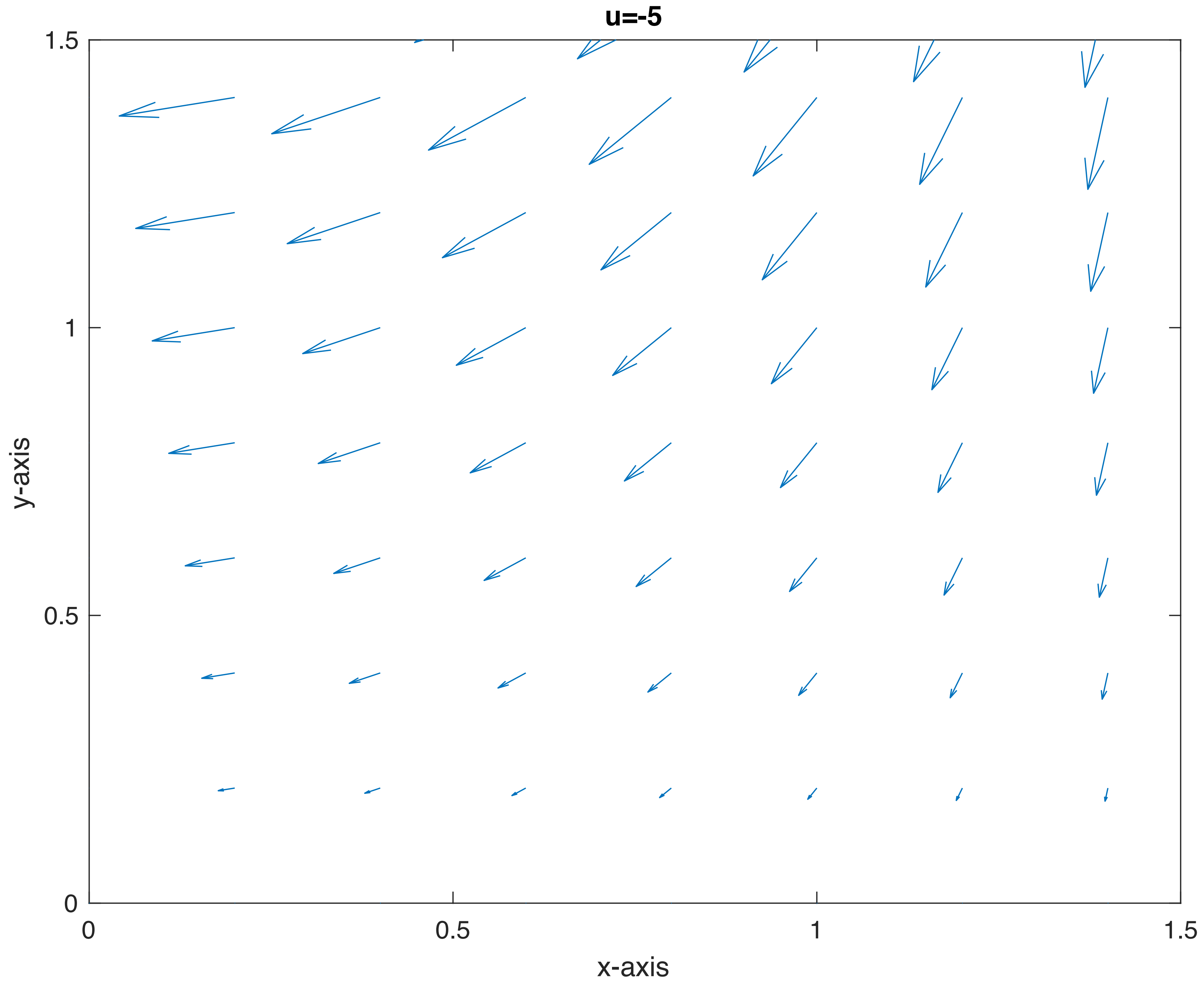
Note that the double integrator can be written in state-space form with states $\mathbf{x}_1 = \mathbf{q}$ and $\mathbf{x}_2 = \dot{\mathbf{q}}$:

$$\begin{cases} \dot{\mathbf{x}}_1 = \mathbf{x}_{\mathbf{2}} \\ \dot{\mathbf{x}}_2 = \mathbf{f}(\mathbf{x}_1, \mathbf{u}) \\ \mathbf{y} = \mathbf{h}(\mathbf{x}_{\mathbf{1}}, \mathbf{u}) \end{cases}$$

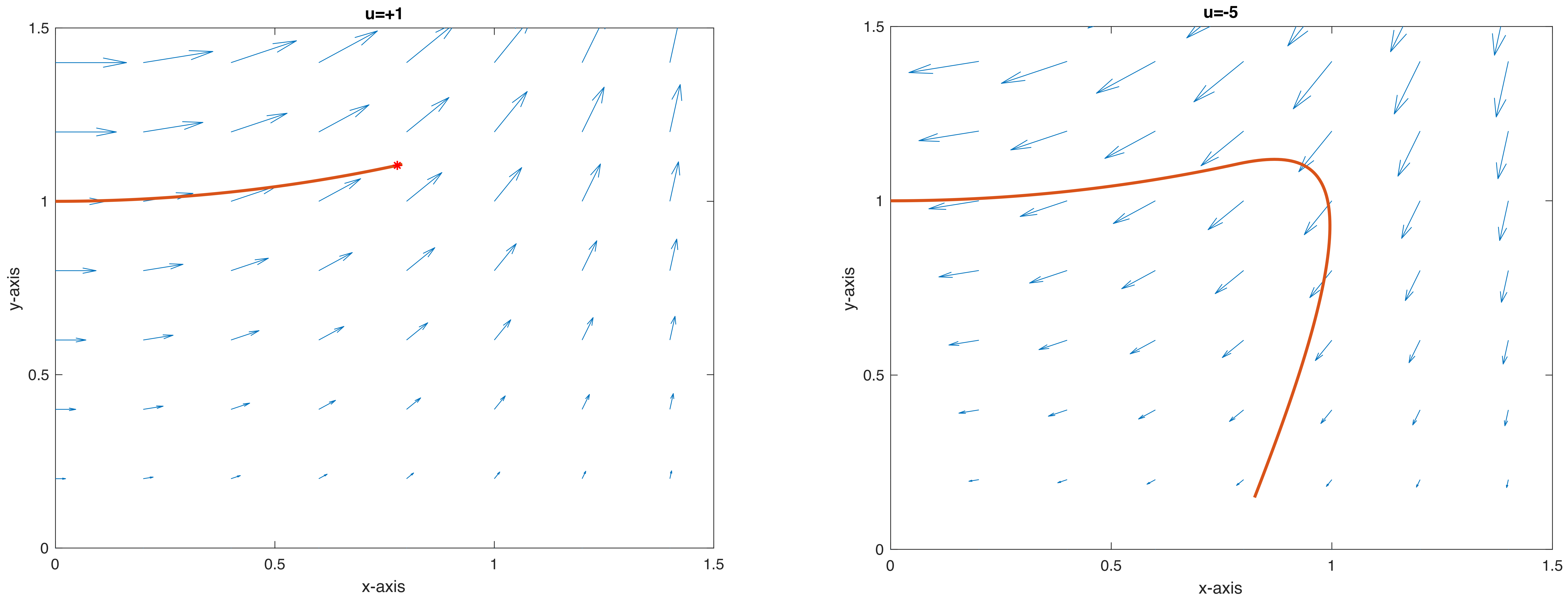
The same left-hand side as before is considered.

In this case $u = +1$ and $u = -5$ respectively:



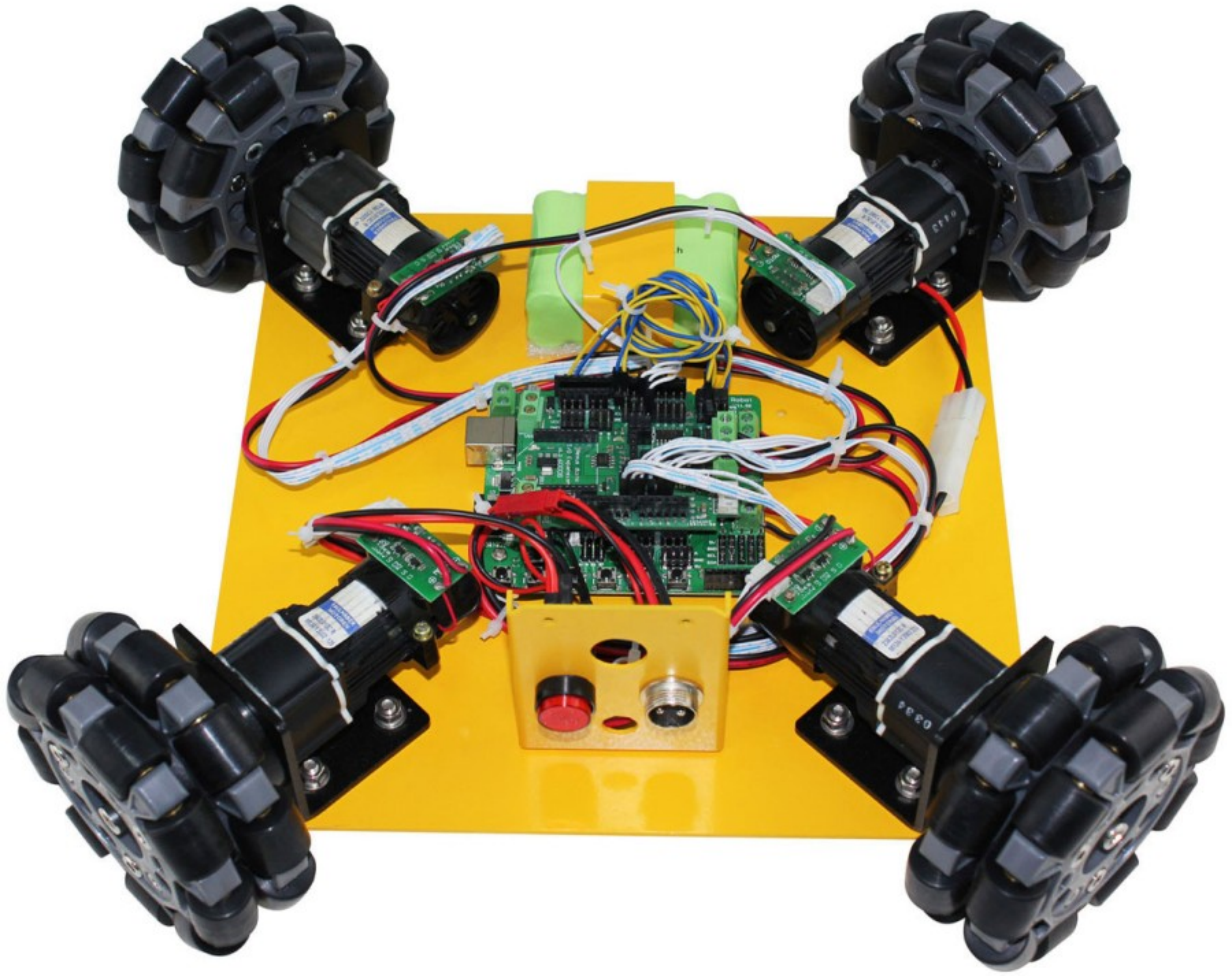


In this case the left-hand side is equal to the acceleration:



The Omnidirectional robot

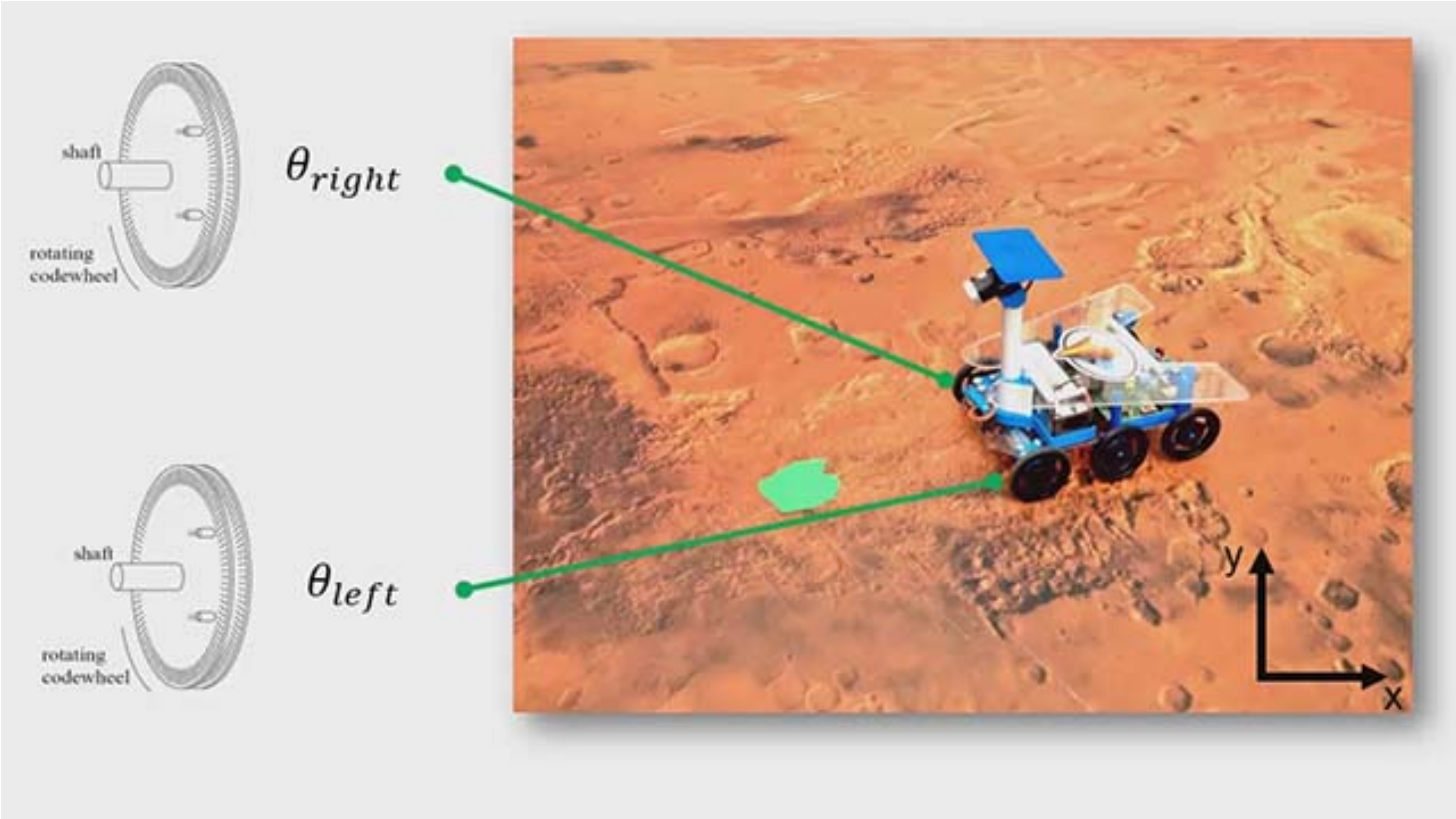
The Omnidirectional Robot



The Differentially-Driven Mobile Robot

The Differentially-Driven Mobile Robot

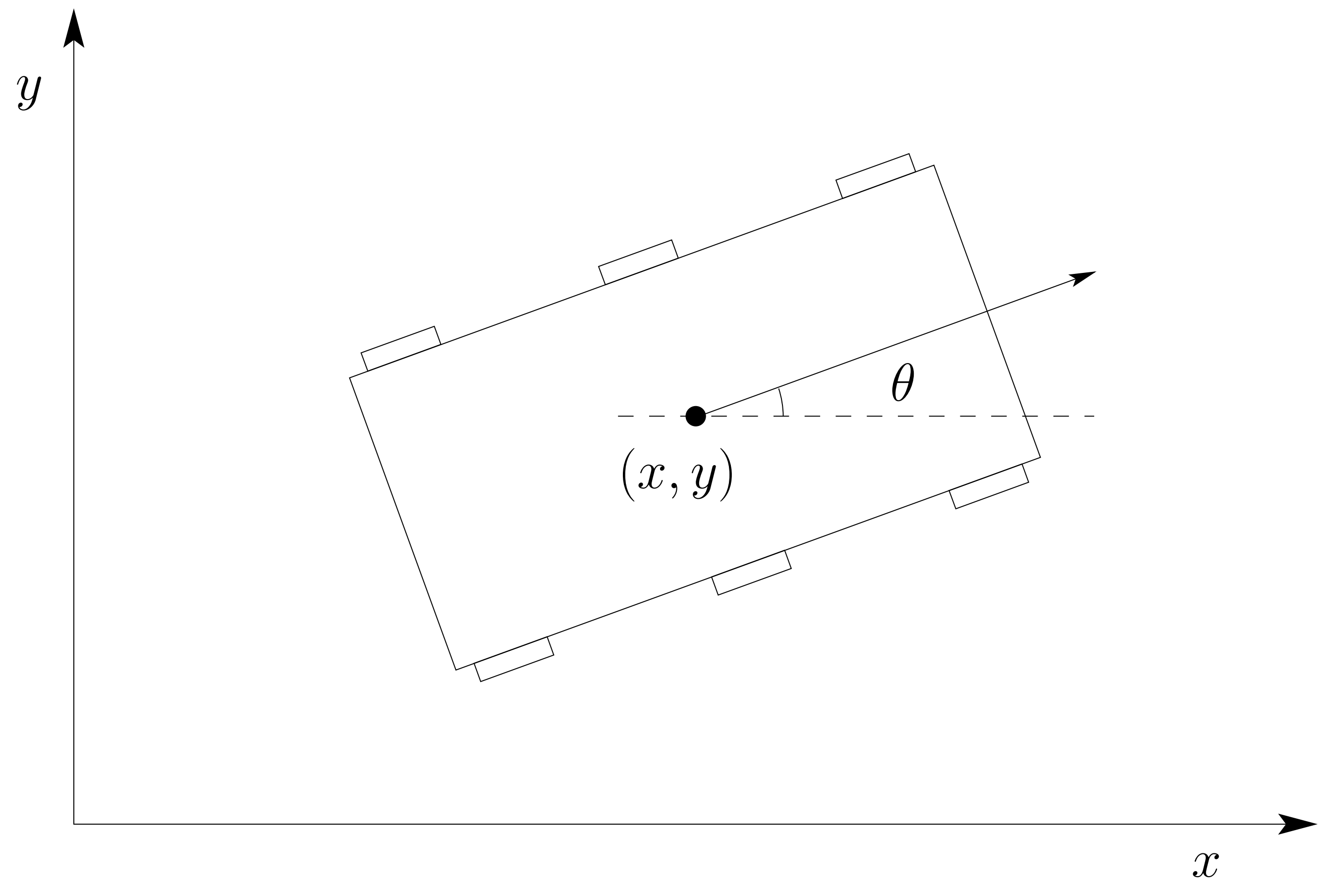
Has two separately driven wheels placed on either side of the body of the robot.



If the wheels rotate in the same direction, then the robot moves forward, and if they rotate in opposite direction, then the robot rotates.

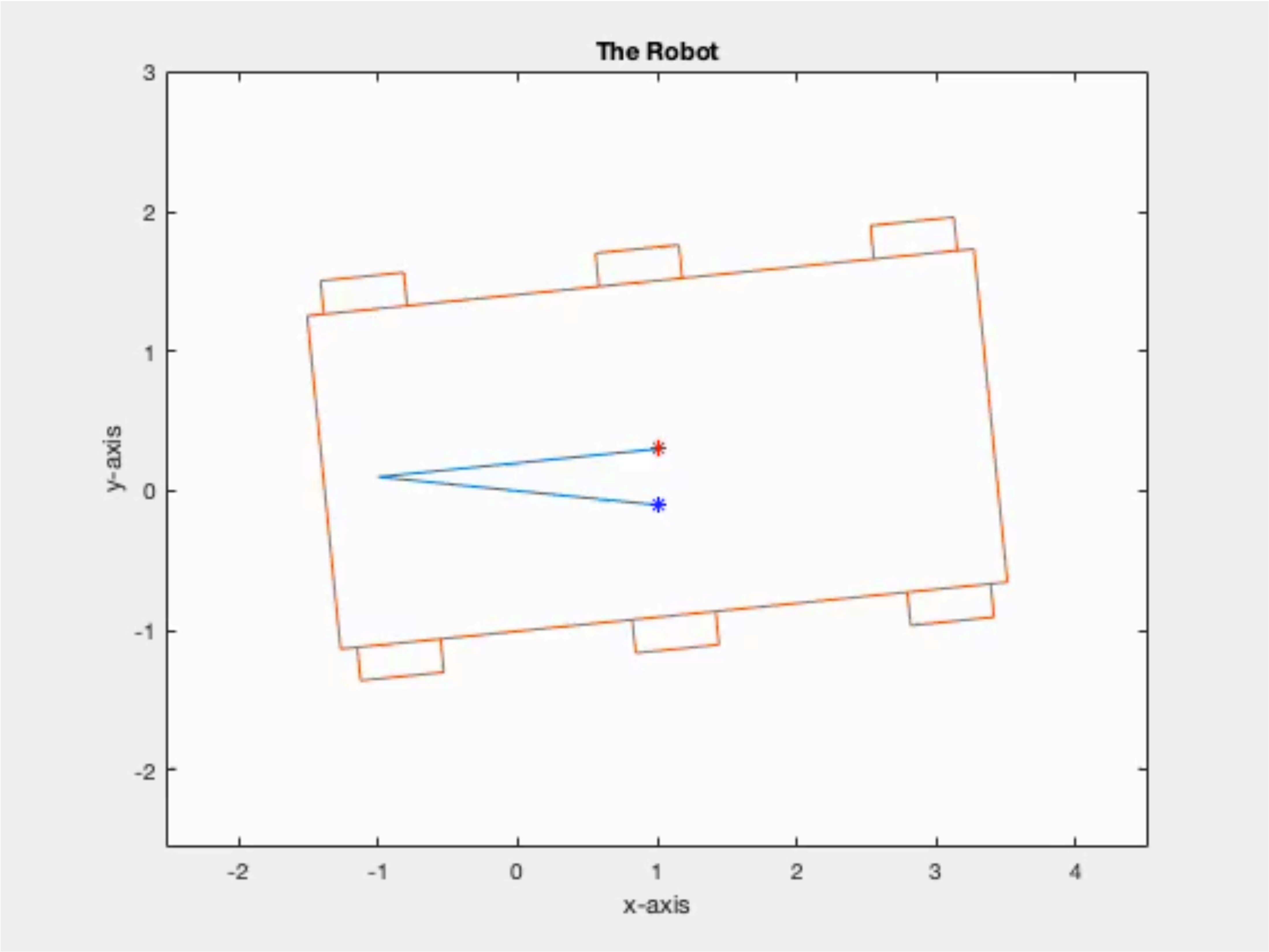
Underactuated Systems

For the differentially driven mobile robot, we shall use kinematic model with three states (position (x, y) and orientation θ with respect to x-axis), and linear combinations of the two wheel speeds as control signals. The system is called **underactuated** since the number of actuators is less than the dimension of the state space.



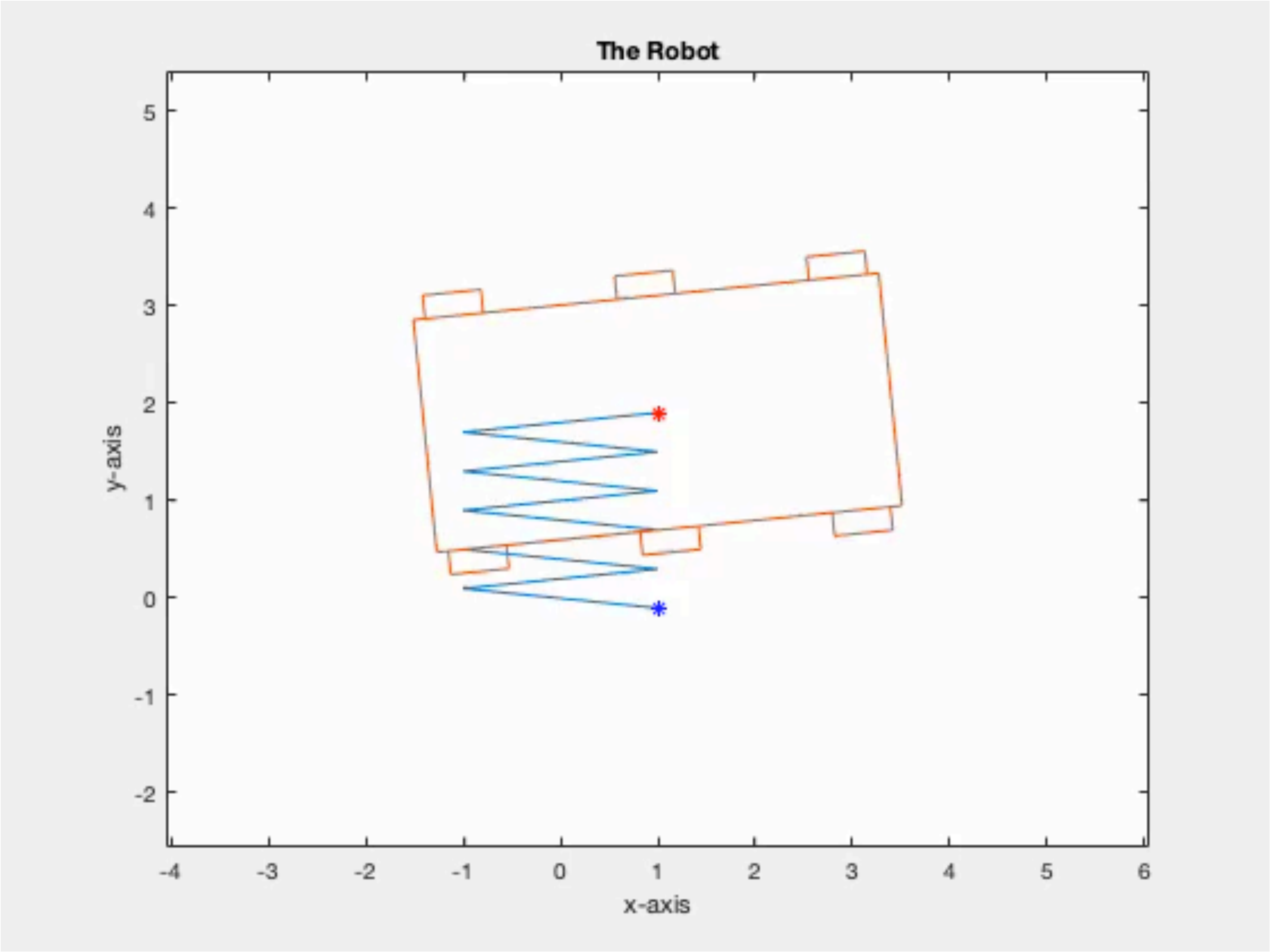
How can we accomplish a movement in the direction perpendicular the orientation of the vehicle?

Solution inspired by parallel parking

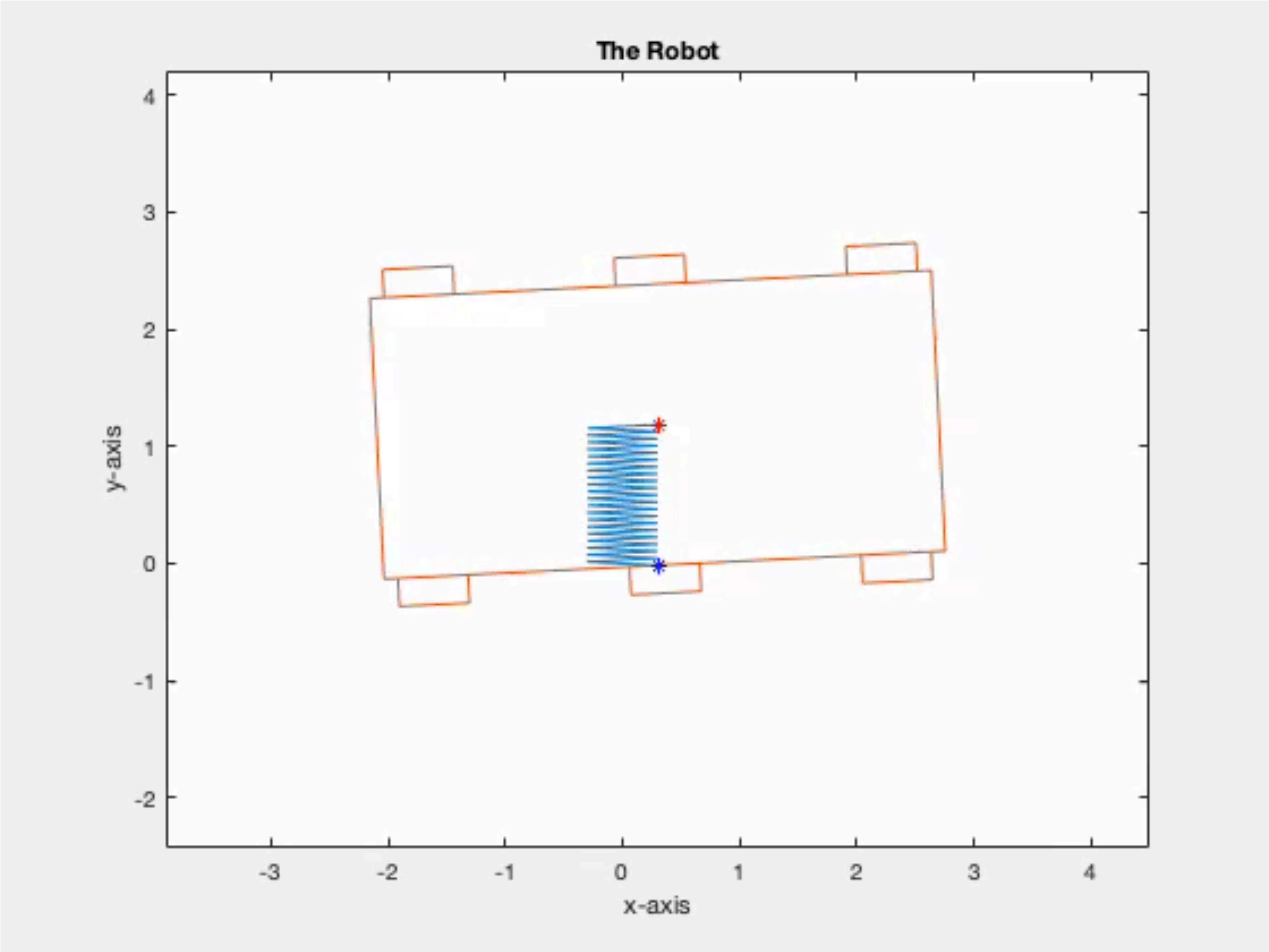


Conclusion: The robot moves one step upwards

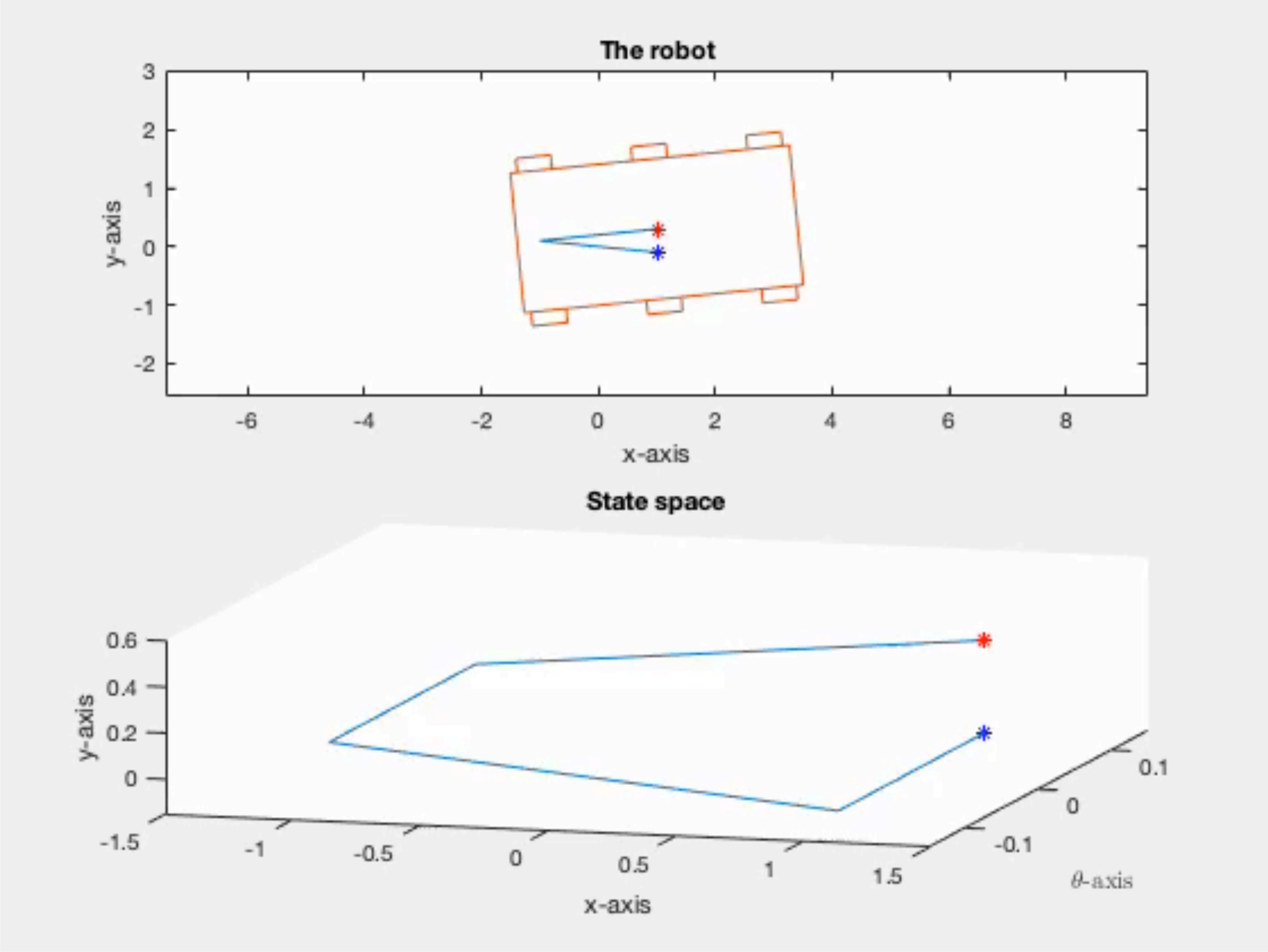
Repeating the procedure:



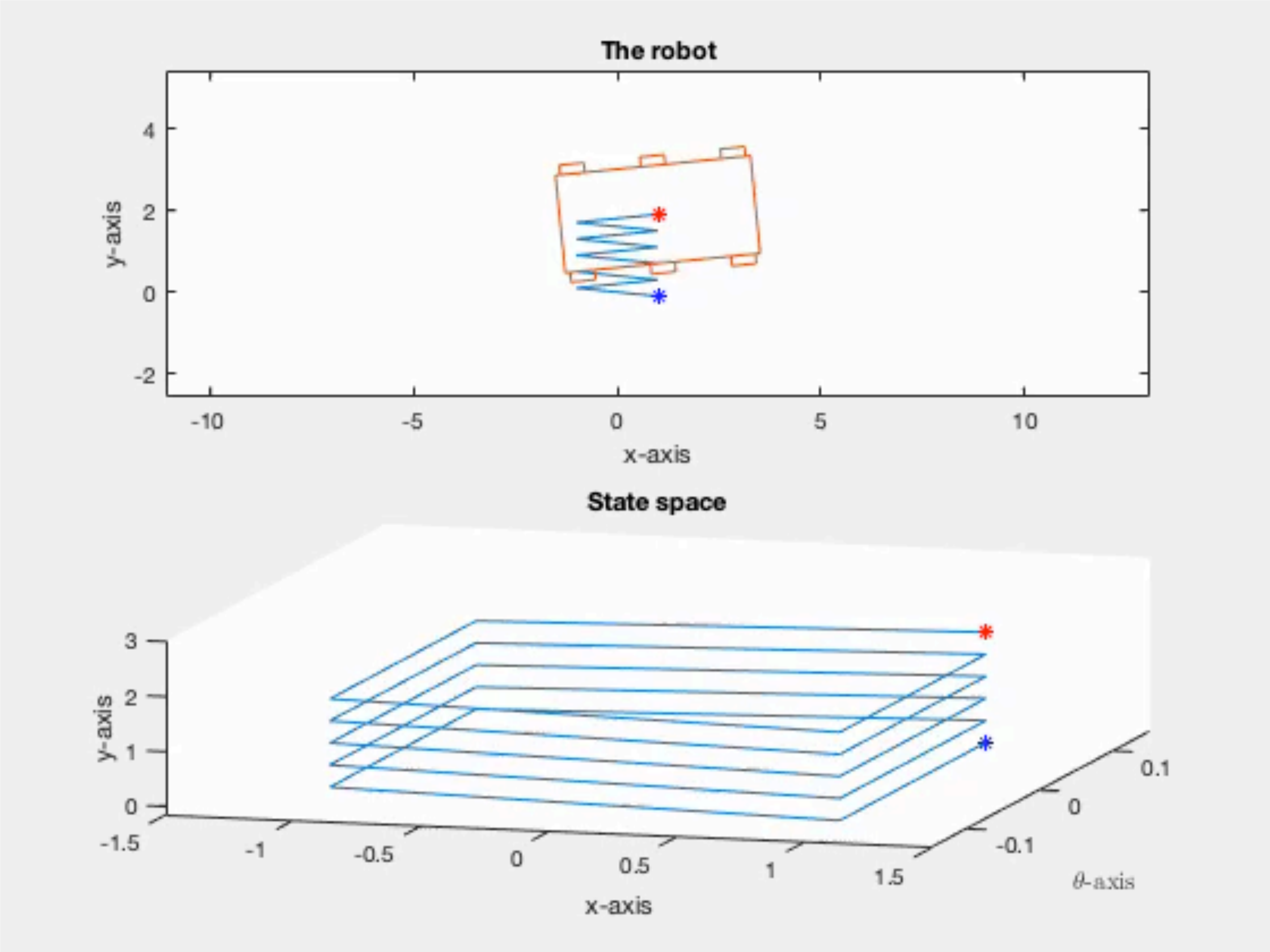
By taking smaller and smaller steps, the procedure can approximate a motion straight in the *y*-direction:



Add the trajectory in the state space:



Repeat the procedure:



Model of the robot

With the state vector $\mathbf{q} = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix}$ and control vector $\mathbf{u} = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$, a model of the differential driven mobile robot can be written in the form

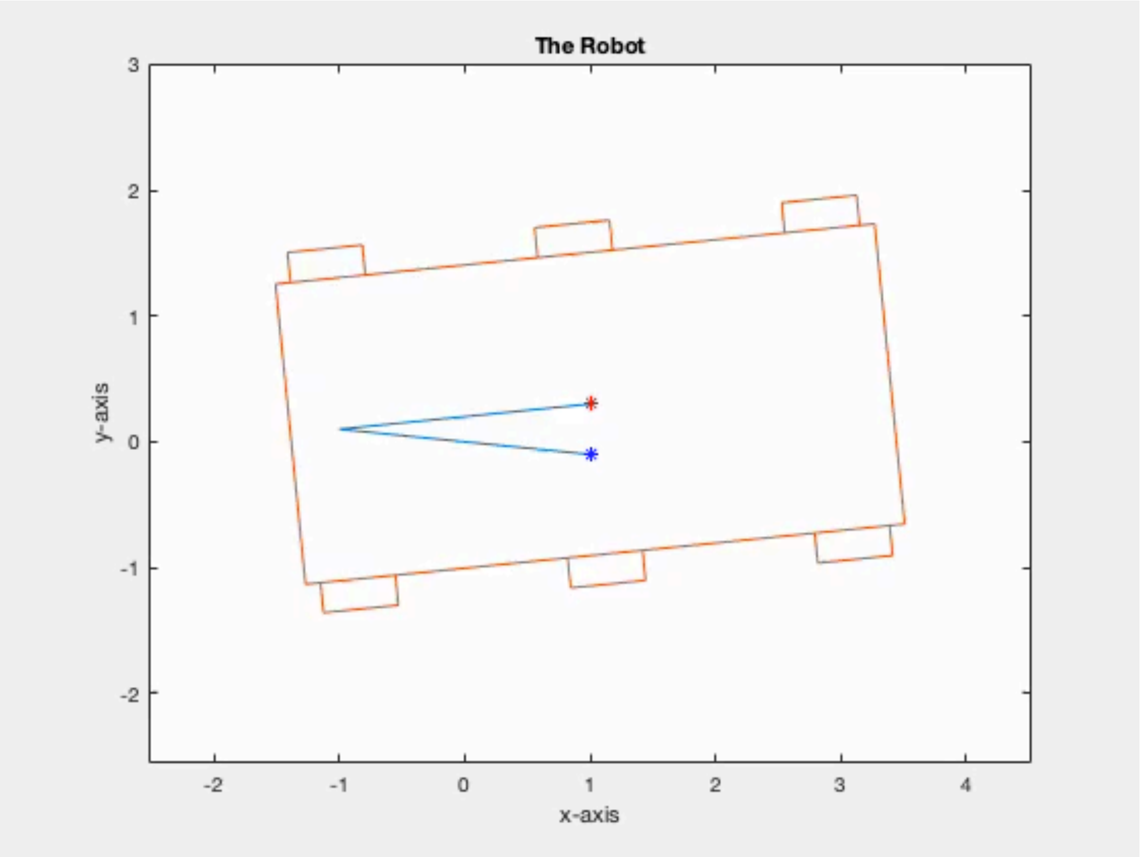
$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \dot{\mathbf{q}} = \mathbf{f}(\mathbf{q}, \mathbf{u}) = u_1\mathbf{f}_1(\mathbf{q}) + u_2\mathbf{f}_2(\mathbf{q})$$

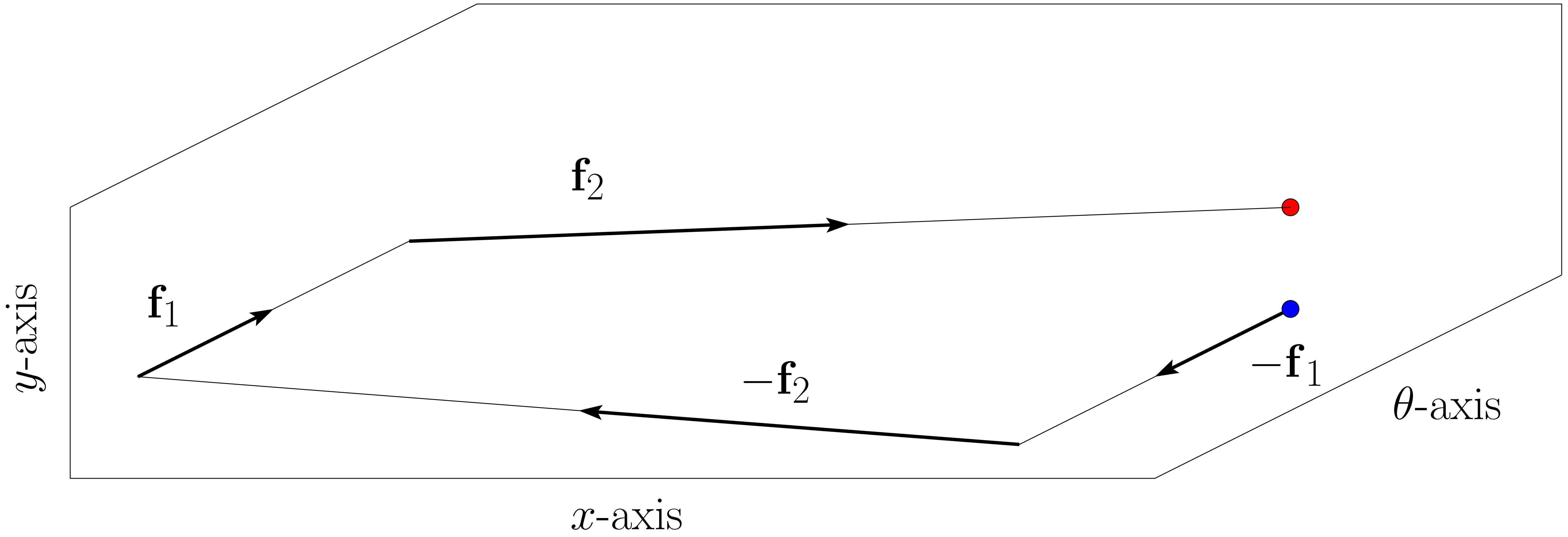
where $\mathbf{f}_1(\mathbf{q}) = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$ and $\mathbf{f}_2(\mathbf{q}) = \begin{pmatrix} \cos\theta \\ sin\theta \\ 0 \end{pmatrix}$

The speed of the driven wheels are $v_r = u_2 + bu_1$ and $v_l = u_2 - bu_1$ where b is half of the distance between the right and left wheels.

In the maneuvers described above, the robot moved in the directions

$$\mathbf{f}_1(\mathbf{q}) = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \text{ and } \mathbf{f}_2(\mathbf{q}) = \begin{pmatrix} \cos\theta \\ sin\theta \\ 0 \end{pmatrix}$$



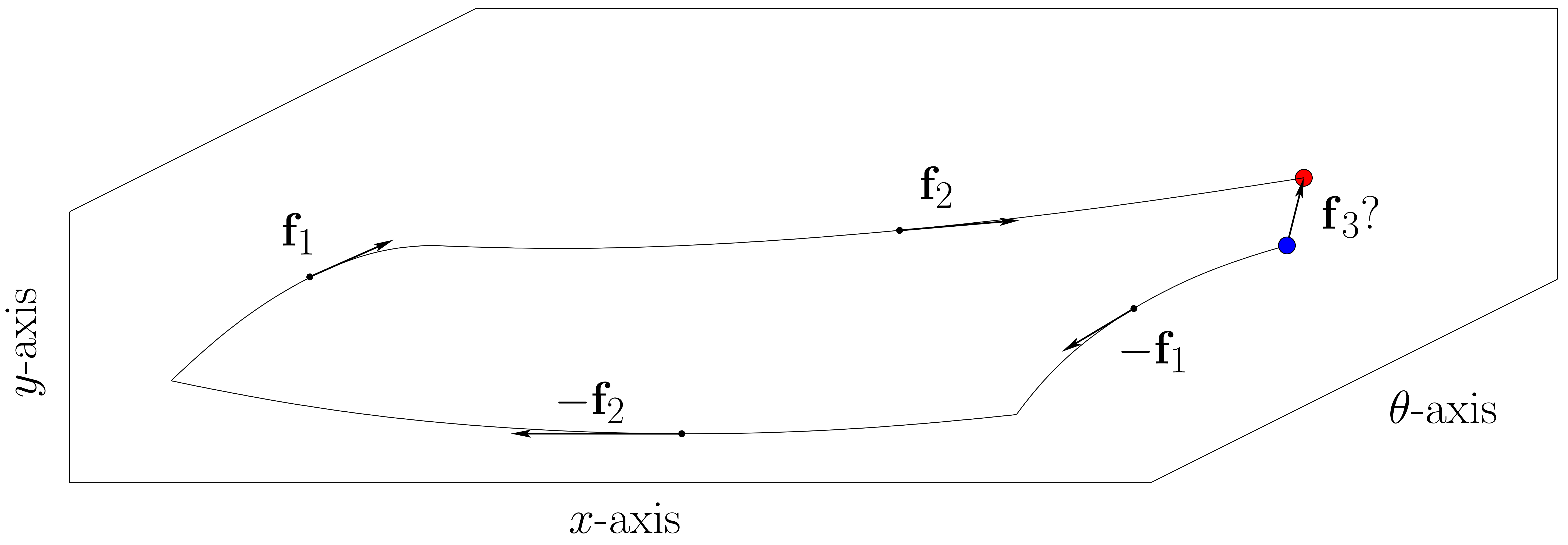


$$\mathbf{q} = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix}$$

This is the trajectory of $\dot{\mathbf{q}} = u_1\mathbf{f}_1(\mathbf{q}) + u_2\mathbf{f}_2(\mathbf{q})$ with the control sequence:

$(u_1, u_2) = (-1,0), (0, -1), (1,0), (0,1)$

Now we shall investigate what happens if we repeat the procedure for a system
$\dot{\mathbf{q}} = u_1 f_1(\mathbf{q}) + u_2 f_2(\mathbf{q})$
with more general vector fields $f_1(\mathbf{q})$ and $f_2(\mathbf{q})$ and arbitrary small time steps $\Delta t = \varepsilon$.



Question: How can we compute an approximation of $\mathbf{f}_3$?

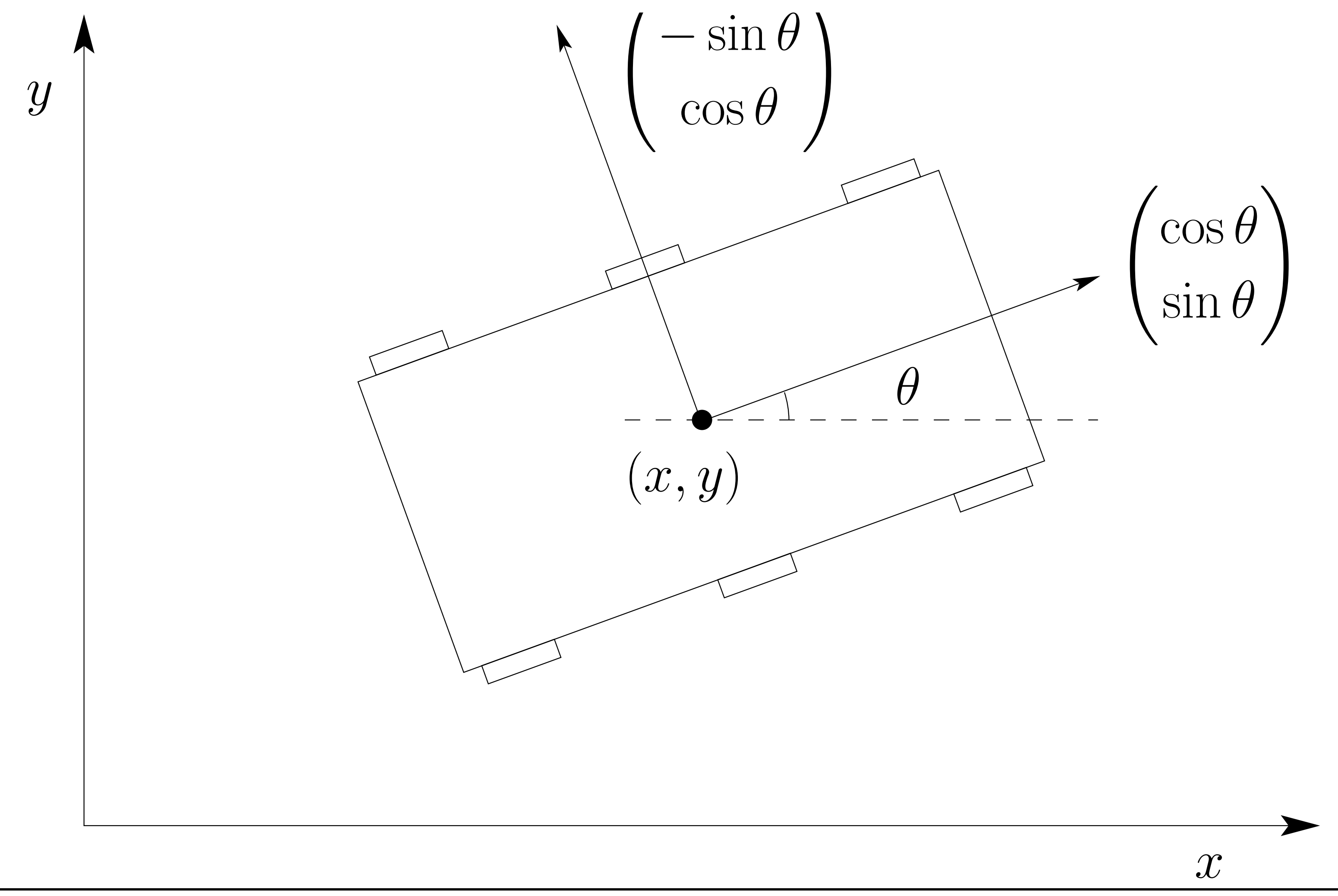
Answer: $\mathbf{f}_3 = \varepsilon^2((\mathbf{f}_2)_\mathbf{q}\mathbf{f}_1 - (\mathbf{f}_1)_\mathbf{q}\mathbf{f}_2) + \mathcal{O}(\varepsilon^3)$ where $(\mathbf{f}_i)_\mathbf{q} = \left(\frac{\partial \mathbf{f}_i}{\partial x} \frac{\partial \mathbf{f}_i}{\partial y} \frac{\partial \mathbf{f}_i}{\partial \theta}\right)$ is the Jacobian matrix.

The expression $(\mathbf{f}_2)_{\mathbf{q}}\mathbf{f}_1 - (\mathbf{f}_1)_{\mathbf{q}}\mathbf{f}_2$ is called the Lie bracket and is denoted $[\mathbf{f}_1, \mathbf{f}_2]$

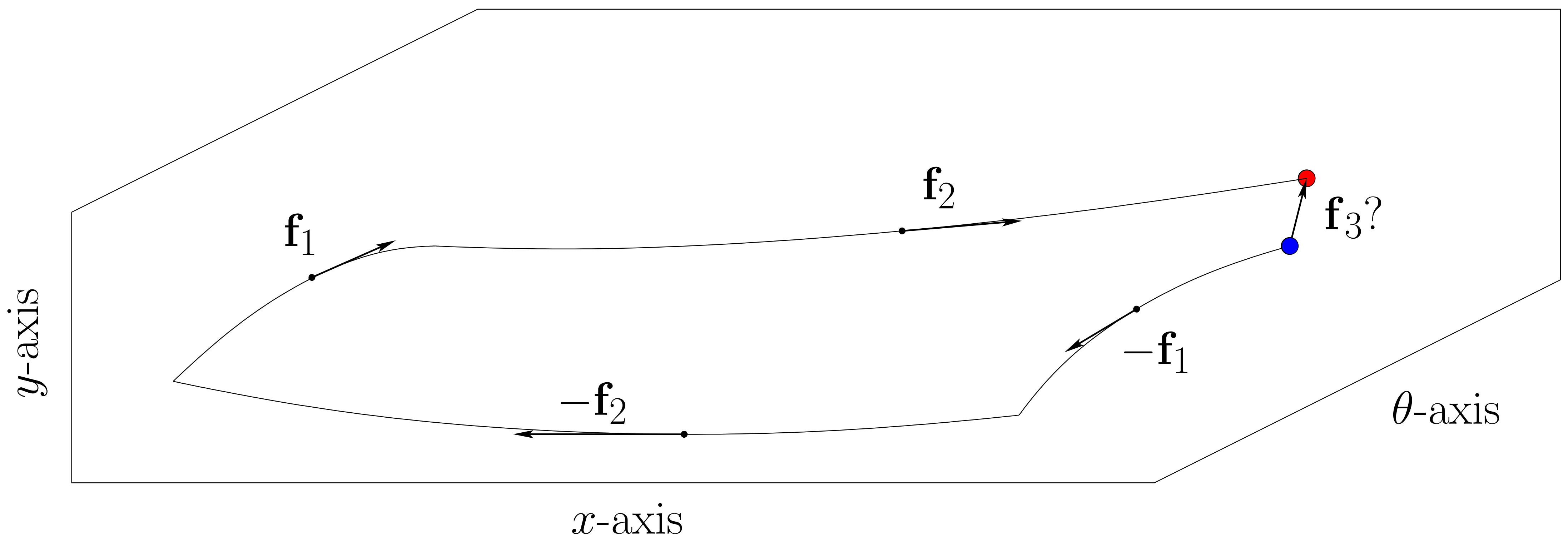
For example, with $\mathbf{f}_1(\mathbf{q}) = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$ and $\mathbf{f}_2(\mathbf{q}) = \begin{pmatrix} \cos\theta \\ sin\theta \\ 0 \end{pmatrix}$ the result is

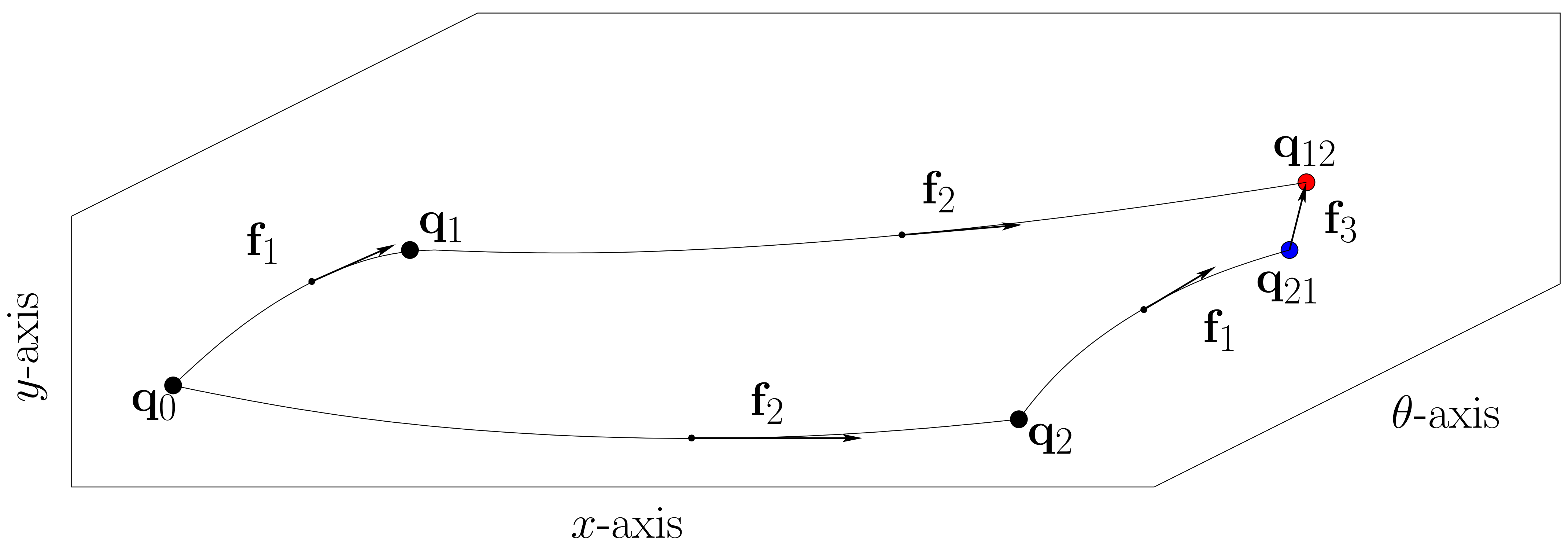
$$[\mathbf{f}_1, \mathbf{f}_2] = (\mathbf{f}_2)_{\mathbf{q}}\mathbf{f}_1 - (\mathbf{f}_1)_{\mathbf{q}}\mathbf{f}_2 = \left(\frac{\partial \mathbf{f}_2}{\partial x}\frac{\partial \mathbf{f}_2}{\partial y}\frac{\partial \mathbf{f}_2}{\partial \theta}\right)\mathbf{f}_1 - \left(\frac{\partial \mathbf{f}_1}{\partial x}\frac{\partial \mathbf{f}_1}{\partial y}\frac{\partial \mathbf{f}_1}{\partial \theta}\right)\mathbf{f}_2$$

$$= \begin{pmatrix} 0 & 0 & -\sin\theta \\ 0 & 0 & \cos\theta \\ 0 & 0 & 0 \end{pmatrix}\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} - \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}\begin{pmatrix} \cos\theta \\ sin\theta \\ 0 \end{pmatrix} = \begin{pmatrix} -\sin\theta \\ cos\theta \\ 0 \end{pmatrix}$$

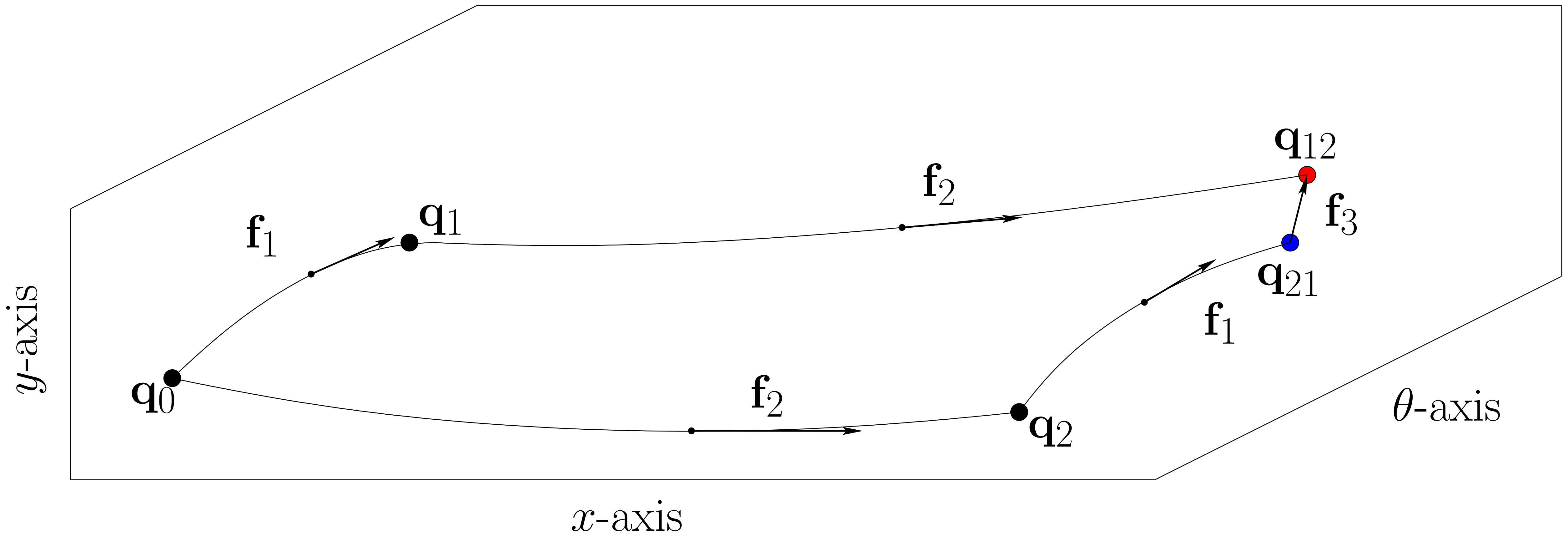


Sketch of proof





Instead of moving as in the figure to the left, we start at position $\mathbf{q}_0$ and first move as in the figure to the right to the two positions $\mathbf{q}_{12}$ and $\mathbf{q}_{21}$ and then compute $\mathbf{f}_3 = \mathbf{q}_{12} - \mathbf{q}_{21}$



$$\mathbf{q} = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix}$$

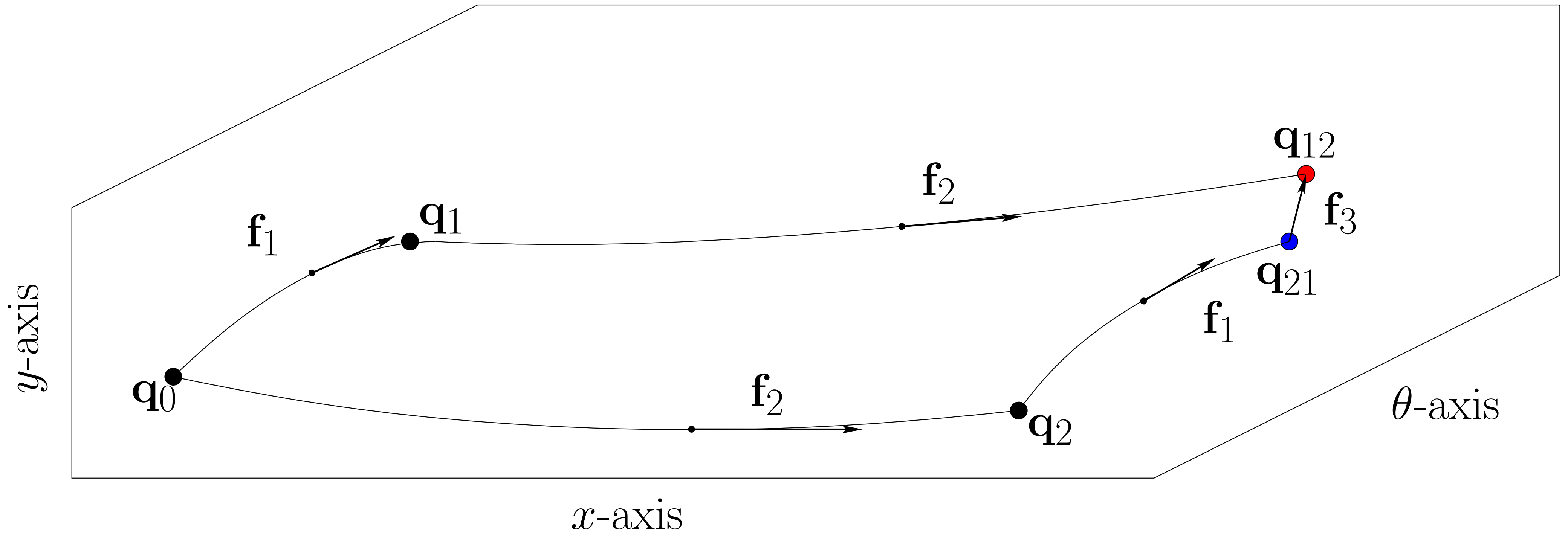
First move from $\mathbf{q}_0$ to $\mathbf{q_1}$ along $\mathbf{f}_1$ ($\Delta t = \varepsilon$):

$$\mathbf{q}_1 = \mathbf{q}_0 + \varepsilon \dot{\mathbf{q}}(0) + \frac{\varepsilon^2}{2} \ddot{\mathbf{q}}(0) + \mathcal{O}(\varepsilon^3)$$

$$= \mathbf{q}_0 + \varepsilon \mathbf{f}_1(\mathbf{q}_0) + \frac{\varepsilon^2}{2} (\mathbf{f}_1)_q(\mathbf{q}_0)\mathbf{f}_1(\mathbf{q}_0) + \mathcal{O}(\varepsilon^3)$$

where $(\mathbf{f}_1)_q = \left[\frac{\partial \mathbf{f}_1}{\partial x} \frac{\partial \mathbf{f}_1}{\partial y} \frac{\partial \mathbf{f}_1}{\partial \theta} \right]$ is the Jacobian matrix. We used that $\dot{\mathbf{q}} = \mathbf{f}_1(\mathbf{q})$ and

$$\ddot{\mathbf{q}} = \frac{d}{dt}\dot{\mathbf{q}} = \frac{d}{dt}\mathbf{f}_1(\mathbf{q}) = \frac{\partial \mathbf{f}_1}{\partial x}(\mathbf{q})\dot{x} + \frac{\partial \mathbf{f}_1}{\partial y}(\mathbf{q})\dot{y} + \frac{\partial \mathbf{f}_1}{\partial \theta}(\mathbf{q})\dot{\theta} = (\mathbf{f}_1)_q(\mathbf{q})\mathbf{f}_1(\mathbf{q})$$



By repeating the process we get

$$\mathbf{q}_1 = \mathbf{q}_0 + \varepsilon \mathbf{f}_1(\mathbf{q}_0) + \frac{\varepsilon^2}{2}(\mathbf{f}_1)_q(\mathbf{q}_0)\mathbf{f}_1(\mathbf{q}_0) + \mathcal{O}(\varepsilon^3)$$

$$\mathbf{q}_{12} = \mathbf{q}_1 + \varepsilon \mathbf{f}_2(\mathbf{q}_1) + \frac{\varepsilon^2}{2}(\mathbf{f}_2)_q(\mathbf{q}_1)\mathbf{f}_2(\mathbf{q}_1) + \mathcal{O}(\varepsilon^3)$$

Substitute $\mathbf{q}_1$ into $\mathbf{q}_{12}$:

$$\begin{aligned}\mathbf{q}_{12} = \mathbf{q}_0 + \varepsilon \mathbf{f}_1(\mathbf{q}_0) + \frac{\varepsilon^2}{2}(\mathbf{f}_1)_q(\mathbf{q}_0)\mathbf{f}_1(\mathbf{q}_0) \\ + \varepsilon \mathbf{f}_2(\mathbf{q}_0 + \varepsilon \mathbf{f}_1(\mathbf{q}_0)) \\ + \frac{\varepsilon^2}{2}(\mathbf{f}_2)_q(\mathbf{q}_0)\mathbf{f}_2(\mathbf{q}_0) + \mathcal{O}(\varepsilon^3)\end{aligned}$$

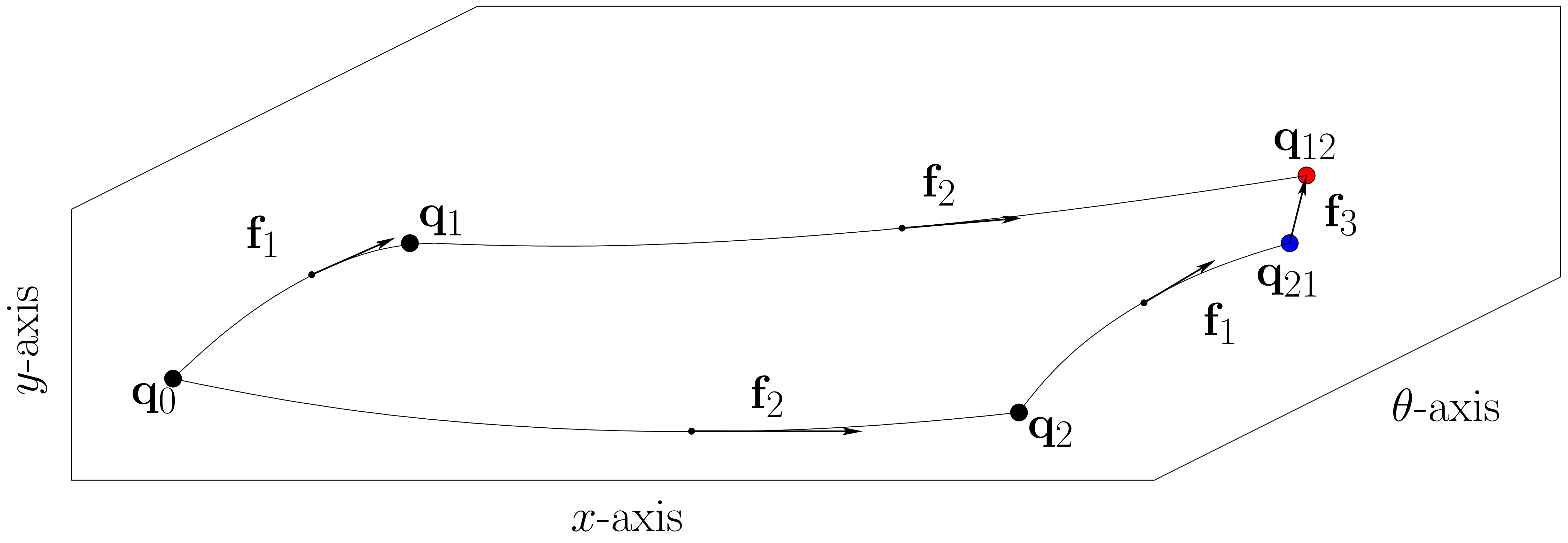
The Taylor expansion $\mathbf{f}_2(\mathbf{q}_0 + \varepsilon \mathbf{f}_1(\mathbf{q}_0)) = \mathbf{f}_2(\mathbf{q}_0) + \varepsilon(\mathbf{f}_2)_{\mathbf{q}}(\mathbf{q}_0)\mathbf{f}_1(\mathbf{q}_0) + \mathcal{O}(\varepsilon^2)$ gives

$$\begin{aligned}\mathbf{q}_{12} = \mathbf{q}_0 + \varepsilon(\mathbf{f}_1(\mathbf{q}_0) + \mathbf{f}_2(\mathbf{q}_0)) \\ + \frac{\varepsilon^2}{2}((\mathbf{f}_1)_q(\mathbf{q}_0)\mathbf{f}_1(\mathbf{q}_0) + (\mathbf{f}_2)_q(\mathbf{q}_0)\mathbf{f}_2(\mathbf{q}_0)) \\ + \varepsilon^2(\mathbf{f}_2)_{\mathbf{q}}(\mathbf{q}_0)\mathbf{f}_1(\mathbf{q}_0) + \mathcal{O}(\varepsilon^3)\end{aligned}$$

The corresponding expression for $\mathbf{q}_{21}$ is obtained by letting $\mathbf{f}_1$ and $\mathbf{f}_2$ switch places:

$$\begin{aligned}\mathbf{q}_{12} = \mathbf{q}_0 &+ \varepsilon(\mathbf{f}_1(\mathbf{q}_0) + \mathbf{f}_2(\mathbf{q}_0)) \\ &+\frac{\varepsilon^2}{2}((\mathbf{f}_1)_q(\mathbf{q}_0)\mathbf{f}_1(\mathbf{q}_0) + (\mathbf{f}_2)_q(\mathbf{q}_0)\mathbf{f}_2(\mathbf{q}_0)) \\ &+\varepsilon^2(\mathbf{f}_2)_\mathbf{q}(\mathbf{q}_0)\mathbf{f}_1(\mathbf{q}_0) + \mathcal{O}(\varepsilon^3)\end{aligned}$$

$$\begin{aligned}\mathbf{q}_{21} = \mathbf{q}_0 &+ \varepsilon(\mathbf{f}_2(\mathbf{q}_0) + \mathbf{f}_1(\mathbf{q}_0)) \\ &+\frac{\varepsilon^2}{2}((\mathbf{f}_2)_q(\mathbf{q}_0)\mathbf{f}_2(\mathbf{q}_0) + (\mathbf{f}_1)_q(\mathbf{q}_0)\mathbf{f}_1(\mathbf{q}_0)) \\ &+\varepsilon^2(\mathbf{f}_1)_\mathbf{q}(\mathbf{q}_0)\mathbf{f}_2(\mathbf{q}_0) + \mathcal{O}(\varepsilon^3)\end{aligned}$$

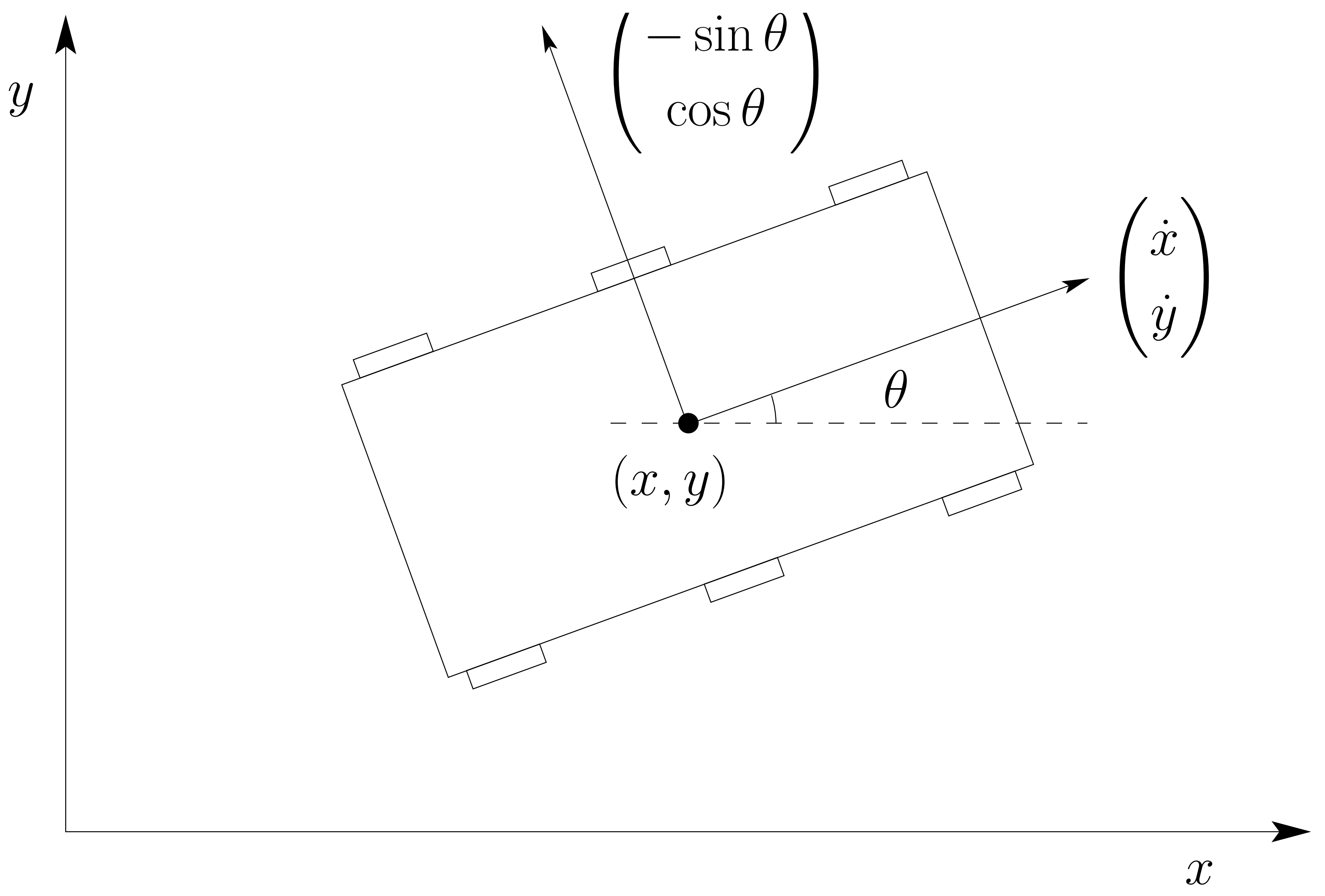


Finally: $\mathbf{f}_3 = \mathbf{q}_{12} - \mathbf{q}_{21} = \varepsilon^2((\mathbf{f}_2)_\mathbf{q}\mathbf{f}_1 - (\mathbf{f}_1)_\mathbf{q}\mathbf{f}_2) + \mathcal{O}(\varepsilon^3)$

Holonomic and Non-holonomic Systems

A non-holonomic example

The differentially driven robot is an example a non-holonomic systems



The directions which the robot can move are given by the condition

$$-\dot{x}\sin\theta + \dot{y}\cos\theta = 0$$

A non-holonomic example

Non-holonomic means that the condition

$$-\dot{x}\sin\theta + \dot{y}\cos\theta = 0$$

can not be written in the form

$$\frac{dG(x, y, \theta)}{dt} = 0$$

for any function $G(x, y, \theta)$.

An example of a holonomic system

Consider a robot moving in three dimensions and the kinematic constraint is

$$\dot{x}x + \dot{y}y + \dot{z}z = 0$$

This constraint is holonomic since it can be rewritten as

$$\frac{1}{2}\frac{d}{dt}(x^2 + y^2 + z^2) = 0$$

which is equivalent to

$$x^2 + y^2 + z^2 = r^2,$$

for some constant r, i.e., the equation of a sphere where the radius is given by the the initial state of the system.

A holonomic example

Two vector field that span the possible direction to move with the restriction $\dot{x}x + \dot{y}y + \dot{z}z = 0$ are

$$\mathbf{f}_1 = \begin{pmatrix} y \\ -x \\ 0 \end{pmatrix} \text{ and } \mathbf{f}_2 = \begin{pmatrix} z \\ 0 \\ -x \end{pmatrix}$$

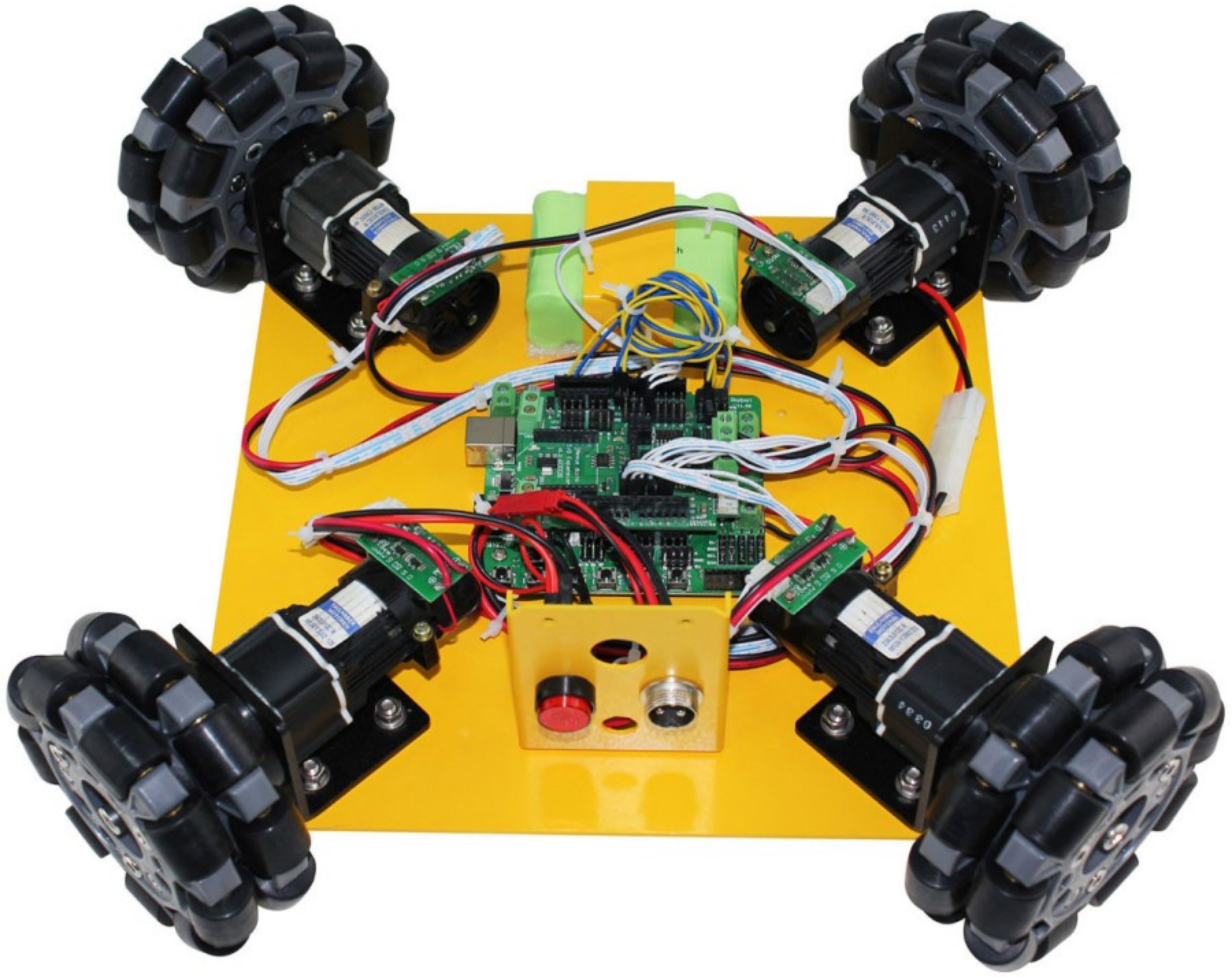
Can we use a the trick describe above to move in a third direction?

The answer is no.

The Lie bracket $[\mathbf{f}_1, \mathbf{f}_2] = \begin{pmatrix} 0 \\ z \\ -y \end{pmatrix}$ also fulfils the restriction and does not add an extra direction.

Conclusion: We can move in any direction as long as we do not leave the sphere.

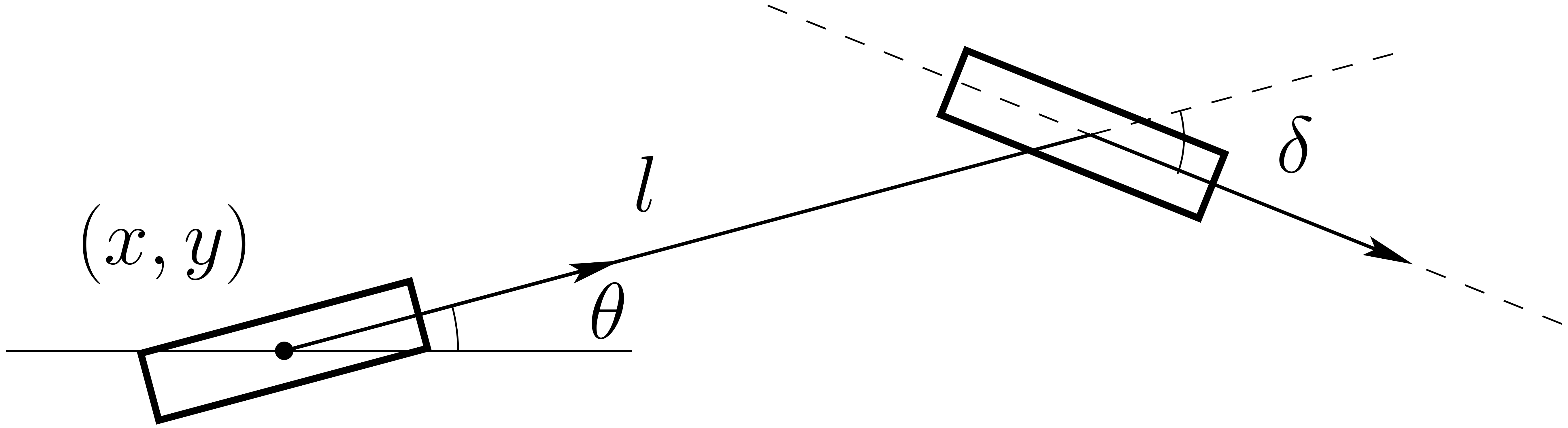
The Omnidirectional Robot



Car with Front Wheel Steering

The Kinematic Single Track Model

A simple kinematic two wheel model of a car with front wheel steering. The model will have at least three states, position (x, y) of the center of the rear wheel, and orientation θ with respect to x-axis.

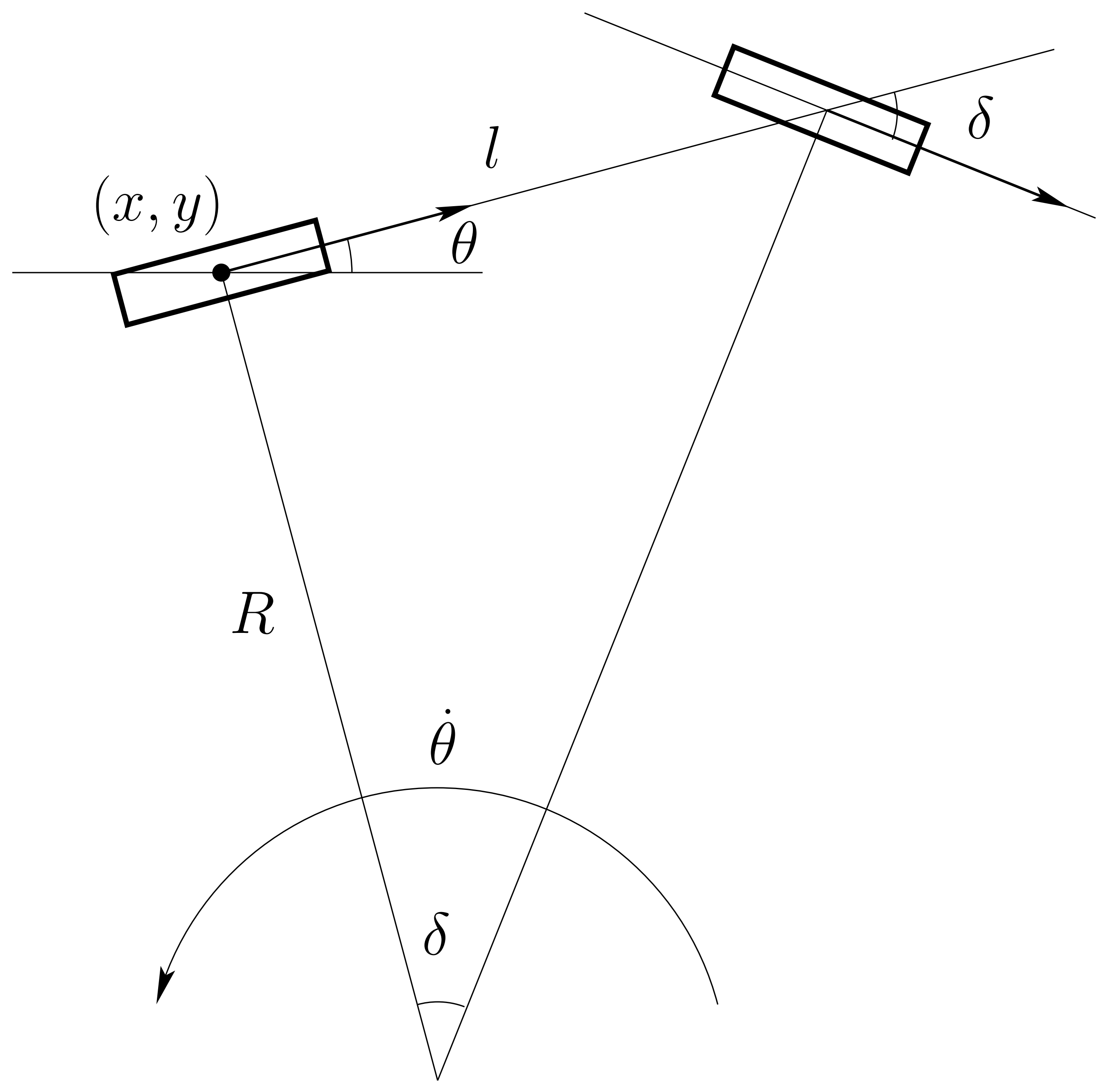


In the the figure, δ is the steering angle and l is the wheel base.

It is assumed that wheels are rolling without slipping, i.e.,
The velocity vector is parallel to the direction of the wheels.

Assume that the car is moving and the speed at the rear wheels is v

$$\dot{x} = v\cos\theta$$
$$\dot{y} = v\sin\theta$$
$$\dot{\theta} = v\frac{\tan\delta}{l}$$

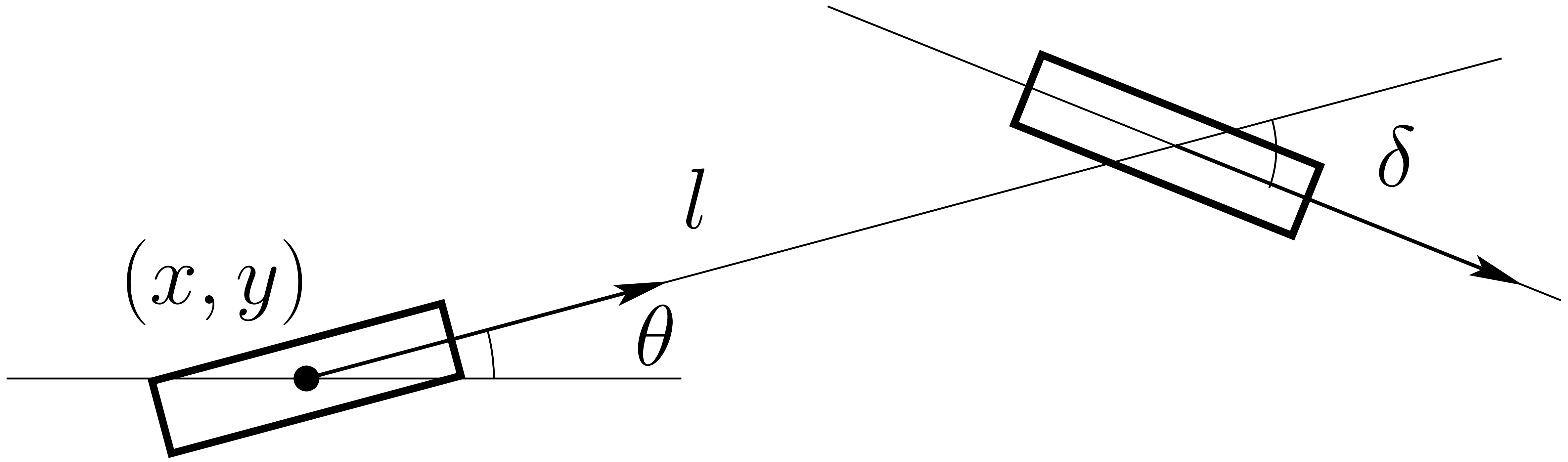


Correction

The steering angle is negative in this figure and δ should be replaced with $-\delta$ in the figure.

The relation $\dot{\theta} = v\tan\delta/l$ follows by eliminating the turning radius R from the equations $-\tan\delta = l/R$ (the triangle) and $-R\dot{\theta} = v$ (circular motion).

In addition to the three differential equations on the previous slide, we need to specify some control inputs.



$$\dot{x} = v\cos\theta$$
$$\dot{y} = v\sin\theta$$
$$\dot{\theta} = v\frac{\tan\delta}{l}$$

One example is to choose the acceleration and tangent of the steering angle

$$\dot{x} = v\cos\theta$$
$$\dot{y} = v\sin\theta$$
$$\dot{\theta} = vu_1/l$$
$$\dot{v} = u_2$$
$$u_1 \in [-\tan\delta_{max}, \tan\delta_{max}]$$

Note that we have added and restriction on the steering angle.

The model

$$\dot{x} = v\cos\theta$$
$$\dot{y} = v\sin\theta$$
$$\dot{\theta} = vu_1/l$$
$$\dot{v} = u_2$$

can be written in the form $\dot{\mathbf{q}} = \mathbf{f}(\mathbf{q}, \mathbf{u})$ with

$$\mathbf{q} = \begin{pmatrix} x \\ y \\ \theta \\ v \end{pmatrix}, \quad \mathbf{f}(\mathbf{q}, \mathbf{u}) = \begin{pmatrix} v\cos\theta \\ v\sin\theta \\ vu_1/l \\ u_2 \end{pmatrix}$$

One drawback with the model is that allows discontinuities in the steering angle. One way to get a smoother solution is to use the differentiated steering angle as input $\dot{\delta}$ and use the steering angle δ as an additional state.

$$
\begin{aligned}
\dot{x} &= v\cos\theta \\
\dot{y} &= v\sin\theta \\
\dot{\theta} &= v\tan\delta/l \\
\dot{\delta} &= u_1 \\
\dot{v} &= u_2 \\
\delta &\in [-\delta_{max}, \delta_{max}] \\
u_1 &\in [-\dot{\delta}_{max}, \dot{\delta}_{max}]
\end{aligned}
$$

The model

$$\dot{x} = v\cos\theta$$
$$\dot{y} = v\sin\theta$$
$$\dot{\theta} = v\tan\delta/l$$
$$\dot{\delta} = u_1$$
$$\dot{v} = u_2$$

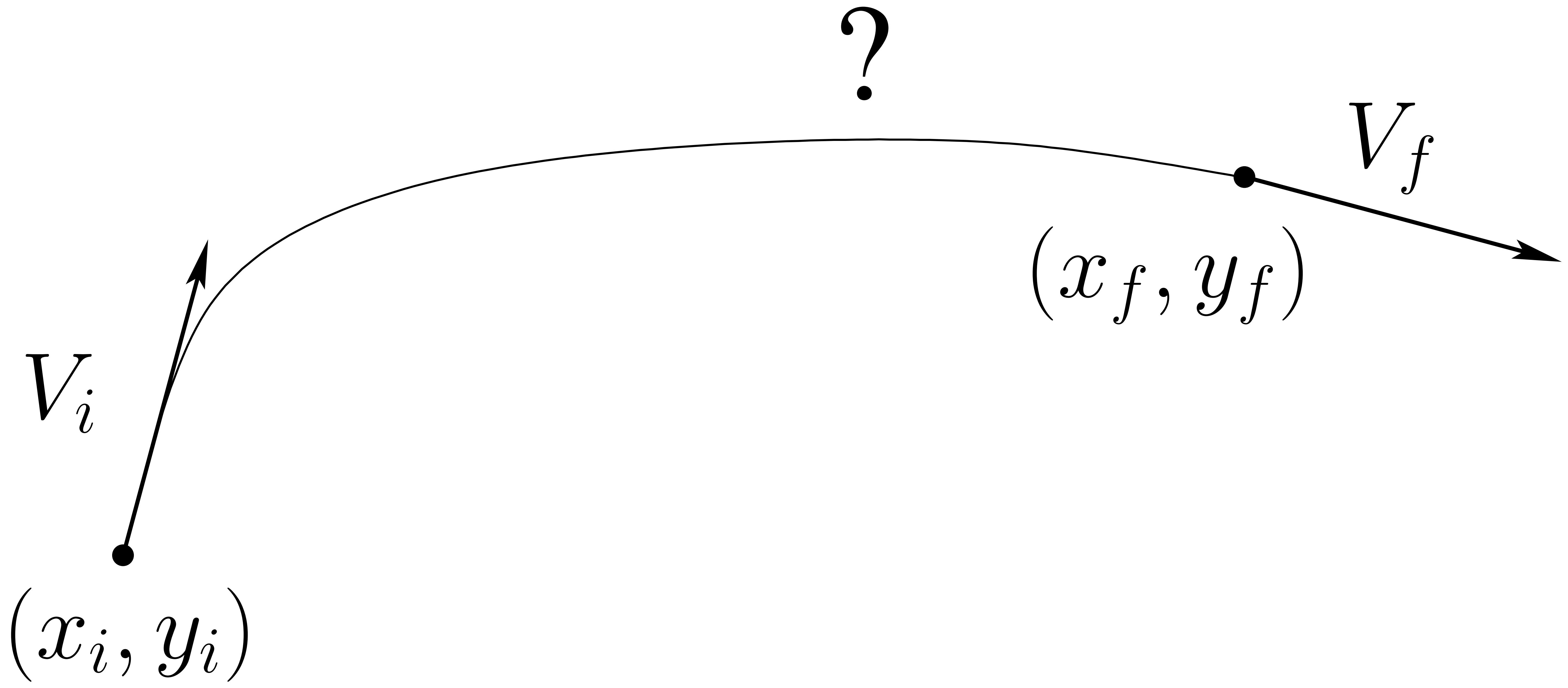
can be written in the form $\dot{\mathbf{q}} = \mathbf{f}(\mathbf{q}, \mathbf{u})$ with

$$\mathbf{q} = \begin{pmatrix} x \\ y \\ \theta \\ \delta \\ v \end{pmatrix}, \quad \mathbf{f}(\mathbf{q}, \mathbf{u}) = \begin{pmatrix} v\cos\theta \\ v\sin\theta \\ v\tan\delta/l \\ u_1 \\ u_2 \end{pmatrix}$$

Motion Planning: Two Classical Problems

Dubins path

One classic motion planning problem: Determine the shortest path between two points with the orientation specified at the initial and final point, and the turning radius limited from below



This problem can be formulated as an optimal control problem with the model

$$
\begin{aligned}
\dot{x} &= v\cos\theta \\
\dot{y} &= v\sin\theta \\
\dot{\theta} &= vu/l \\
v &= 1 \\
u &\in [-\tan\delta_{max}, \tan\delta_{max}]
\end{aligned}
$$

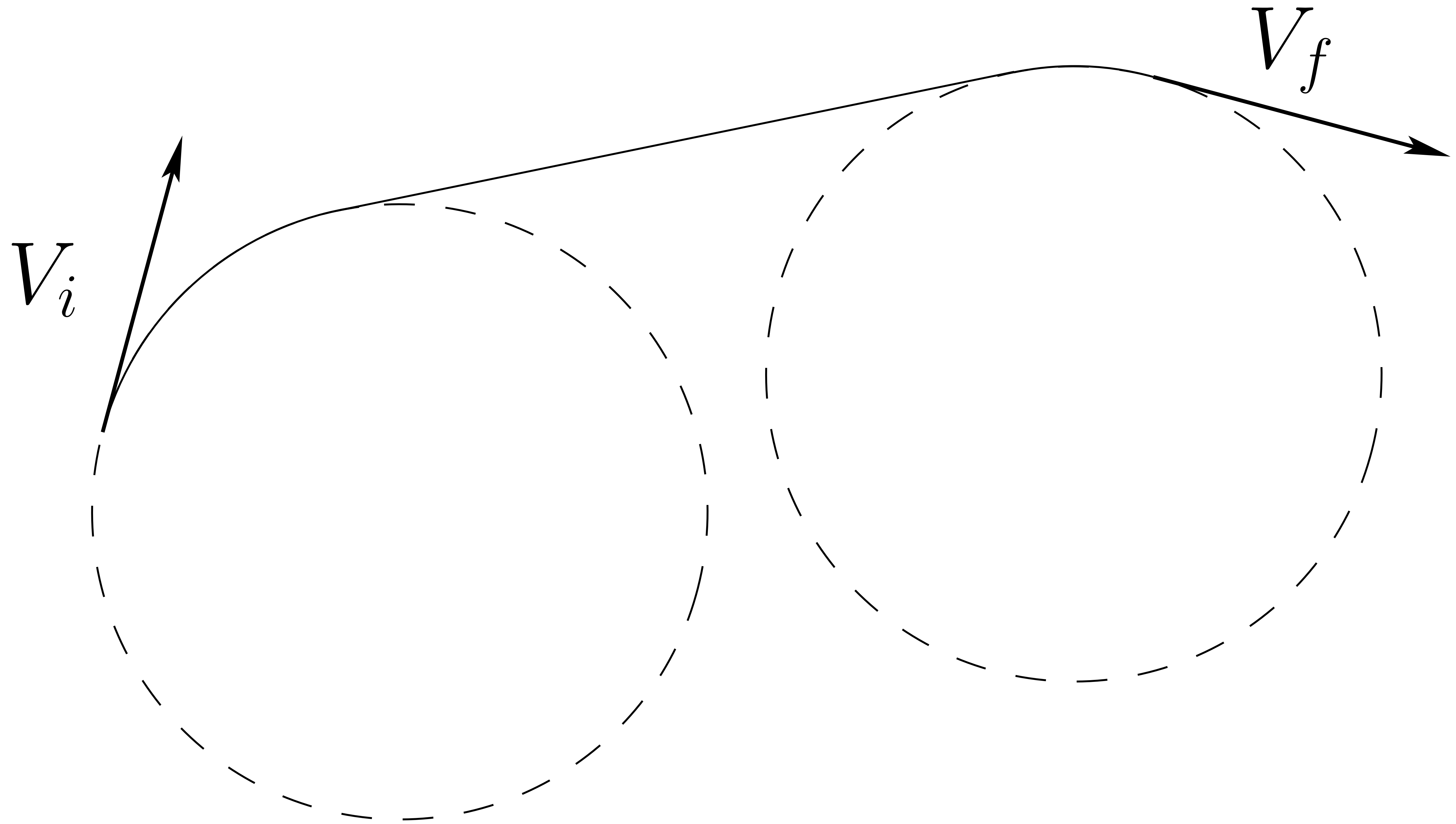
where the objective function is travelling time and $\tan\delta_{max} = l/R_{min}$

This problem was studied in the classical paper

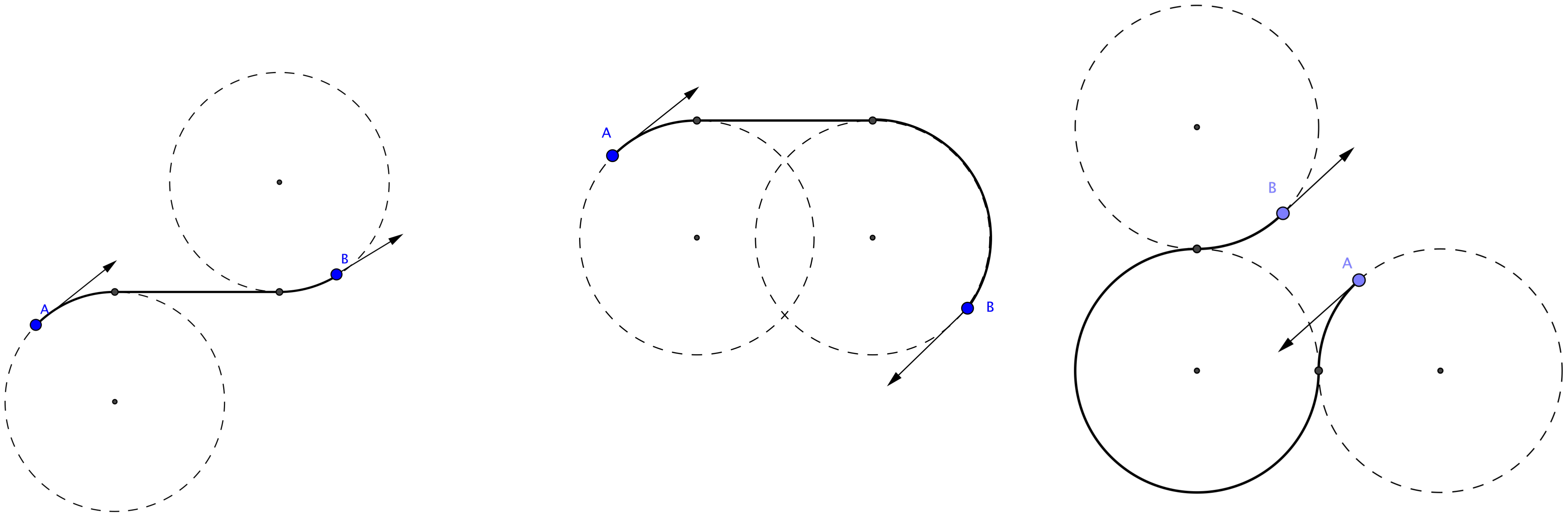
Dubins, L.E. (July 1957). "On Curves of Minimal Length with a Constraint on Average Curvature, and with Prescribed Initial and Terminal Positions and Tangents". *American Journal of Mathematics*. **79** (3): 497–516 doi:10.2307/2372560.

In the paper it was shown that any optimal solution will consist of segments with minimal curvature R_{min} and straight lines.

The solution of the example before is



Three examples of solutions: RSL (right-straight-left), RSR, and LRL



It was shown that there exist three more types of solutions: LSR, LSL, and RLR.

Reeds-Shepp Paths

The results were extended to the case where the car is allowed to move backward and forwards in the paper:

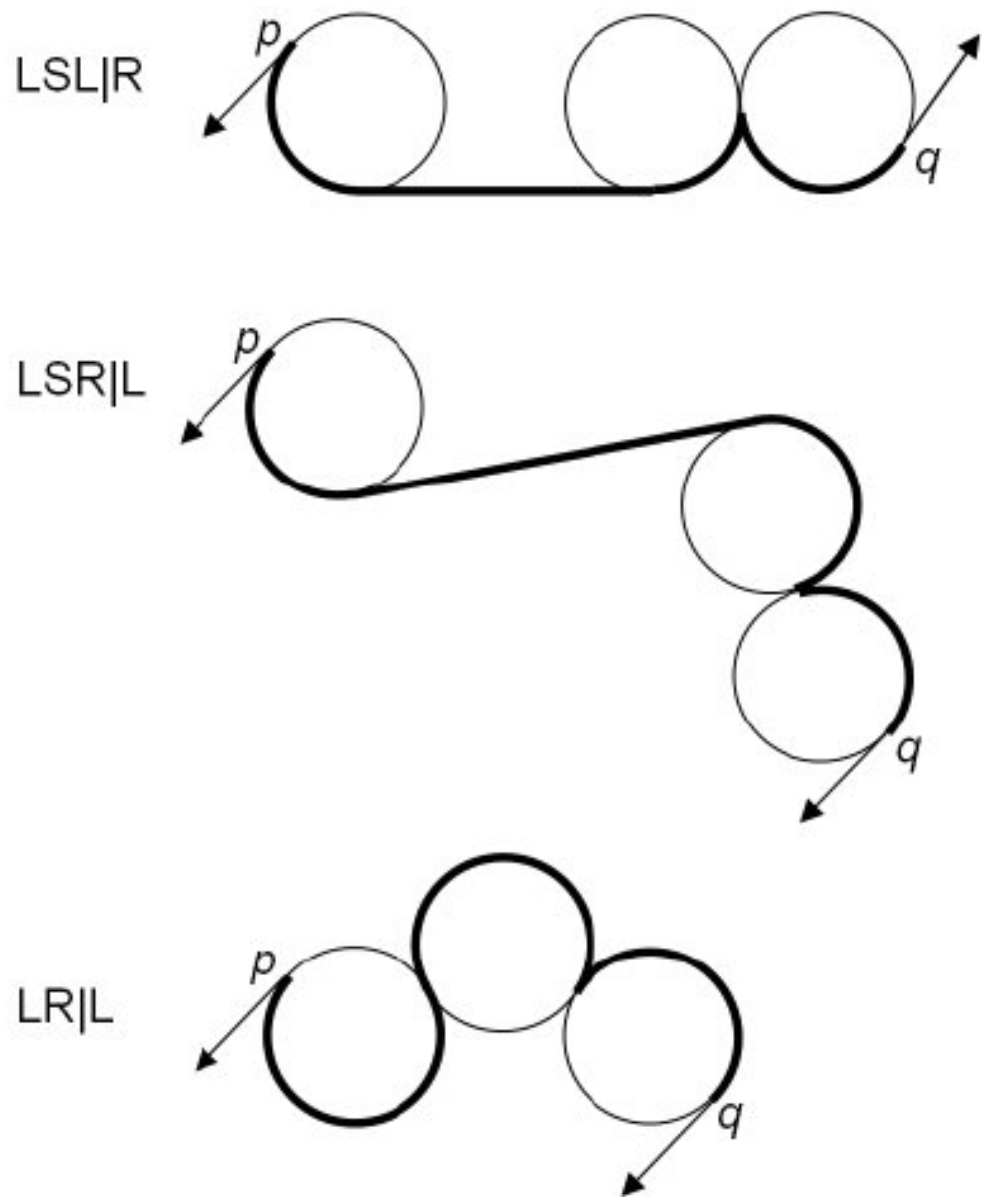
Reeds, J.A. and L.A. Shepp, "Optimal paths for a car that goes both forwards and backwards", Pacific J. Math., 145 (1990), pp. 367–393.

Can be formulated as an optimal control problem with the model

$$\dot{x} = v\cos\theta$$
$$\dot{y} = v\sin\theta$$
$$\dot{\theta} = vu/l$$
$$v \in \{-1, 0, +1\}$$
$$u \in [-\tan\delta_{max}, \tan\delta_{max}]$$

and time as objective function.

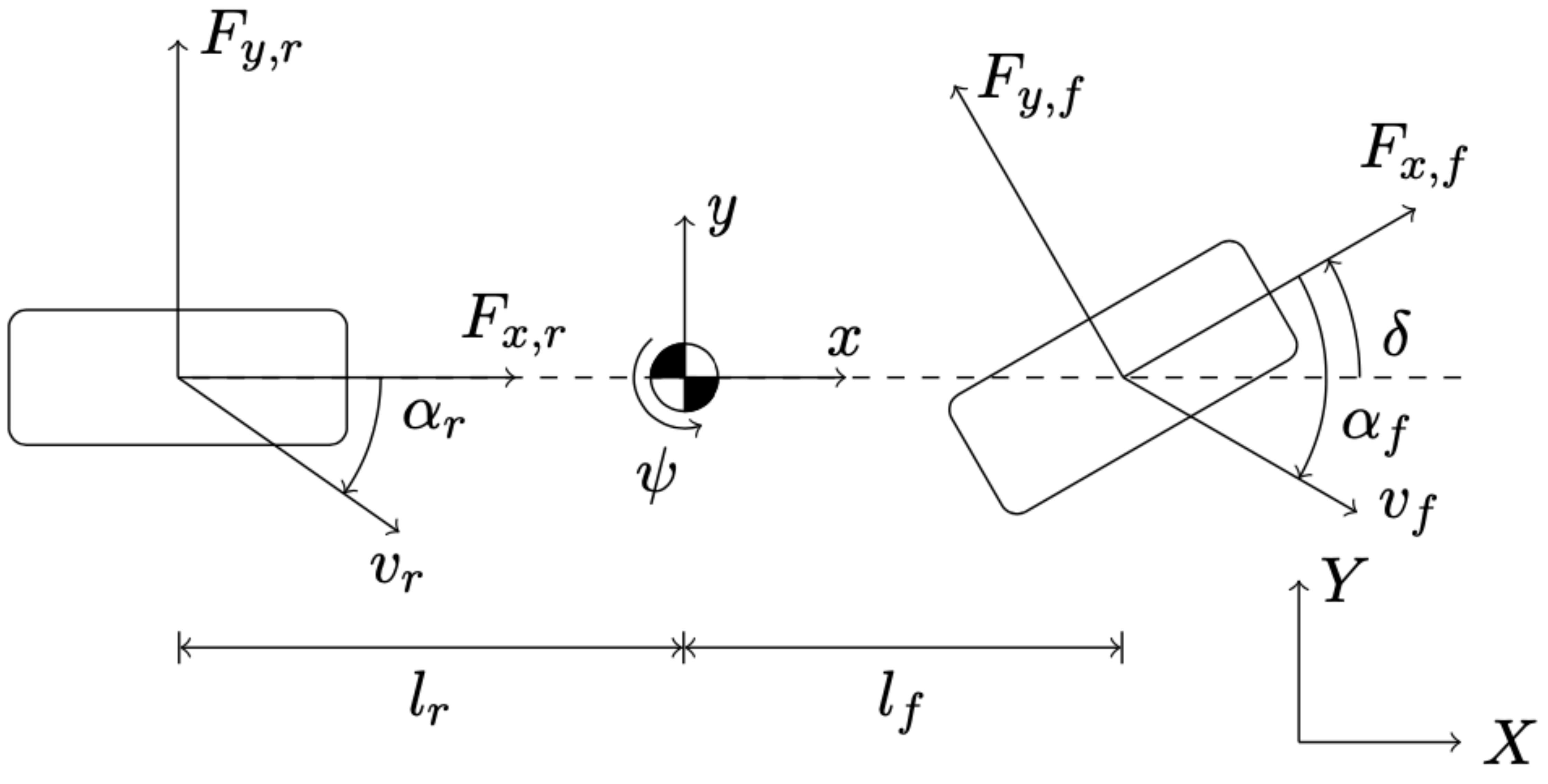
Some examples of optimal Reeds-Shepp paths:



A dynamic vehicle model (used in Lecture 7)

Dynamic Vehicle Model (1/4)

- Consider the **single-track vehicle model** (lumped wheels on each axle).
- **Front steering** and **front-wheel driven**.



$$m(\dot{v}_x - v_y\dot{\psi}) = F_{x,f}\cos(\delta) + F_{x,r} - F_{y,f}\sin(\delta),$$

$$m(\dot{v}_y + v_x\dot{\psi}) = F_{y,f}\cos(\delta) + F_{y,r} + F_{x,f}\sin(\delta),$$

$$I_Z\ddot{\psi} = l_f F_{y,f}\cos(\delta) - l_r F_{y,r} + l_f F_{x,f}\sin(\delta)$$

Dynamic Vehicle Model (2/4)

- The **slip angles** (angle between the velocity vector and the direction of the wheel) are given by:

$$\alpha_f = -\arctan\left(\frac{v_{f,y}}{v_{f,x}}\right), \qquad \alpha_r = -\arctan\left(\frac{v_{r,y}}{v_{r,x}}\right)$$

- A simple **tire–road interaction model** for normal driving with linear tire stiffnesses is adopted:

$$F_{y,f} = C_{\alpha,f}\alpha_f, \qquad F_{y,r} = C_{\alpha,r}\alpha_r$$

Dynamic Vehicle Model (3/4)

- A **path parameter** is introduced to describe the traversal along the reference path (traversal computed in the MPC):

$$\dot{s} = \frac{\mathrm{d}s}{\mathrm{d}t}$$

- The **vehicle position** in the **global coordinate frame** is obtained by integration of the quantities:

$$\begin{pmatrix} \dot{p}_X \\ \dot{p}_Y \end{pmatrix} = \begin{pmatrix} \cos(\psi) & -\sin(\psi) \\ \sin(\psi) & \cos(\psi) \end{pmatrix} \begin{pmatrix} v_x \\ v_y \end{pmatrix}$$

Faulwasser, T., & Findeisen, R: "Nonlinear model predictive control for constrained output path following". IEEE Transactions on Automatic Control, 61(4), 1026-1039, 2015.

Dynamic Vehicle Model (4/4)

- Collect the **states** and the **control inputs** in the vectors:

$$x(t) = \begin{pmatrix} p_X & p_Y & \psi & v_x & v_y & \dot{\psi} & s \end{pmatrix}^\mathrm{T}$$
$$u(t) = \begin{pmatrix} \delta & F_{x,f} & F_{x,r} & \dot{s} \end{pmatrix}$$

- With these variables, the **vehicle dynamics** can be written as an **explicit ordinary differential equation system** as:

$$\dot{x}(t) = f_{\mathrm{car}}(x(t), u(t))$$