• Chained forms

This is a canonical form for Kinematic models of WMR (Wheeled Mobile Robots)

• Note1: Not all WMR's can be put this form, but most majority can.

The structure of chained forms is given as below:

$$\dot{z} = \begin{pmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \vdots \\ \dot{z}_n \end{pmatrix} = \begin{pmatrix} \upsilon_1 \\ \upsilon_2 \\ z_2 \upsilon_1 \\ \vdots \\ z_{n-1} \upsilon_1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ z_2 \\ \vdots \\ z_{n-1} \end{pmatrix} \upsilon_1 + \begin{pmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \upsilon_2 \quad (1)$$

According to (1), we can say that, this system is particular driftless and it has 2 inputs.

In equation (1): $z \in \mathbb{R}^n \rightarrow$ state variable
$\upsilon \in \mathbb{R}^2 \rightarrow$ inputs

Block scheme of this structure is given as below

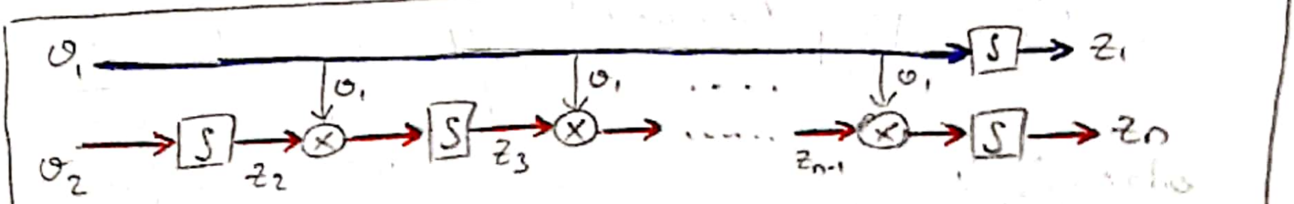


Figure1: Block scheme of chained form

Because there is chain of integrators the structure is called chained forms. Notice that, the first channel (blue) has only 1 simple integrator and the second channel (red) has the chain of integrators.

This form is controllable. How can we prove it?

We need to prove Accessibility Algebra has dimension n. For simplicity we will denote vector fields as below:

$$\mathcal{g}_1 = \begin{pmatrix} 1 \\ 0 \\ z_2 \\ \vdots \\ z_{n-1} \end{pmatrix} \quad \mathcal{g}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad (2)$$

In order to prove controllability we'll need to show that, $\mathcal{g}_1$, $\mathcal{g}_2$ and all possible Lee-Brackets will span a space with dimension of n. In other words, we need to prove the following equation:

$$\dim \Delta_A = n \quad (3)$$

In (3) :

$$\Delta_A = \{ [\mathcal{g}_1, \mathcal{g}_2], [\mathcal{g}_1, [\mathcal{g}_1, \mathcal{g}_2]], \underbrace{\ldots}_{\text{all possible Lee Brackets.}} \} \quad (4)$$

There exist some Necessary and Sufficient (N & S) conditions for transforming generic driftless system, which is given by the equation (5), into the system is given by the equation (6).

$$\dot{q} = g_1(q)\, u_1 + g_2(q)\, u_2 \quad (5)$$

$$\dot{z} = \mathcal{g}_1(z)\, v_1 + \mathcal{g}_2(z)\, v_2 \quad (6)$$

(6) is the generic representation of chained form which was represented by the equation (1).

There is interesting fact that, all systems in form of equation (5) with $n \leq 4$ dimension satisfy those N & S conditions. For instance, unicycle ($n=3$) and bicycle ($n=4$) can be put into this form. When $n > 4$ (for instance, tricycle with trailer) may or may not be put in chained form (i.e it is not guaranteed).

- How can we transform?

→ 1. Coordinate transformation:

At this point, we need to transform variables from q to z. In order to do this transformation we'll use the following function:

$$z = \alpha(q) \quad (7)$$

→ 2. Input transformation:

At this point, we need to transform inputs from u to v. It'll be done by the following function:

$$v = \beta(q)\,u \quad (8)$$

In (8), we can see that β depends on q (state) variables which denotes the configuration of robot. It's actually a FEEDBACK transformation, because it needs the state of the robot.

• Transformation on unicycle

Unicycle robot is defined by the following equations :

$$\dot{x} = v \cos\theta \quad (9)$$
$$\dot{y} = v \sin\theta \quad (10)$$
$$\dot{\theta} = \omega \quad (11)$$

$$q = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix} \quad (12) \qquad u = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} v \\ \omega \end{pmatrix} \quad (13)$$

q → state variable (configuration of robot)

u → input variable

(9), (10), (11) were generated in the previous lectures (check them).

As we mentioned before, because of n=3, unicycle can be put in chained form and in chained form n will be 3. Thus using the generic representation, we can rewrite the system with n=3, as following:

$$\dot{z}_1 = v_1 \quad (14)$$
$$\dot{z}_2 = v_2 \quad (15)$$
$$\dot{z}_3 = z_2 v_1 \quad (16)$$

Note : v in the equations (9), (10), (13) are driving velocity of the unicycle and, v_1 and v_2 are inputs for chained form representation. Do not be CONFUSED! They aren't the same thing.

AMR Lecture2 part2

The constructive N&S conditions are used for transforming the equations (9),(10),(11) to (14),(15),(16), respectively. However, these conditions are not the part of lecture. That is why we'll show only the results.

Equations (9),(10),(11) becomes (14),(15),(16), respectively, letting:

$$z_1 = \theta \quad (17)$$
$$z_2 = x\cos\theta + y\sin\theta \quad (18)$$
$$z_3 = x\sin\theta - y\cos\theta \quad (19)$$

If we analyze (17),(18),(19) we can see that z is defined by function of q, as it shown in the equation (7).

Geometrical interpretation of z_1, z_2, z_3 variables can be given as in figure 2.

It's seen from (18) and (19) that, z_2 and z_3 are coordinates of the contact point in rotated frame which is alligned with sagital axis of unicycle. In other words, z_2, z_3 are coordinates of unicycle in the moving frame.

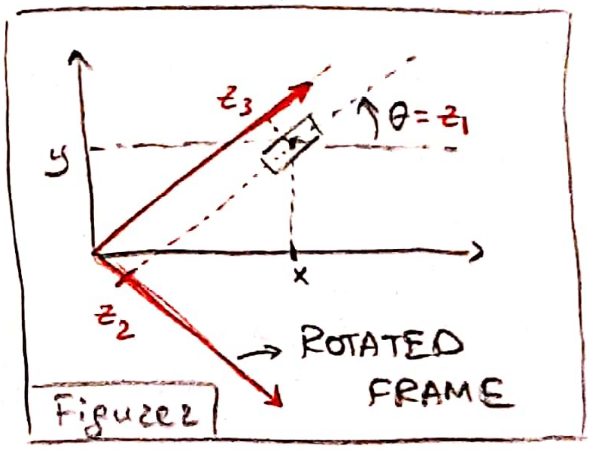


Figure2

On the other hand, from the equation (17), we know that z_1 is θ.

Note: When wheel moves, ROTATED FRAME also moves according to wheel's motion.

MNR

The chained form that is given by (14), (15), (16) is called as (2,3) chained form. It is called in that way, because of 2 inputs and 3 states.

Now we can compute the chained form equations by applying coordinate transformation which will give us also the results of input transformations.

=> From (14) and (17):

$$\dot{z}_1 = \dot{\theta} = \omega = v_1 \quad => \quad v_1 = \omega \ (20)$$

=> From (15) and (18):

$$\dot{z}_2 = \dot{x}\cos\theta - x\sin\theta\cdot\dot{\theta} + \dot{y}\sin\theta + y\cos\theta\cdot\dot{\theta} \ (21)$$

From (9) and (10) we can use the following substitutions:

$$(9) => \dot{x}\cos\theta = v\cos\theta\cos\theta = v\cos^2\theta \ (a)$$

$$(10) => \dot{y}\sin\theta = v\sin\theta\sin\theta = v\sin^2\theta \ (b)$$

Using (a) and (b) in (21):

$$\dot{z}_2 = v\cos^2\theta + v\sin^2\theta - (x\sin\theta - y\cos\theta)\dot{\theta} =$$

$$= v - (x\sin\theta - y\cos\theta)\dot{\theta} \ (22)$$

Using (20) and (19) in (22):

$$\dot{z}_2 = v - v_1 z_3 = v_2 \ (23)$$

According to (20) and (23), we have already done the input transformation:

$$\boxed{\begin{array}{l} (20) => v_1 = \omega \\ (23) => v_2 = v - v_1 z_3 \end{array}}$$

MAR

⇒ From (16) and (13) :

$$(13) \Rightarrow \dot{z}_3 = \dot{x}\sin\theta + x\cos\theta\cdot\dot{\theta} - \dot{y}\cos\theta + y\sin\theta\cdot\dot{\theta} \quad (24)$$

From (9) and (10), If we apply (a) and (b) in (24):

$$\dot{z}_3 = v\sin\theta\cos\theta - v\sin\theta\cos\theta + \dot{\theta}(x\cos\theta + y\sin\theta) =$$
$$= (x\cos\theta + y\sin\theta)\dot{\theta} \quad (25)$$

Using (18) and (20) in (25):

$$\dot{z}_3 = (x\cos\theta + y\sin\theta)\dot{\theta} = z_2 v_1 \quad (26)$$

It's seen that (26) is the same with (16).

• Use of chained forms.

Chained forms are good for planning and control because

→ it applies to a large numbers of robots

→ it can be easily integrated under appropriate inputs

↳ Because the system is nonlinear in general it is not integrated. By saying appropriate inputs, we mean to represent variable in PIECEWISE-CONSTANT (Figure 3).

By using this method we get linear functions in each sample that can be integrated.

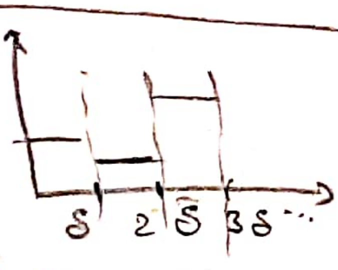


Figure 3

MNV (7)

- Path and Trajectory Planning (for WMR)

Outline will be :

→ Definition of Path, Trajectory and Timing Law

→ Differential Flatness

→ Path Planning

→ Trajectory Planning.

- Definition of Path, Trajectory and Timing Law.

The configuration space with initial and final configurations which are q_i and q_f, respectively. (Figure 4)

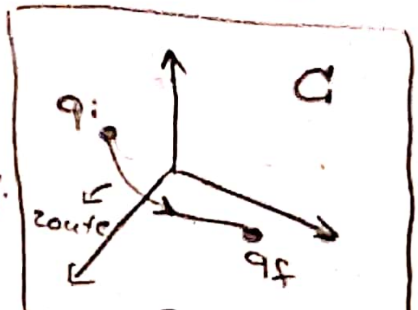


Figure 4

Path is a description of q as a function of parameter s. s is the path parameter (typically arc length) that can change in certain interval $s \in [s_i, s_f]$.

$$q_i = q(s_i) \quad (27) \quad \text{and} \quad q_f = q(s_f) \quad (28)$$

Trajectory is a description of q as a function of time t. t is the time that changes in given interval, such as $t \in [t_i, t_f]$.

$$q_i = q(t_i) \quad (29) \quad \text{and} \quad q_f = q(t_f) \quad (30)$$

Note 2: Time is special parameter because it can't revert. In terms of s, we can move back and forth. Contrarily in terms of t we can't do it.

If we consider path, we consider only the geometry of motion. In other words, we put only geometrical constraints.

If we consider trajectory, we consider geometry and speed of motion.

On the other hand, we can separate the given trajectory to 2 parts, which are PATH and TIMING LAW. It can be described as below:

$$q = q(t) \quad , \text{where} \quad t \in [t_i, t_f] \quad (31)$$

(31) can be shown in 2 parts:

① Path : $q = q(s) \quad , \quad s \in [s_i, s_f]$ (32)

$q_i = q(s_i)$ (33) $\quad q_f = q(s_f)$ (34)

② Timing Law : $s = s(t) \quad , \quad t \in [t_i, t_f]$ (35)

$s_i = s(t_i)$ (36) $\quad s_f = s(t_f)$ (37)

To sum up we can define path planning and trajectory planning as following.

-> Path planning.

Given $q_i, q_f \in C$ find a path $q = q(s)$, $s \in [s_i, s_f]$ such that brings the robot from $q(s_i) = q_i$ (initial config.) to $q(s_f) = q_f$ (final config.).

It is geometric problem.

Trajectory planning.

Given $q_i, q_f \in C$, find a trajectory $q(t)$, $t \in [t_i, t_f]$ such that brings the robot from $q_i = q(t_i)$ (initial config.) to $q_f = q(t_f)$ (final configuration).

It is kinematic problem because it involves time.

When we'll analyze those plannings we can neglect presence of obstacles. We consider obstacles in motion planning.

- Trajectory planning addresses to 2 approaches.

① ALL-IN-ONE
Directly plan a trajectory $q(t)$

② DECOUPLED (two-phase)
First plan a path, then choose a timing law. (It is more powerful ↳ and flexible).

NH

• Nonholonomic constraints

We'll analyze that these kind of constraints make the planning methods more difficult or not.

We know that RWS constraint causes local mobility restriction (RWS is one of the NH constraints)

In other words, except generalized velocities that belong to the Null Space of Constraint Matrix $(A(q))$, not all generalized velocities are admissible. Because of that, not all trajectories in C are admissible. Admissible generalized velocities which belong to $N(A^T(q))$ can be given by the equation (38):

$$\dot{q} \in N(A^T(q)) \quad (38)$$

On the other hand, are paths are restricted because of NH constraints? Because this restrictions are not matter of velocity but matter of geometry, paths are also restricted. To sum up, BECAUSE OF NON-HOLONOMIC CONSTRAINTS, VELOCITIES, TRAJECTORIES AND PATHS ARE RESTRICTED!

Example

Configuration Space of Unicycle is given with the Figure 5. It's seen from the figure that, any trajectory along the Zero Motion Line (ZML) is not admissible. Additionally, there is not any velocity component along that line. That is why, we can say no matter which trajectory is chosen along ZML it is not admissible. Because there is not any possible velocity which allows us to move along that line, we can certainly say that, the PATH ALONG THE ZML is not ACTUALLY ADMISSIBLE!

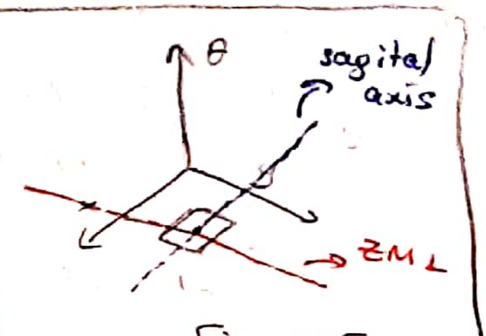


Figure 5

To sum up, we see that it is not the matter of speed, it is the matter of the geometry.

Before showing this as analytically we need to understand the concept.

What are these $\dot{q}$ (generalized velocities)?

It is time derivative of the configuration of robot.

$$\dot{q} = \frac{dq}{dt} = \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{pmatrix} \quad (39)$$

We have already said that the trajectory can be separated to 2 parts, which are path and timing law.

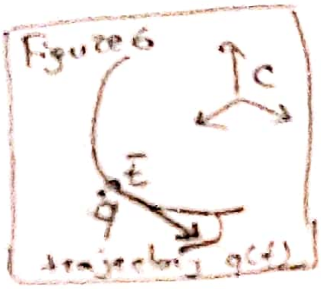
$$q = q(t) = q(s(t)) \quad (40)$$

(40) says that, q can be seen as a function of t through s.

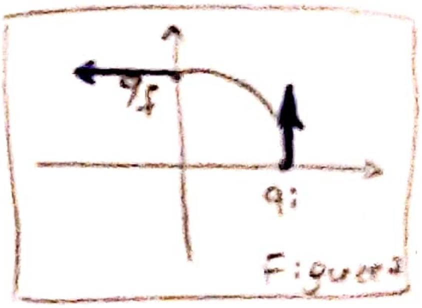
Thus we can rewrite (39) as below:

$$\dot{q} = \frac{dq}{dt} = \frac{\partial q}{\partial s} \frac{\partial s}{\partial t} = q' \dot{s} \quad (40)$$

q' is the tangent vector to the path and $\dot{s}$ is the change of the path with respect to the time. Notice that, q' shows the direction and magnitude of that direction vector depends on $\dot{s}$ (Figure 6).



In the figure $\dot{q}$ is the vector that is a tangent to the trajectory at time t. As we said, trajectory change wrt to the path param defines the direction of $\dot{q}$ vector (q') and change of path with respect to time defines the magnitude of $\dot{q}$ vector ($\dot{s}$). In other words, bigger $\dot{s}$ is larger vector.



We can show another example in $\mathbb{R}^2$ as it is demonstrated in the Figure 7.

We consider path parameter as arc length. Notice that, if path parameter is chosen as arc length, then q' will have always unit magnitude. Trajectory wrt path can be given as below:

$$q(s) = \begin{pmatrix} \cos s \\ \sin s \end{pmatrix}, \quad s \in [0, \tfrac{\pi}{2}] \quad (41)$$

In (41), s_i and s_f are initial and final path parameters. q' can be find as below:

$$q' = \begin{pmatrix} -\sin s \\ \cos s \end{pmatrix} \quad (42)$$

If we use s_i and s_f, we can get blue vectors in the figure 7.

$$s_i \rightarrow \quad q'(s_i) = \begin{pmatrix} -\sin(0) \\ \cos(0) \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \Rightarrow \|q'_i\| = 1 \quad (42)$$

$$s_f \rightarrow \quad q'(s_f) = \begin{pmatrix} -\sin(\pi/2) \\ \cos(\pi/2) \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \end{pmatrix} \Rightarrow \|q'_f\| = 1 \quad (43)$$

To sum up, we can say that q' is the tangent vector to the trajectory.

Note 3: As we said, because s is the arc-length $\|q'\|$ is always one (look at (42) and (43))

Now we can show how trajectory and path are restricted by NH constraints, analytically. We know that RWS constraint is the following:

$$A^T(q)\,\dot{q} = 0 \quad (44)$$

If we apply (40) to (44):

$$A^T(q)\,\dot{q} = A^T(q)\,q'\,\dot{s} = 0 \quad (45)$$

Because $\dot{s}$ is a scalar value and because this constraint must be satisfied for all possible timing law, we cannot rely on $\dot{s}$, in order to satisfy (45). Thus, the following must be satisfied in order to satisfy the equation (45):

$$A^T(q)\,q' = 0 \quad (46)$$

To sum up, we can say RWS constraint restricts the trajectories because of the restrictions on tangent vector q'. That's why; these restrictions are matter of the geometry.

In other words, Kinematic Constraints actually are constraints on Tangent Vectors to the path.

In order to show constraints on tangent vector we will find admissible q' for a given trajectory with RWS constraint. In order to do that we will write again the Basis of $N(A^T(q))$, which will be the same vector fields as before. For m dimensional $N(A^T(q))$, it will be $\{g_1(q), \dots, g_m(q)\}$.

We can show admissible tangent vector as a linear combination of those vector fields:

$$q' = \sum_{j=1}^{m} g_j(q)\, \tilde{u}_j \quad (46)$$

We have to emphasize 2 things related with (46). For this purpose, we must recall $\dot{q}$ representation that we derived before for generalized velocities under RWS constraints.

$$\dot{q} = \sum_{j=1}^{m} g_j(q)\, u_j \quad (47)$$

→ 1. u_j and $\tilde{u}_j$ are different coefficients. Because they are inputs to the model, we can say also, they are different inputs, as well.

→ 2. In (46), time derivatives are replaced with derivatives wrt s. By writing (46), we are defining admissible tangent vectors, but not admissible generalized velocities.

(46) is called as geometric version of the kinematic model or geometric model. It is evolution of the model wrt s.

• Relationship between velocity inputs (u_j) and geometric inputs ($\tilde{u}_j$).

If we multiply (46) by $\dot{s}$:

$$(46) :> \quad q'\dot{s} = \sum_{j=1}^{m} g_j(q)\, \tilde{u}_j \dot{s} = \dot{q} = \sum_{j=1}^{m} g_j(q)\, u_j \quad (48)$$

According to (48):

$$u_j = \tilde{u}_j \dot{s} \quad (49).$$

It's seen from (49) that velocity inputs (u_j) are the multiplication of geometric inputs ($\tilde{u}_j$) and $\dot{s}$.

• Interpretation of (46)

We know that Kinematic model predicts which kind of generalized velocity that is used by robot, according to the arbitrarily chosen velocity inputs (u_j). This is given by the equation (47). On the other hand, because (47) is the motion in time, it also predicts which kind of trajectory will the robot follow.

Besides, geometric model will tell us the path that robot will have. In other words, geometric model predicts the evolution of q wrt s according to the arbitrarily chosen geometric velocities ($\tilde{u}_j$).

To sum up, interpretation of (46) is:

-> Choose any $\tilde{u}_j$ ($j = 1, \dots, m$) and this model will predict the resulting path.

That is why; (46) is good for path planning and (47) is good for trajectory planning.

• Unicycle.

Because of the same $g_1(q)$ and $g_2(q)$ we can write q' as below:

$$q' = \begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix} = g_1(q)\,\tilde{v} + g_2(q)\,\tilde{\omega} = \begin{pmatrix} \cos\theta \\ \sin\theta \\ 0 \end{pmatrix}\tilde{v} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}\tilde{\omega} \quad (50)$$

In (50), q' is the evolution of q with respect to s. That is why we write x', y', θ' instead of $\dot{x}, \dot{y}, \dot{\theta}$ (they are not same). $g_1(q)$ is DRIVE and $g_2(q)$ is STEER vector fields. $\tilde{v}$ and $\tilde{\omega}$ are geometric velocity inputs.

• Example

Configuration space of Unicycle is demonstrated by the Figure 8.

Note 4: q' in any direction along the ZML would violate the constraint.

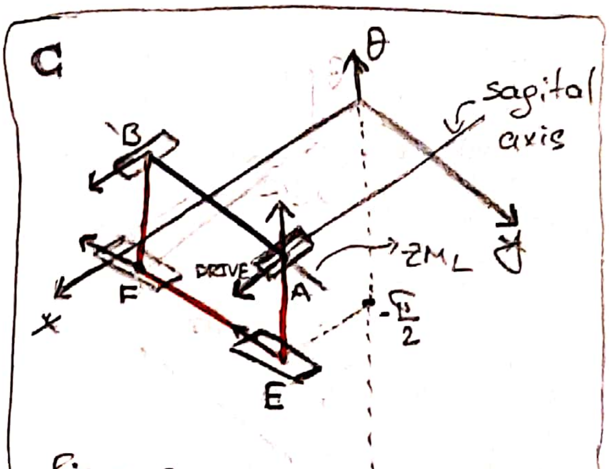


Figure 8

We know that unicycle can't go from A to B along the ZML (blue path). In order to go from A to B:

→ steer till $\theta = -\frac{\pi}{2}$ and DRIVE vector will get the direction as at the point E.

→ drive till the point F without steering

→ steer till $\theta = 0$ and it will get to the point B.

If we look at the cartesian "foot print" (blue) of the motion we will see that it is along the ZML. However, this is not the Configuration Space motion (red one is the C-Space motion).

Note 5:

Do Not Confuse about distinguishing Cartesian Space Motion (blue) and Configuration Space Motion (red)!

• Differential Flatness.

→ A system is differentially flat if states and inputs can be reconstructed from the evolution of some outputs (FLAT outputs) of system.

We can apply differential flatness to any non-linear system. In this case we will apply it to driftless system which is given by the equation (47) or (46)

Kinematic model: (47) => $\dot{q} = \sum_{i=1}^{m} g_j(q) u_j = G(q) u$

Geometric model: (46) => $q' = \sum_{i=1}^{m} g_j(q) \tilde{u}_j = G(q) \tilde{u}$

* Definition

Kinematic model is differentially flat, if there exists a set of outputs, such that,

$$W = h(q) \quad (51)$$

such that,

$$q = \eta(w, \dot{w}, \ddot{w}, \ldots, w^{(r)}) \quad (52)$$
$$u = \mu(w, \dot{w}, \ddot{w}, \ldots, w^{(r)}) \quad (53)$$

$W(q)$ is FLAT output and r is the certain order of derivative. (52) and (53) are (ALGEBRAIC) RECONSTRUCTION OF STATE / INPUTS.

MNV

Scanned with CamScanner

For geometric model, the same method is available. However in that case, derivations will be with respect to s:

$$q = \eta(w, w', w'', \dots, w^{(r)'}) \quad (54)$$
$$\tilde{u} = \mu(w, w', w'', \dots, w^{(r)'}) \quad (55)$$

• Unicycle

In order to show the model, we will recall equations (9), (10) and (11):

$$(9) \Rightarrow \dot{x} = v\cos\theta$$
$$(10) \Rightarrow \dot{y} = v\sin\theta$$
$$(11) \Rightarrow \dot{\theta} = \omega$$

This model is diferentially flat and flat outputs are cartesian coordinates of the contact point:

$$w = \begin{pmatrix} x \\ y \end{pmatrix} \quad (56)$$

Proof: How do we do reconstruction for unicycle?

1. State reconstruction. $(w \to q)$

Because of (56):

$$\omega_1 = x \quad (57)$$
$$\omega_2 = y \quad (58)$$

Now we need to define θ. It can be written as below:

$$\theta = \arctan\left(\frac{\dot{y}}{\dot{x}}\right) \quad (59)$$

Because, we will loose the sign by using (59), we will define θ as below.

$$\theta = ATAN2(\dot{y}, \dot{x}) + k\pi \quad (60)$$

→ We add $k\pi$ because $\tan(\theta) = \tan(\pi + \theta)$

→ $k = 0$ if $v > 0$ and $k = 1$ if $v < 0$.

MNY

By using (57), (58) and (60) we reconstructed states (q) by using flat outputs (w).

→ Input reconstruction ($w \to \vartheta, \omega$)

Reconstruction of ϑ can be done as below:

$$\vartheta = \pm\sqrt{\dot{x}^2 + \dot{y}^2} \quad (61)$$

In (61), $\vartheta > 0$ (+) for forward motion and $\vartheta < 0$ (−) for backward motion.

Reconstruction of ω can be done as below;

Because of (10) we know that ω is $\dot{\theta}$.

Thus if we take derivative of (60) we can write it as below:

$$\omega = \dot{\theta} = \frac{1}{1 + \left(\frac{\dot{y}}{\dot{x}}\right)^2} \frac{\ddot{y}\dot{x} - \ddot{x}\dot{y}}{\dot{x}^2} = \frac{\ddot{y}\dot{x} - \ddot{x}\dot{y}}{\dot{x}^2 + \dot{y}^2} \quad (62)$$

In (62), it is seen that $r = 2$.

⇒ Few comments:

- From the cartesian trajectory which are $x(t)$ and $y(t)$, we can derive state trajectory ($q(t)$) and input history ($u(t)$).
- The possibility of reconstruction is lost when

$$\dot{x} = \dot{y} = 0 \quad (63)$$

In other words motion degenerates to a point.

If we check (60), we'll see that ATAN2(0,0) is undefined (state reconstruction is lost) and if we check (62), we'll see that there will be zero division (input reconstruction is lost).

Interpretation of losing the possibility of reconstruction of state and input can be given by the Figure 9.

According to the motion (or trajectory) we can reconstruct states (q) and inputs (u) from Figure 9. According to the trajectory given in Figure (9a), we could draw the wheels in Figure (9b) because states (q) and inputs (u) can be reconstructed. In other words if we have movie we can reconstruct states and inputs.

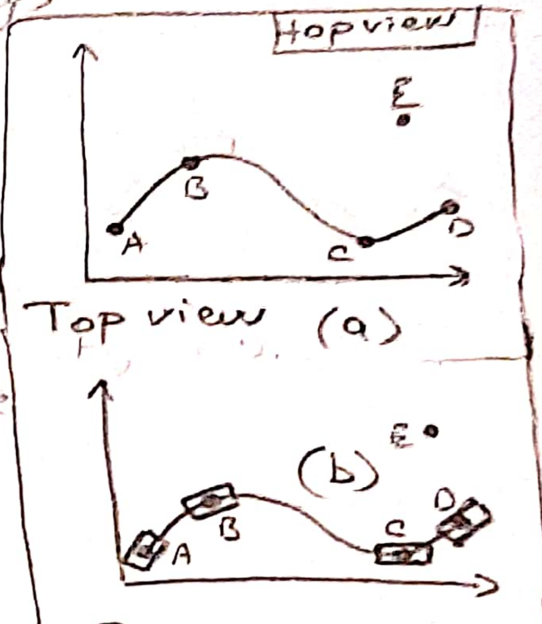


Figure 9

Additionally, assume that we are given a cartesian point such as E. Can we reconstruct states and inputs?

→ State reconstruction is lost:

If there is no motion we cannot derive θ because it can be found by the tangent to the motion. Thus, we can't tell what θ is which leads us to loose the state reconstruction.

→ Input reconstruction is lost:

We can't say there is no motion in that case. We know that $v=0$, because of (61). However, we can't define ω:

→ Because of (62)

→ Maybe it rotates around itself, but we can't see it unless it moves forward or backward.

To sum up, input reconstruction is lost, because, we can't define w.

MW