Car-following Models, Basic

CIVE.5490, UMass Lowell

Zhengbing He, Ph.D.
https://www.GoTrafficGo.com

March 28, 2024

Outline

1. What are car-following (CF) models?
2. Generic formulation of CF models
   - Continuous-time CF model
   - Discrete-time CF model
3. Numerical integration
   - Runge-Kutta scheme

Outline

1. What are car-following (CF) models?
2. Generic formulation of CF models
   - Continuous-time CF model
   - Discrete-time CF model
3. Numerical integration
   - Runge-Kutta scheme

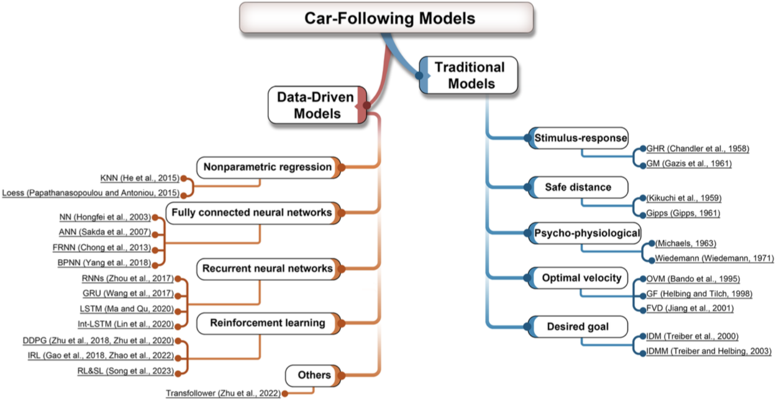
What are car-following models?

Mathematical and data-driven[1] models that describe how a vehicle moves on a road

- In a strict sense, car-following models describe the driver's behavior only in the presence of interactions with other vehicles while free traffic flow is described by a separate model.
- In a more general sense, car-following models include all traffic situations such as car-following situations, free traffic.
- See video - traffic simulation

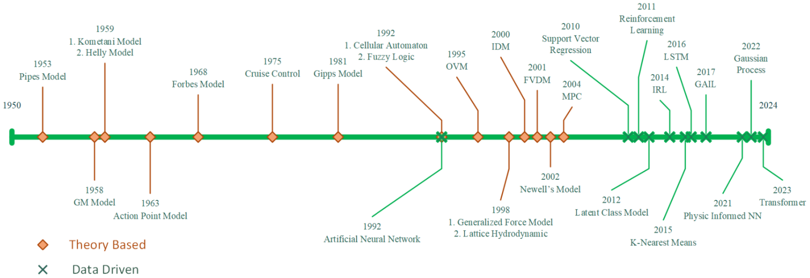
[1] Traditionally speaking, they are mathematical models, before the emergence of data-driven and machine learning methods

CF models in History



Xianda Chen, et al, FollowNet: A Comprehensive Benchmark for Car-Following Behavior Modeling, Scientific Data, 2023

CF models in History



Tianya Zhang, et al., Car-Following Models: A Multidisciplinary Review, arXiv:2304.07143v, 2024

Outline

1. What are car-following (CF) models?

2. Generic formulation of CF models
   - Continuous-time CF model
   - Discrete-time CF model

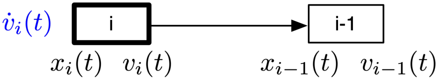
3. Numerical integration
   - Runge-Kutta scheme

Continuous-time CF model

Defined by an acceleration function A:

$$\boxed{\dot{v}_i(t) = A[\ x_i(t),\ x_{i-1}(t),\ v_i(t),\ v_{i-1}(t)\ ]} \qquad (1)$$

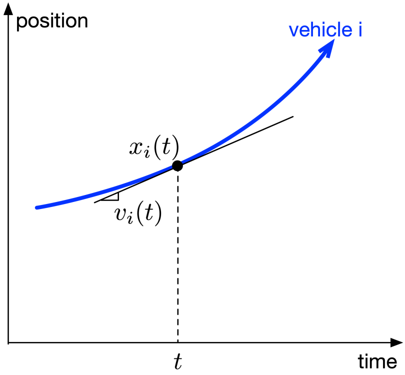
- vehicle $(i-1)$ is the leader of vehicle i;
- $x_i(t)$ and $v_i(t)$ are the position and speed of vehicle i at time t, respectively;
- **Input**: position and speed of vehicle i and $(i-1)$ at time t
- **Output**: changes in speed



Continuous-time CF model

Other inherent relationships:

$$\dot{x}_i(t) = v_i(t) \tag{2}$$

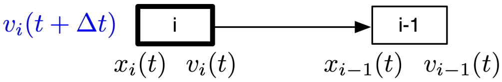


Discrete-time CF model

Defined by a speed function V:

$$\boxed{v_i(t + \Delta t) = V[\ x_i(t),\ x_{i-1}(t),\ v_i(t),\ v_{i-1}(t)\ ]} \qquad (3)$$

where Δt is the time step of updating the model.



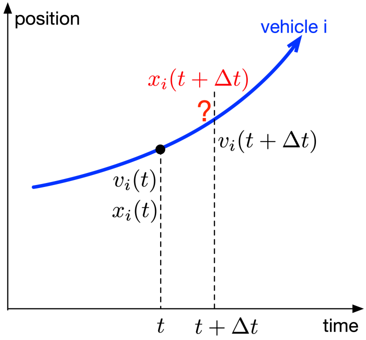
The input is the same as the continuous-time function, and the output is the speed after interval Δt, i.e., $v_i(t + \Delta t)$.

Outline

1. What are car-following (CF) models?
2. Generic formulation of CF models
   - Continuous-time CF model
   - Discrete-time CF model
3. Numerical integration
   - Runge-Kutta scheme

Runge-Kutta scheme

In practice (like a computer simulation), we cannot directly obtain the position, from the speed or acceleration given by a CF model, and an integration scheme is necessary for an approximate numerical solution.



Runge-Kutta scheme

To get the position of vehicle i at time t:

For the continuous-time function

- $\boxed{\dot{v}_i(t) = A(\cdot)} \xrightarrow{\textbf{RK}} \boxed{\dot{x}_i(t) = v_i(t+\Delta t)} \xrightarrow{\textbf{RK}} \boxed{x_i(t+\Delta t)}$

For the discrete-time function

- $\boxed{\dot{x}_i(t) = v_i(t+\Delta t) = V(\cdot)} \xrightarrow{\textbf{RK}} \boxed{x_i(t+\Delta t)}$

Runge-Kutta scheme

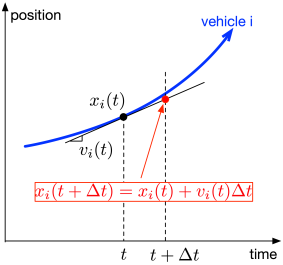
General knowledge:

- There is even six-order RK method.
- Higher order methods give us more accurate results,
- and higher computational burden, meanwhile
- In solving CF model, the first- and second-order methods are usually good enough,
- namely, Euler's method and Heun's method, respectively.

Euler's method: first-order RK method

Assume constant speed of $v_i(t)$ during $[t, t + \Delta t]$.

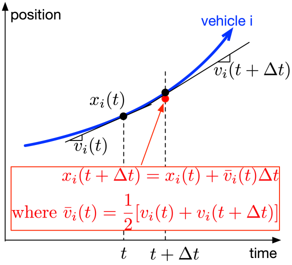
$$\boxed{x_i(t + \Delta t) = x_i(t) + v_i(t)\Delta t} \tag{4}$$



Heun's method: second-order RK method

Assume constant speed of $\frac{v_i(t)+v_i(t+\Delta t)}{2}$ during $[t, t+\Delta t]$.

$$\boxed{x_i(t+\Delta t) = x_i(t) + \frac{v_i(t)+v_i(t+\Delta t)}{2} \cdot \Delta t} \tag{5}$$



- 2nd-order: CF model TWICE: for $v_i(t)$ and $v_i(t+\Delta t)$
- 1st-order: only ONCE (for $v_i(t)$)

Thank you!