# 1 Geometry

**Table 1:** Some operations for vectors in 2D continuous space.

| Notation | Name | Return |
|---|---|---|
| $\mathbf{x}$ | | $(x, y) \in \mathbb{R}^2$ |
| $\Vert\mathbf{x}\Vert$ | `hypot`(x, y) | $d \in [0, \infty)$ |
| angle($\mathbf{x}$) | `arctan2`$(y, x)$ | $\varphi \in [-\pi, \pi]$ |
| $R(90°) \cdot \mathbf{x}$ | | $(-y, x)$ |
| $R(-90°) \cdot \mathbf{x}$ | | $(y, -x)$ |

# 2 Constants

| Symbol | Unit | Value | Explanation |
|---|---|---|---|
| $\Delta t$ | s | $0.01 - 0.001$ | Timestep |
| $\tau_{adj}$ | s | 0.5 | Characteristic time in which agent adjusts its movement. |
| $k$ | N | 1.5 | Social force scaling constant. |
| $\tau_0$ | s | 3.0 | Interaction time horizon. |
| $\mu$ | kg/s$^2$ | $1.2e + 05$ | Compression counteraction constant. |
| $\kappa$ | kg/(m s) | $2.4e + 05$ | Sliding friction constant. |
| $A$ | N | $2.0e + 03$ | Scaling coefficient for social force. |
| $B$ | m | 0.08 | Coefficient for social force. |
| $\Vert\mathbf{f}_{max}\Vert$ | N | | Force magnitude limit. |

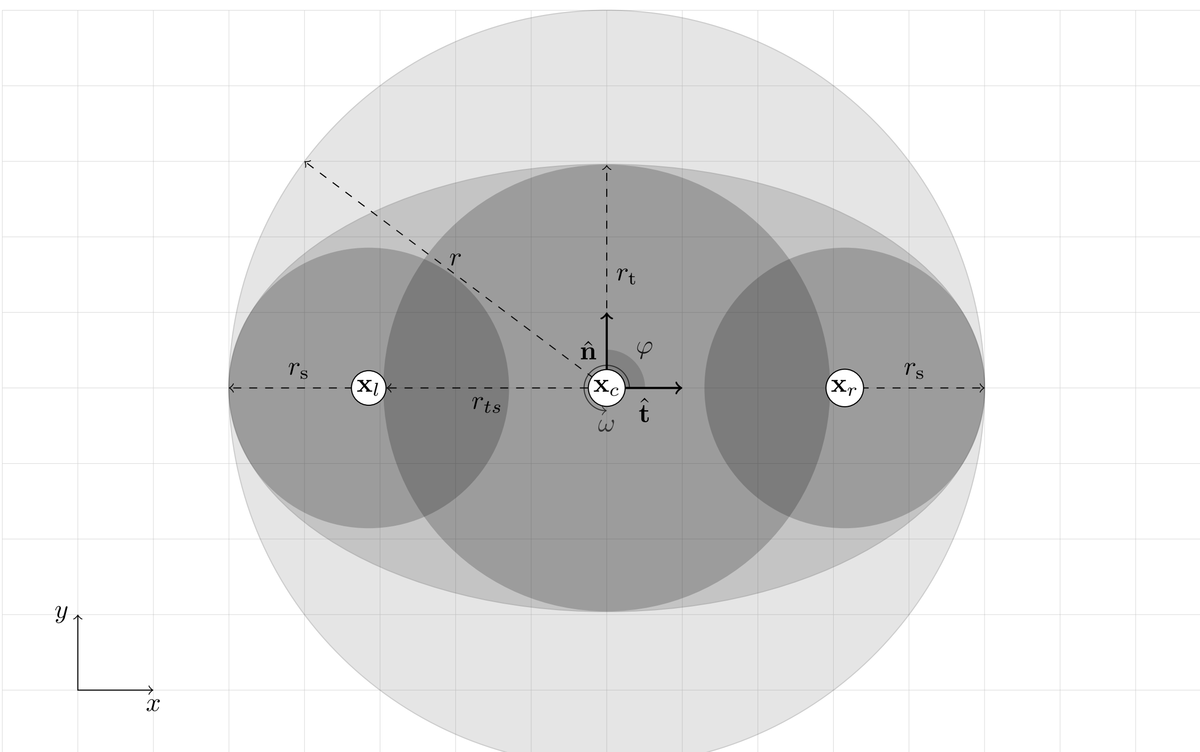


**Figure 1:** Circle, ellipse and three circle representations of an agent.

# 3 Agents

## 3.1 Properties

**Table 2:** Shoulder, torso and total radii.

| | Total | | Torso | Shoulder | |
|---|---|---|---|---|---|
| | $r$ | $\pm$ | $k_t = \frac{r_t}{r}$ | $k_s = \frac{r_s}{r}$ | $k_{ts} = \frac{r_{ts}}{r}$ |
| adult | 0.255 | 0.035 | 0.5882 | 0.3725 | 0.6275 |
| child | 0.210 | 0.015 | 0.5714 | 0.3333 | 0.6667 |
| eldery | 0.250 | 0.020 | 0.6000 | 0.3600 | 0.6400 |
| female | 0.240 | 0.020 | 0.5833 | 0.3750 | 0.6250 |
| male | 0.270 | 0.020 | 0.5926 | 0.3704 | 0.6296 |

**Table 3:** Properties

| | | | |
|---|---|---|---|
| $r$ | m | | Total radius |
| $r_t$ | m | | Torso radius |
| $r_s$ | m | | Shoulder radius |
| $r_{ts}$ | m | | Distance from torso to shoulder |
| $m$ | kg | 80 | Mass |
| $I$ | $\mathrm{kg} \cdot \mathrm{m}^2$ | 4.0 | Rotational moment |
| $\mathbf{x}$ | m | | Position |
| $\mathbf{v}$ | m/s | | Velocity |
| $v_0$ | m/s | | Goal velocity |
| $\hat{\mathbf{e}}_0$ | | | Goal direction |
| $\hat{\mathbf{e}}$ | | | Target direction |
| $\varphi$ | rad | $[-\pi, \pi]$ | Body angle |
| $\omega$ | rad/s | | Angular velocity |
| $\varphi_0$ | rad | $[-\pi, \pi]$ | Target angle |
| $\omega_0$ | rad/s | $0.4\pi$ | Max angular velocity |
| $p$ | | $0-1$ | Herding tendency |

## 3.2 Models

### 3.2.1 Circular

**Table 4:** Relative

| | |
|---|---|
| $\tilde{\mathbf{x}} = \mathbf{x}_i - \mathbf{x}_j$ | Relative position |
| $\tilde{\mathbf{v}} = \mathbf{v}_i - \mathbf{v}_j$ | Relative velocity |

| | |
|---|---|
| $d = \|\tilde{\mathbf{x}}\|$ | Distance |
| $\hat{\mathbf{n}} = \tilde{\mathbf{x}}/d$ | Normal vector |
| $\hat{\mathbf{t}} = R(-90°) \cdot \hat{\mathbf{n}}$ | Tangent vector |

Total radius and relative distance

$$\tilde{r} = r_i + r_j$$

$$h = d - \tilde{r}$$

### 3.2.2 Three circles

$$\begin{aligned} \mathbf{x}_r &= \mathbf{x}_c + \hat{\mathbf{t}} r_{ts} \\ \mathbf{x}_l &= \mathbf{x}_c - \hat{\mathbf{t}} r_{ts} \\ \hat{\mathbf{t}} &= \begin{bmatrix} -\sin(\varphi) & \cos(\varphi) \end{bmatrix} \end{aligned}$$

$$\mathbf{r}_{tot} = \begin{bmatrix} r_t & r_s & r_s \end{bmatrix}_i + \begin{bmatrix} r_t \\ r_s \\ r_s \end{bmatrix}_j$$

$$\begin{aligned} \mathbf{d} &= \left\| \begin{bmatrix} \mathbf{x}_c & \mathbf{x}_r & \mathbf{x}_l \end{bmatrix}_i - \begin{bmatrix} \mathbf{x}_c \\ \mathbf{x}_r \\ \mathbf{x}_l \end{bmatrix}_j \right\| \\ &= \left\| \begin{bmatrix} 0 & \hat{\mathbf{t}} r_{ts} & -\hat{\mathbf{t}} r_{ts} \end{bmatrix}_i - \begin{bmatrix} 0 \\ \hat{\mathbf{t}} r_{ts} \\ -\hat{\mathbf{t}} r_{ts} \end{bmatrix}_j + (\mathbf{x}_i - \mathbf{x}_j) \right\| \\ &= \left\| \begin{bmatrix} 0 & 1 & -1 \end{bmatrix} \left(\hat{\mathbf{t}} r_{ts}\right)_i - \begin{bmatrix} 0 \\ 1 \\ -1 \end{bmatrix} \left(\hat{\mathbf{t}} r_{ts}\right)_j + \tilde{\mathbf{x}} \right\| \\ &= \left\| \mathbf{k} \left(\hat{\mathbf{t}} r_{ts}\right)_i - \mathbf{k}^T \left(\hat{\mathbf{t}} r_{ts}\right)_j + \tilde{\mathbf{x}} \right\| \\ &= \left\| \mathbf{c}_i - \mathbf{c}_j^T + \tilde{\mathbf{x}} \right\| \end{aligned}$$

$$\mathbf{h} = \mathbf{d} - \mathbf{r}_{tot}$$

1. Find

   $$h = \min(\mathbf{h})$$

   and track minimizing values

   $$\hat{\mathbf{e}}_{ij}, k_i, k_j, r_i, r_j$$

2. 

   $$\begin{aligned} \mathbf{r}_i^{moment} &= \mathbf{x}_i^c + k_i \cdot \hat{\mathbf{t}}_i r_i^{ts} + r_i \hat{\mathbf{e}}_{ij} \\ \mathbf{r}_j^{moment} &= \mathbf{x}_j^c + k_j \cdot \hat{\mathbf{t}}_j r_j^{ts} - r_j \hat{\mathbf{e}}_{ij} \end{aligned}$$

3. Return $(\tilde{\mathbf{x}}, r_{tot}, h, \mathbf{r}_i^{moment}, \mathbf{r}_j^{moment})$

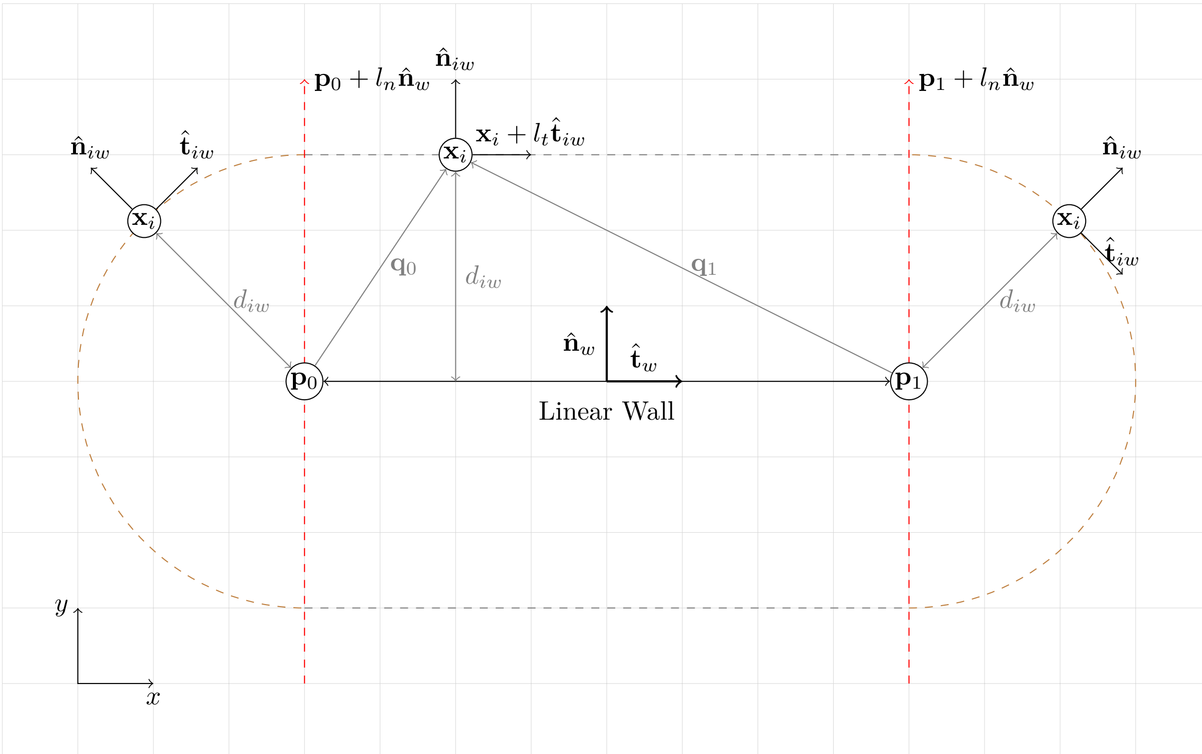


**Figure 2:** Absolute distance from a linear wall.

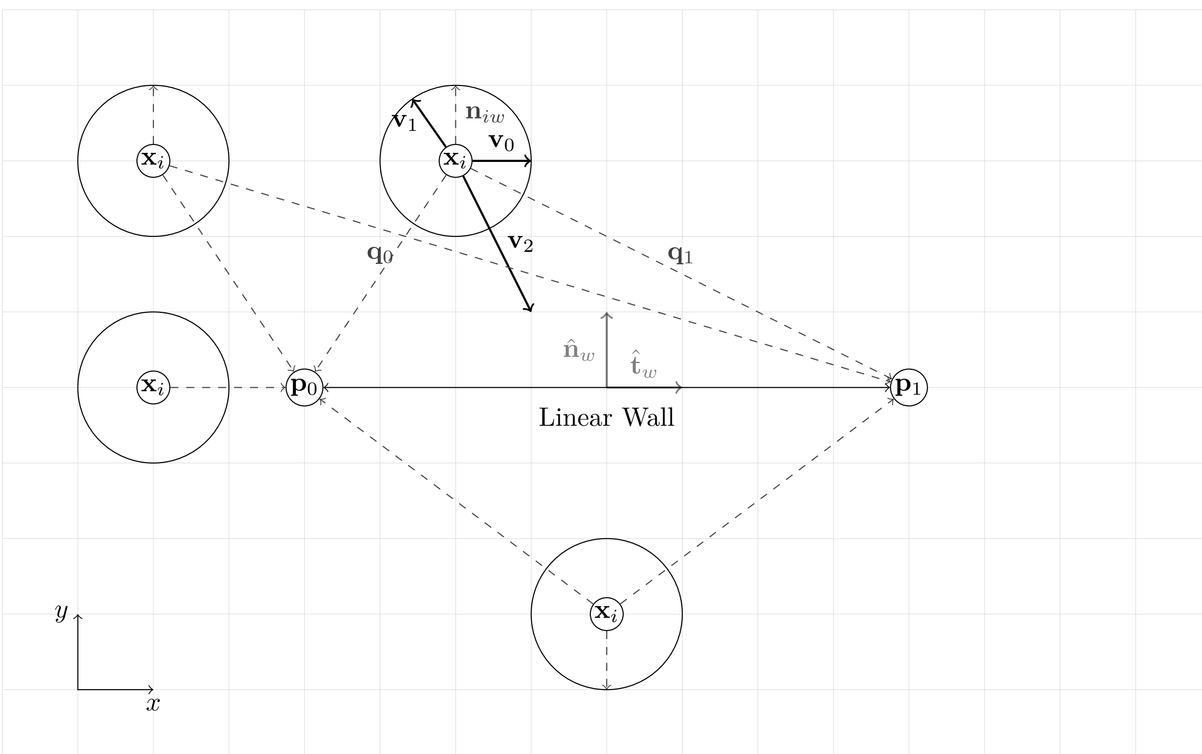


**Figure 3:** Velocity dependent distance from a linear wall.

# 4 Linear wall

## 4.1 Properties

| | |
|---|---|
| $\mathbf{p}_0$ | Start point |
| $\mathbf{p}_1$ | End point |
| $h_{iw} = d_{iw} - r_i$ | |
| $l_w = \|\mathbf{p}_1 - \mathbf{p}_0\|$ | Length |
| $\hat{\mathbf{t}}_w = (\mathbf{p}_1 - \mathbf{p}_0)/l_w$ | |
| $\hat{\mathbf{n}}_w = R(90°) \cdot \hat{\mathbf{t}}_w$ | |

## 4.2 Absolute distance

Solving linear system of equations determining the position of the agent $\mathbf{x}_i$ in relation to wall

$$\begin{cases} \mathbf{p}_0 + l_{n_0}\hat{\mathbf{n}}_w = \mathbf{x}_i + l_{t_0}\hat{\mathbf{t}}_w \\ \mathbf{p}_1 + l_{n_1}\hat{\mathbf{n}}_w = \mathbf{x}_i + l_{t_1}\hat{\mathbf{t}}_w \end{cases}$$

$$\begin{cases} l_{n_0}\hat{\mathbf{n}}_w - l_{t_0}\hat{\mathbf{t}}_w = \mathbf{x}_i - \mathbf{p}_0 = \mathbf{q}_0 \\ l_{n_1}\hat{\mathbf{n}}_w - l_{t_1}\hat{\mathbf{t}}_w = \mathbf{x}_i - \mathbf{p}_1 = \mathbf{q}_1 \end{cases}$$

In matrix form

$$\begin{bmatrix} l_{n_0} & l_{n_1} \\ l_{t_0} & l_{t_1} \end{bmatrix} = \mathbf{A}^{-1} \begin{bmatrix} \mathbf{q}_0 & \mathbf{q}_1 \end{bmatrix}$$

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} \hat{\mathbf{n}}_w & -\hat{\mathbf{t}}_w \end{bmatrix} = \begin{bmatrix} -t_1 & -t_0 \\ t_0 & -t_1 \end{bmatrix} \\ \mathbf{A}^{-1} &= \frac{1}{t_0^2 + t_1^2} \begin{bmatrix} -t_1 & t_0 \\ -t_0 & -t_1 \end{bmatrix} = \begin{bmatrix} -t_1 & t_0 \\ -t_0 & -t_1 \end{bmatrix} \\ &= \begin{bmatrix} \hat{\mathbf{n}}_w \\ -\hat{\mathbf{t}}_w \end{bmatrix} = \mathbf{A}^T \end{aligned}$$

Conditions

$$\begin{aligned} l_n &= l_{n_0} \vee l_{n_1} = \hat{\mathbf{n}}_w \cdot \mathbf{q}_0 \vee \hat{\mathbf{n}}_w \cdot \mathbf{q}_1 \\ l_t &= l_{t_1} + l_{t_0} = -\hat{\mathbf{t}}_w \cdot \mathbf{q}_1 - \hat{\mathbf{t}}_w \cdot \mathbf{q}_0 \end{aligned}$$

Distance between agent and linear wall

$$d_{iw} = \begin{cases} \|\mathbf{q}_0\| & l_t > l_w \\ |l_n| & \text{otherwise} \\ \|\mathbf{q}_1\| & l_t < -l_w \end{cases}$$

Normal vector away from the wall

$$\hat{\mathbf{n}}_{iw} = \begin{cases} \hat{\mathbf{q}}_0 & l_t > l_w \\ \operatorname{sign}(l_n)\hat{\mathbf{n}}_w & \text{otherwise} \\ \hat{\mathbf{q}}_1 & l_t < -l_w \end{cases}$$

## 4.3 Velocity relative distance

| | |
|---|---|
| $\tilde{\mathbf{x}} = \mathbf{x}_{iw}$ | Relative position |
| $\tilde{\mathbf{v}} = \mathbf{v}_{iw} = \mathbf{v}_i$ | Relative velocity |
| $\tilde{r} = r_{iw}$ | Total radius |
| $d = \|\tilde{\mathbf{x}}\|$ | Distance |
| $h = d - \tilde{r}$ | Relative distance |

$$\begin{aligned} \mathbf{q}_0 &= \mathbf{p}_0 - \mathbf{x} \\ \mathbf{q}_1 &= \mathbf{p}_1 - \mathbf{x} \\ \hat{\mathbf{n}}_{iw} &= -\operatorname{sign}(\hat{\mathbf{n}}_w \cdot \mathbf{q}_0)\hat{\mathbf{n}}_w \end{aligned}$$

$$\begin{aligned} \boldsymbol{\alpha} &= [\operatorname{angle}(\mathbf{q}_0), \operatorname{angle}(\mathbf{q}_1), \operatorname{angle}(\hat{\mathbf{n}}_{iw})] \\ \varphi &= \operatorname{angle}(\mathbf{v}) \\ \boldsymbol{\alpha}_2 &= \boldsymbol{\alpha} - \varphi \mod 2\pi \end{aligned}$$

$$i = (\arg\min(\boldsymbol{\alpha}_2), \arg\max(\boldsymbol{\alpha}_2))$$

Intersection

$$\begin{aligned} \mathbf{x} + a\mathbf{v} &= \mathbf{p}_0 + b(\mathbf{p}_1 - \mathbf{p}_0), \quad a \in \mathbb{R}^+, \quad b \in [0, 1] \\ [\mathbf{v}, -\mathbf{p}] \cdot [a, b] &= \mathbf{q}_0, \quad \mathbf{p} = \mathbf{p}_1 - \mathbf{p}_0 \end{aligned}$$

# 5 Motion

## 5.1 Social force

Total force exerted on the agent is the sum of movement adjusting, social and contact forces between other agents and wall.

$$\mathbf{f}_i(t) = \mathbf{f}_i^{adj} + \sum_{j \neq i} \left( \mathbf{f}_{ij}^{soc} + \mathbf{f}_{ij}^{c} \right) + \sum_{w} \left( \mathbf{f}_{iw}^{soc} + \mathbf{f}_{iw}^{c} \right) + \boldsymbol{\xi}_i$$

### 5.1.1 Adjusting force

Force adjusting agent's movement towards desired in some characteristic time

$$\mathbf{f}^{adj} = \frac{m}{\tau^{adj}} (v_0 \cdot \hat{\mathbf{e}} - \mathbf{v})$$

### 5.1.2 Social force

Psychological force for collision avoidance. Naive velocity independent equation

$$\mathbf{f}^{soc} = A \exp\left(-\frac{h}{B}\right) \hat{\mathbf{n}}$$

Improved velocity dependent algorithm

$$\begin{aligned}
\mathbf{f}^{soc} &= -\nabla_{\tilde{\mathbf{x}}} E(\tau) \\
&= -\nabla_{\tilde{\mathbf{x}}} \left( \frac{k}{\tau^2} \exp\left(-\frac{\tau}{\tau_0}\right) \right) \\
&= -\left(\frac{k}{a\tau^2}\right) \left(\frac{2}{\tau} + \frac{1}{\tau_0}\right) \exp\left(-\frac{\tau}{\tau_0}\right) \left( \tilde{\mathbf{v}} - \frac{a\tilde{\mathbf{x}} + b\tilde{\mathbf{v}}}{d} \right),
\end{aligned}$$

where

$$\begin{aligned}
a &= \tilde{\mathbf{v}} \cdot \tilde{\mathbf{v}} \\
b &= -\tilde{\mathbf{x}} \cdot \tilde{\mathbf{v}} \\
c &= \tilde{\mathbf{x}} \cdot \tilde{\mathbf{x}} - \tilde{r}^2 \\
d &= \sqrt{b^2 - ac}, \quad b^2 - ac > 0 \\
\tau &= \frac{b - d}{a} > 0.
\end{aligned}$$

### 5.1.3 Contact force

Physical contact force

$$\mathbf{f}^{c} = -h \cdot \left( \mu \cdot \hat{\mathbf{n}} - \kappa \cdot (\mathbf{v} \cdot \hat{\mathbf{t}}) \hat{\mathbf{t}} \right), \quad h < 0$$

### 5.1.4 Random Fluctuation

Uniformly distributed random fluctuation force

$$\boldsymbol{\xi} = f \cdot \begin{bmatrix} \cos(\varphi) & \sin(\varphi) \end{bmatrix},$$

where

$$f \in [0, f_{max}], \quad \varphi \in [0, 2\pi]$$

## 5.2 Rotational

Total torque exerted on agent, is the sum of adjusting contact and social torques

$$M_i(t) = M_i^{adj} + \sum_{j \neq i} \left( M_{ij}^{soc} + M_{ij}^{c} \right) + \sum_{w} \left( M_{iw}^{soc} + M_{iw}^{c} \right) + \eta_i(t)$$

### 5.2.1 Adjusting torque

Torque adjusting agent's rotational motion towards desired

$$M^{adj} = \frac{I}{\tau} \left( (\varphi(t) - \varphi^0) \omega^0 - \omega(t) \right)$$

### 5.2.2 Social torque

Torque from social forces acting with other agent or wall

$$\mathbf{M}^{soc} = \mathbf{r}^{soc} \times \mathbf{f}^{soc}$$

### 5.2.3 Contact torque

Torque from contact forces acting with other agent or wall

$$\mathbf{M}^{c} = \mathbf{r}^{c} \times \mathbf{f}^{c}$$

### 5.2.4 Related equations

Torque calculated using cross product

$$\mathbf{M} = \mathbf{r} \times \mathbf{f} = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ R_1 & R_2 & R_3 \\ f_1 & f_2 & f_3 \end{vmatrix}$$

which in two dimensions is

$$M = \begin{vmatrix} R_1 & R_2 \\ f_1 & f_2 \end{vmatrix} = R_1 \cdot f_2 - R_2 \cdot f_1$$

# 6 Integrators

## 6.1 Differential systems

Position and velocity

$$m\frac{d^2}{dt^2}\mathbf{x}(t) = \mathbf{f}(t)$$

Rotational motion

$$I\frac{d^2}{dt^2}\varphi(t) = M(t)$$

## 6.2 Excplicit Euler Method

Updating using discrete time step $\Delta t$

$$\begin{aligned} t_0 &= 0 \\ t_1 &= t_0 + \Delta t \\ &\vdots \\ t_k &= t_{k-1} + \Delta t \end{aligned}$$

Acceleration on an agent

$$\begin{aligned} a_k &= \mathbf{f}_k / m \\ \mathbf{v}_{k+1} &= \mathbf{v}_k + a_k \Delta t \\ \mathbf{x}_{k+1} &= \mathbf{x}_k + \mathbf{v}_{k+1} \Delta t \end{aligned}$$

Angular acceleration

$$\begin{aligned} \alpha_k &= M_k / I \\ \omega_{k+1} &= \omega_k + \alpha_k \Delta t \\ \varphi_{k+1} &= \varphi_k + \omega_{k+1} \Delta t \end{aligned}$$

## 6.3 Velocity verlet

Velocity verlet algorithm

$$\begin{aligned} \mathbf{v}_{k+\frac{1}{2}} &= \mathbf{v}_k + \frac{1}{2} a_k \Delta t \\ \mathbf{x}_{k+1} &= \mathbf{x}_k + \mathbf{v}_{k+\frac{1}{2}} \Delta t \\ \mathbf{v}_{k+1} &= \mathbf{v}_{k+\frac{1}{2}} + \frac{1}{2} a_{k+1} \Delta t \end{aligned}$$

or more simply

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{x}_k + \mathbf{v}_k \Delta t + \frac{1}{2} a_k \Delta t^2 \\ \mathbf{v}_{k+1} &= \mathbf{v}_k + \frac{1}{2}\left(a_k + a_{k+1}\right) \Delta t \end{aligned}$$

# 7 Navigation

## 7.1 Theory

Navigation algorithm is a function that takes at least coordinate $\mathbf{x}$ as an argument and returns an unit vector $\hat{\mathbf{e}}$ that is used as target direction for the agent

$$f(\mathbf{x}, \ldots) \to \hat{\mathbf{e}}$$

## 7.2 Manual construction

## 7.3 Fluid flow

One way to find suitable function is to solve how *incompressible*, *irrotational* and *inviscid* fluid (ideal fluid) would flow out of the constructed space.

`https://en.wikipedia.org/wiki/Conservative_vector_field#Irrotational_flows`

`https://en.wikipedia.org/wiki/Inviscid_flow`

`https://en.wikipedia.org/wiki/Euler_equations_(fluid_dynamics)`

## 7.4 Combination

# 8 Spatial game

Spatial game for egress congestion.

## 8.1 Game matrix

| | |
|---|---|
| $T_i = \lambda_i / \beta$ | Estimated evacuation time |
| $\lambda_i$ | Number of other agents closer to the exit |
| $\beta$ | Capacity of the exit |
| $T_{ij} = (T_i + T_j)/2$ | Average evacuation time |
| $T_{ASET}$ | Available safe egress time |
| $T_0 (= T_{ASET})$ | Time difference between $T_{ASET}$ and $T_i$ before agents start playing the game |
| $C > 0$ | Cost of conflict |

Number of other agents closer to the exit can be solved

$$\lambda = \operatorname{argsort} \|\mathbf{p}_0 - \mathbf{x}\|$$

where $\mathbf{p}_0$ is the center of the exit

Cost function

$$u(T_i, T_{ASET}), \quad u'(T_i) \geq 0, \quad u''(T_i) \geq 0$$

Increase/decrease in cost

$$\Delta u(T_{ij}) = u(T_{ij}) - u(T_{ij} - \Delta T) \approxeq u'(T_{ij}) \Delta T$$

| | Impatient | Patient |
|---|---|---|
| Impatient | $C, C$ | $-\Delta u(T_{ij}), \Delta u(T_{ij})$ |
| Patient | $\Delta u(T_{ij}), -\Delta u(T_{ij})$ | $0, 0$ |

| | Impatient | Patient |
|---|---|---|
| Impatient | $\frac{C}{\Delta u(T_{ij})}, \frac{C}{\Delta u(T_{ij})}$ | $-1, 1$ |
| Patient | $1, -1$ | $0, 0$ |

i) Prisoner's dilemma (PD)

$$0 < \frac{C}{\Delta u(T_{ij})} \leq 1$$

ii) Hawk-dove (HD)

$$\frac{C}{\Delta u(T_{ij})} > 1$$

Assumptions

1. Game is not played
$$T_{ij} \leq T_{ASET} - T_0$$
2. Cost function starts to increase quadratically
$$T_{ij} > T_{ASET} - T_0$$
3. Game turns into prisoner's dilemma
$$u'(T_{ASET}) = C, \quad T_{ij} \geq T_{ASET}$$

Cost function that meets the assumptions

$$u(T_{ij}) = \begin{cases} 0 & T_{ij} \leq T_{ASET} - T_0 \\ \frac{C}{2T_0} \left(T_{ij} - T_{ASET} + T_0\right)^2 & T_{ij} > T_{ASET} - T_0 \end{cases}$$

Derivative

$$u'(T_{ij}) = \begin{cases} 0 & T_{ij} \leq T_{ASET} - T_0 \\ \frac{C}{T_0} \left(T_{ij} - T_{ASET} + T_0\right) & T_{ij} > T_{ASET} - T_0 \end{cases}$$

Loss/gain of overtaking

$$\Delta u(T_{ij}) \approxeq u'(T_{ij}) \Delta T = \frac{C}{T_0} \left(T_i - T_{ASET} + T_0\right) \Delta T$$

Value parameter of the game matrix

$$\frac{C}{\Delta u(T_{ij})} \approxeq \frac{T_0}{T_{ij} - T_{ASET} + T_0}$$

## 8.2 Settings and best-response dynamics

| | Unit | Value | |
|---|---|---|---|
| | | 4 | von Neumann neighborhood |
| | | 8 | Moore neighborhood |
| $r_n$ | m | 0.40 | Distance to agent that is considered as neighbor |
| $v_i$ | | | Loss defined by game matrix |
| $S$ | | | Set of strategies $\{\text{Patient}, \text{Impatient}\}$ |
| $s$ | | | Strategy $\in \{\text{Patient}, \text{Impatient}\}$ |

The best-response strategy

$$s_i^{(t)} = \arg \min_{s_i' \in S} \sum_{j \in N_i} v_i \left(s_i', s_j^{(t-1)}; T_{ij}\right)$$

$s_j^{(t-1)}$ strategy neighbor played on period $t-1$

Updating strategy using poisson process.

# 9 Algorithms

## 9.1 Interactions

### 9.1.1 Circular model

**Algorithm 1** Interaction between circular agents.

**Require:** $i, j \in N, \quad i \neq j$
**Ensure:**
$\tilde{\mathbf{x}} \leftarrow \mathbf{x}_i - \mathbf{x}_j$
$r_{tot} \leftarrow r_i + r_j$
$d \leftarrow \|\tilde{\mathbf{x}}\|$
$h \leftarrow d - r_{tot}$

**if** $h \leq sight$ **then**
  $\tilde{\mathbf{v}} \leftarrow \mathbf{v}_i - \mathbf{v}_j$
  $\mathbf{f} \leftarrow \mathbf{f}_{soc}(\tilde{\mathbf{x}}, \tilde{\mathbf{v}}, r_{tot}, k, \tau_0)$

  **if** $h < 0$ **then**
    $\hat{\mathbf{n}} \leftarrow \tilde{\mathbf{x}}/d$
    $\hat{\mathbf{t}} \leftarrow R(-90°)\hat{\mathbf{n}}$
    $\mathbf{f} \leftarrow +\mathbf{f}_c(\tilde{\mathbf{v}}, h, \hat{\mathbf{n}}, \hat{\mathbf{t}}, \mu, \kappa)$
  **end if**

  $\mathbf{f}_i \leftarrow +\mathbf{f}$
  $\mathbf{f}_j \leftarrow -\mathbf{f}$
**end if**

### 9.1.2 Three circles model

**Algorithm 2** Distance between agent using three circles model.

**Require:** $i, j \in N, \quad i \neq j$
**Ensure:**
**for** $\mathbf{x}_i, r_i \leftarrow (\mathbf{x}_c, \mathbf{x}_l, \mathbf{x}_r)_i, (r_t, r_s, r_s)_i$ **do**
  **for** $\mathbf{x}_j, r_j \leftarrow (\mathbf{x}_c, \mathbf{x}_l, \mathbf{x}_r)_j, (r_t, r_s, r_s)_j$ **do**
    $\tilde{\mathbf{x}} \leftarrow \mathbf{x}_i - \mathbf{x}_j$
    $r_{tot} \leftarrow r_i + r_j$
    $d \leftarrow \|\tilde{\mathbf{x}}\|$
    $h \leftarrow d - r_{tot}$
    **if** $h < h_{min}$ **then**
      $\hat{\mathbf{n}} \leftarrow \tilde{\mathbf{x}}/d$
      $\mathbf{x} \leftarrow \mathbf{x}_i, \mathbf{x}_j$
      $r \leftarrow r_i, r_j$
    **end if**
  **end for**
**end for**
$\mathbf{r}_i^{moment} = (\mathbf{x} + r \cdot \hat{\mathbf{n}} - \mathbf{x}_c)_i$
$\mathbf{r}_i^{moment} = (\mathbf{x} - r \cdot \hat{\mathbf{n}} - \mathbf{x}_c)_j$
**return** $\hat{\mathbf{n}}, h, \mathbf{r}_i^{moment}, \mathbf{r}_j^{moment}$

**Algorithm 3** Interaction between agents using three circles model.

**Require:** $i, j \in N, \quad i \neq j$
**Ensure:**
$\tilde{\mathbf{x}} \leftarrow \mathbf{x}_i - \mathbf{x}_j$
$r_{tot} \leftarrow r_i + r_j$
$d \leftarrow \|\tilde{\mathbf{x}}\|$
$h \leftarrow d - r_{tot}$

**if** $h \leq sight$ **then**
  $\tilde{\mathbf{v}} \leftarrow \mathbf{v}_i - \mathbf{v}_j$
  $\mathbf{f} \leftarrow \mathbf{f}_{soc}(\tilde{\mathbf{x}}, \tilde{\mathbf{v}}, r_{tot}, k, \tau_0)$
  **if** $h \leq cutoff$ **then**
    $\hat{\mathbf{n}}, h, r_{i,j}^{moment} \leftarrow \mathrm{distance}(\mathrm{agent}, i, j)$
    **if** $h < 0$ **then**
      $\hat{\mathbf{t}} \leftarrow R(-90°)\hat{\mathbf{n}}$
      $\mathbf{f} \leftarrow +\mathbf{f}_c(\tilde{\mathbf{v}}, h, \hat{\mathbf{n}}, \hat{\mathbf{t}}, \mu, \kappa)$
    **end if**
    $\mathbf{f}_i \leftarrow +\mathbf{f}$
    $\mathbf{f}_j \leftarrow -\mathbf{f}$
    $M_i \leftarrow +M_i^c(\mathbf{r}_i^{moment}, \mathbf{f})$
    $M_j \leftarrow -M_j^c(\mathbf{r}_j^{moment}, \mathbf{f})$
  **end if**
**end if**