

# 群组动画

## Crowd and Group Animation

金小刚

Email: [jin@cad.zju.edu.cn](mailto:jin@cad.zju.edu.cn)

浙江大学CAD&CG国家重点实验室

紫金港校区蒙民伟楼512

# 基于社会力模型的群体行为模拟



- **参考文献**

**D. Helbing, I. Farkas, T. Vicsek: Simulating dynamical features of escape panic, *Nature* 407: 487 (2000) (Google citations: 6403次, 2024.10.14)**

# Stampedes (踩踏事件)

- Videos and descriptions of disasters

**This is a partial list of human stampedes. The following are some well-known disasters:**

24.07.2010, Love Parade, Duisburg (Germany), [Wikipedia](#), [YouTube](#)

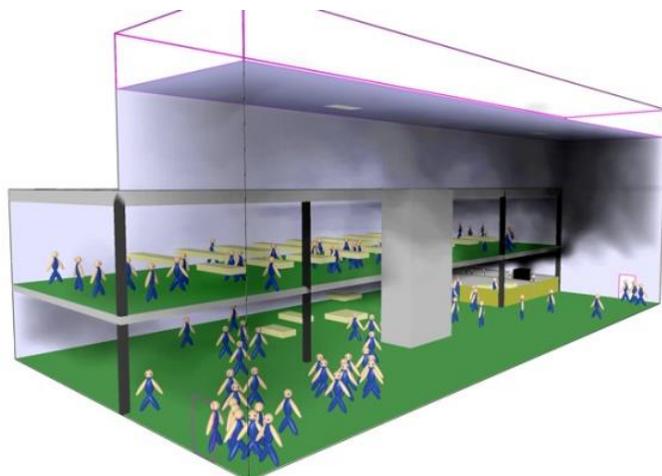
15.04.1989, Hillsborough, Sheffield (UK), [Wikipedia](#), [YouTube](#)

29.05.1985, Heysel stadium, Brussels (Belgium), [Wikipedia](#), [YouTube](#)

02.01.1971, Ibrox (UK), [Wikipedia](#), [YouTube](#)

- **Website:** <http://angel.elte.hu/panic>

# 类似场景



# Demo



# Everything we know so far as dozens killed in crush at religious festival (2021)



**Warning**

Contains video some  
may find distressing

# 问题提出

- 人群的**羊群效应**可以是致命的;
- 当处于恐慌状态时, 人的行为动作会变得笨拙和危险;
- 获取真实的人群恐慌行为数据是相当困难的;
- 公共场合中的人群聚集通常是事故发生的隐患, 人群拥挤度越高, 事故风险也就越大, 而且造成的人员伤亡和损失也更为严重;
- 建立一个有效的行人群体行为仿真模型对公共安全, 降低事故风险有着重要的意义。

# 解决方法

- Helbing等提出了**社会力模型**的概念，并对人群运动进行了仿真，展现了人群疏散过程中的堵塞现象。
- 把人抽象为自我驱动的质点；
- 把人群运动中的物理和社会心理学影响用社会力来进行建模；
- 模拟人群的恐慌现象并看看会发生什么。

# 基本原理

- 社会力模型以牛顿动力学为基础，由各个力的表达式来体现行人不同的动机和影响。在**社会力模型**中，由于对影响个体的因素考虑得比较全面，对个体行为的建模比较合理，该模型可以逼真地模拟人群的疏散过程。
- 个体的实际行为受**主观意识**、**其他个体**及**障碍物**三方面因素的影响，均可等效为**力**在个体上的作用。

# 仿真人群的牛顿动力学模型

$$m_i \frac{d\mathbf{v}_i}{dt} = m_i \underbrace{\frac{v_i^0(t)\mathbf{e}_i^0(t) - \mathbf{v}_i(t)}{\tau_i}}_{\text{自身驱动力}} + \sum_{j(\neq i)} \mathbf{f}_{ij} + \sum_w \mathbf{f}_{iw}$$

其他个体对*i* 边界力产生的力

$v_i(t)$ =行人*i*的实际速度

$r_i(t)$ =行人*i*的位置

# 1. 个体*i*的自身驱动力

- 主观意识对个体行为的影响可等效为个体受自身施加的“**社会力**”，体现了行人以期望的速度移动到目的地的动机：

$$\mathbf{f}_i^0(t) = m_i \frac{v_i^0(t)\mathbf{e}_i^0(t) - \mathbf{v}_i(t)}{\tau_i}$$

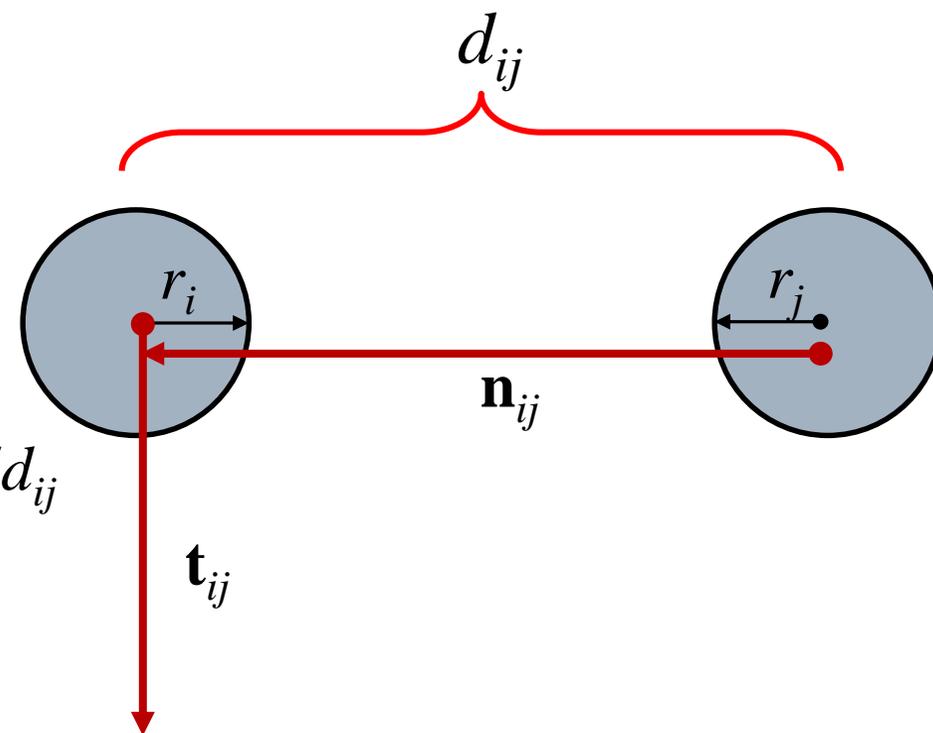
$m_i$  为行人质量， $v_i^0(t)$  为行人的期望速度， $\mathbf{e}_i^0(t)$  为行人的期望运动方向， $\tau_i$  为特征时间。

- 期望的运动方向可由行人此刻的位置 $\mathbf{r}_i$ 和目标位置 $\mathbf{p}$ 算出，即  $\mathbf{e}_i^0(t) = \frac{\mathbf{p} - \mathbf{r}_i}{\|\mathbf{p} - \mathbf{r}_i\|}$

## 2. 个体j对个体i产生的力

$$\mathbf{f}_{ij} = \left\{ A_i \exp \left[ \left( r_{ij} - d_{ij} \right) / B_i \right] + kg \left( r_{ij} - d_{ij} \right) \right\} \mathbf{n}_{ij} + \kappa g \left( r_{ij} - d_{ij} \right) \Delta v_{ji}^t \mathbf{t}_{ij}$$

- $A_i \exp \left[ \left( r_{ij} - d_{ij} \right) / B_i \right] \mathbf{n}_{ij}$  为**心理学力**
- $r_{ij}$ 为个体*i*和个体*j*的半径之和
- $d_{ij}$ 为个体*i*和个体*j*中心之间的距离
- $\mathbf{n}_{ij}$ 为个体*j*到个体*i*的单位矢量 $\mathbf{n}_{ij} = (n_{ij}^1, n_{ij}^2) = (\mathbf{r}_i - \mathbf{r}_j) / d_{ij}$
- $A_i$ 和 $B_i$ 为常数,  $A_i$ 为相互作用的强度



## 2. 个体 $j$ 对个体 $i$ 产生的力

- $kg(r_{ij} - d_{ij})\mathbf{n}_{ij}$  个体 $j$ 对个体 $i$ 施加的**身体力**，是行人为了避免身体接触避让他人所产生的力( $k$ 为一很大的常数):

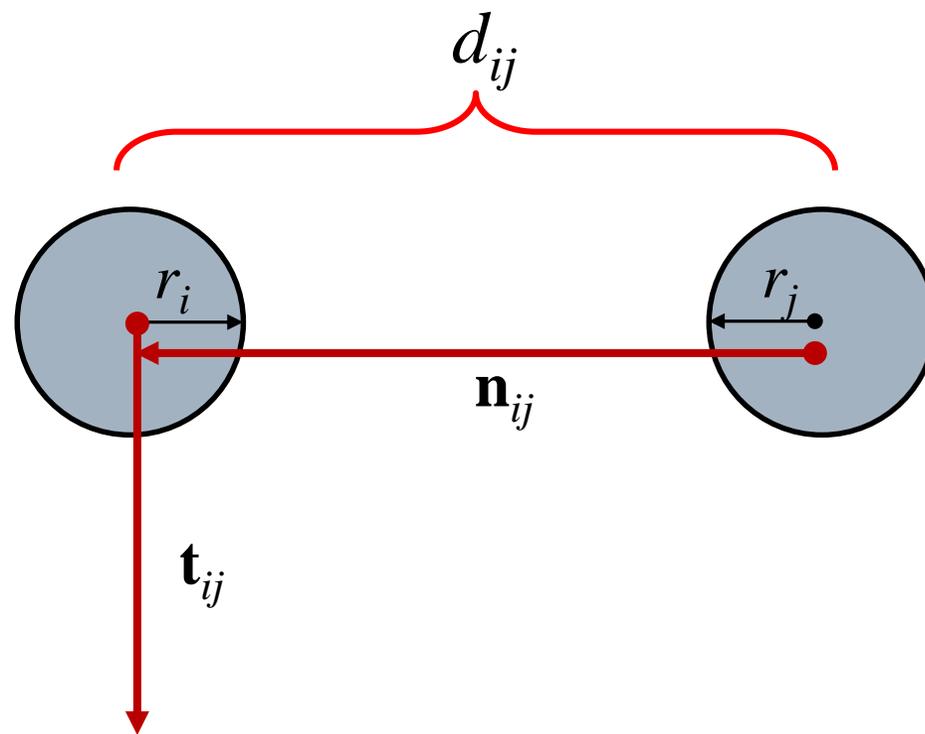
$$g(x) = \begin{cases} x; & x > 0 \\ 0; & x \leq 0 \end{cases}$$

- 当 $x > 0$ 时，表示两个行人之间的半径之和大于两者之间的距离，两人之间**有身体接触**;
- 当 $x \leq 0$ 时，两个行人之间**没有身体接触**，两人之间的身体力为0;

## 2. 个体*j*对个体*i*产生的力

$\kappa g (r_{ij} - d_{ij}) \Delta v_{ji}^t \mathbf{t}_{ij}$  为防止人滑倒的**滑动摩擦力**

- $\mathbf{t}_{ij} = (-\mathbf{n}_{ij}^2, \mathbf{n}_{ij}^1)$ 为切向方向
- $\Delta v_{ji}^t$  为切向方向的速度差
- $\kappa$ 为一个很大的常数



# 3. 环境力

- 边界和障碍物对人的影响类似于人与人之间的作用。边界 $w$ 对个体 $i$ 的影响为：

$$\mathbf{f}_{iw} = \left\{ A_i \exp \left[ (r_i - d_{iw}) / B_i \right] + kg (r_i - d_{iw}) \right\} \mathbf{n}_{iw} - \kappa g (r_i - d_{iw}) (\mathbf{v}_i \bullet \mathbf{t}_{iw}) \mathbf{t}_{iw}$$

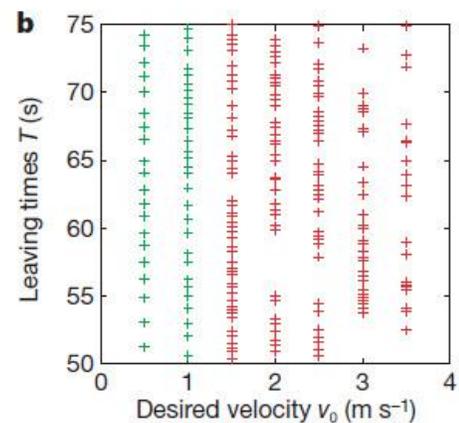
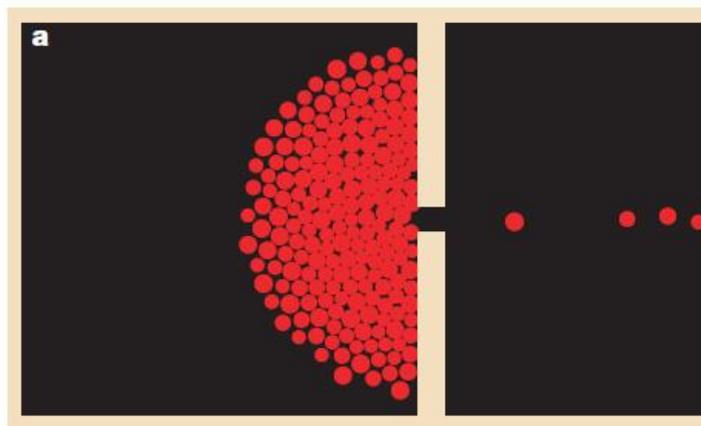
- $d_{iw}$  为个体 $i$ 到环境 $w$ 之间的距离
- $\mathbf{t}_{iw}$  切矢量

# 常量和参数的选择

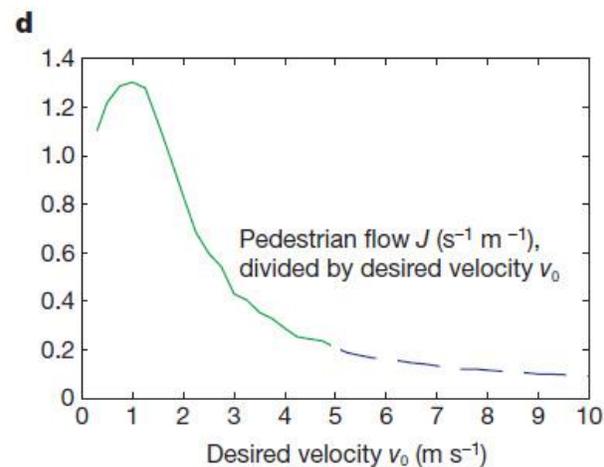
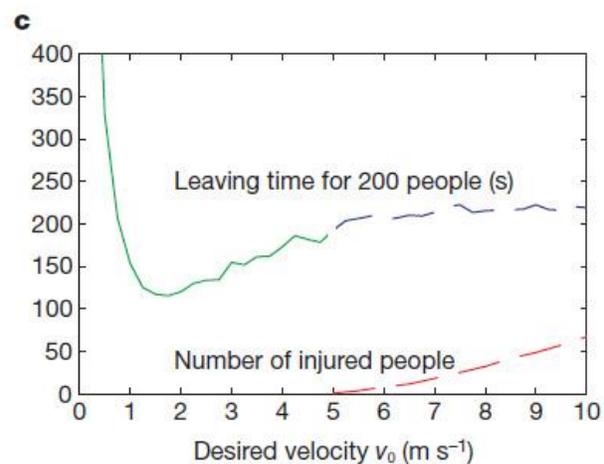
- 根据非恐慌状态下人流的状况来选择参数;
- 行人用半径不同的园来描述;



# 阻塞场景的模拟



纵坐标表示：  
不同个体离开的时间



“faster-is-slower effect” 欲速则不达效应

# 羊群效应

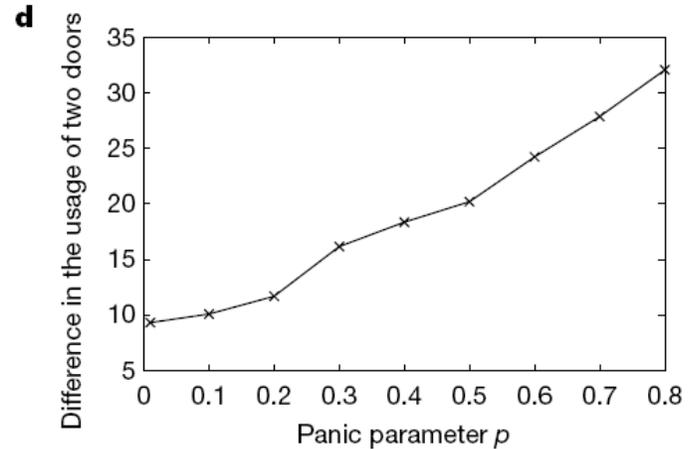
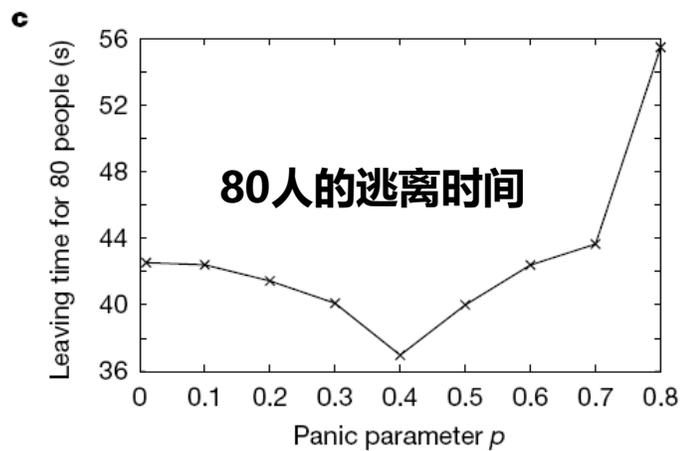
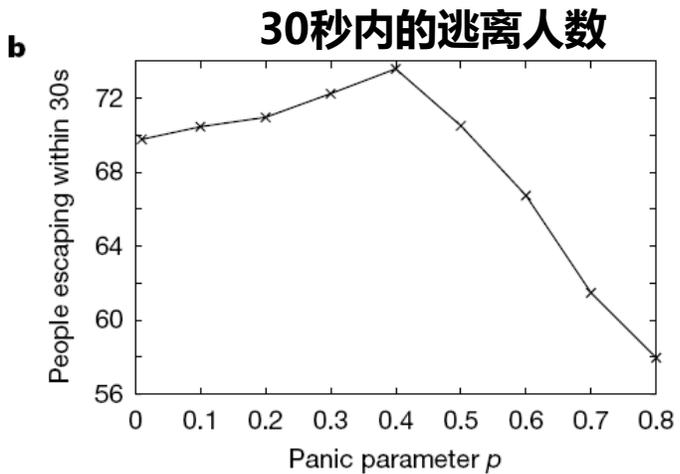
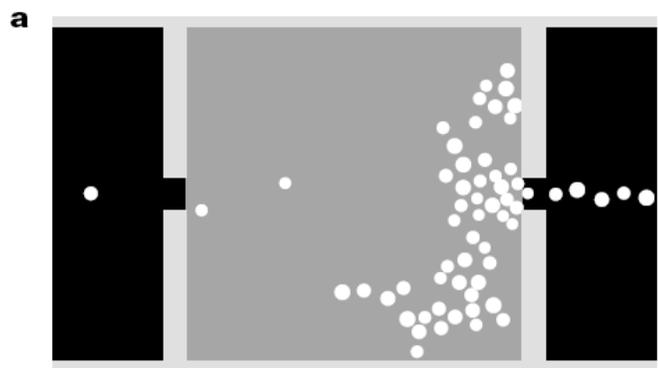
- 恐慌的人群呈现羊群效应、个体行为或两者的混合；
- 羊群效应用参数 $p_i$ 来模拟；

$$\mathbf{e}_i^0(t) = \text{Norm} \left[ (1 - p_i) \mathbf{e}_i + p_i \langle \mathbf{e}_j^0(t) \rangle_i \right]$$

↓  
个体行为的方向

↓  
个体 $i$ 邻居行为的平均

# 羊群效应



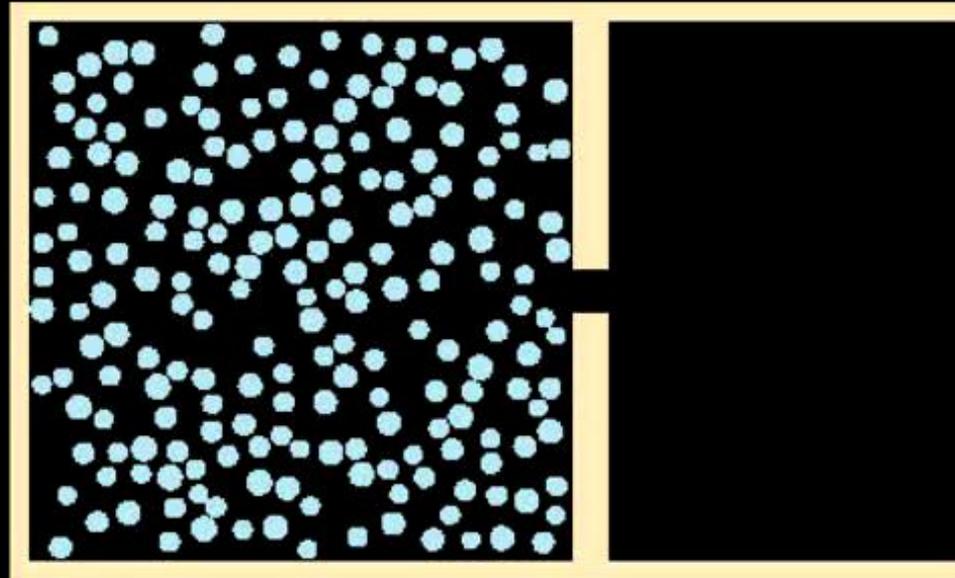
若恐慌参数 $P$ 增大，则左右两个出口的人数差异变大

# Demo

$t = 0$

$N = 200$

$V_0 = 1$



# 参考源程序

- **C Source code available from <http://www.panics.org/>**

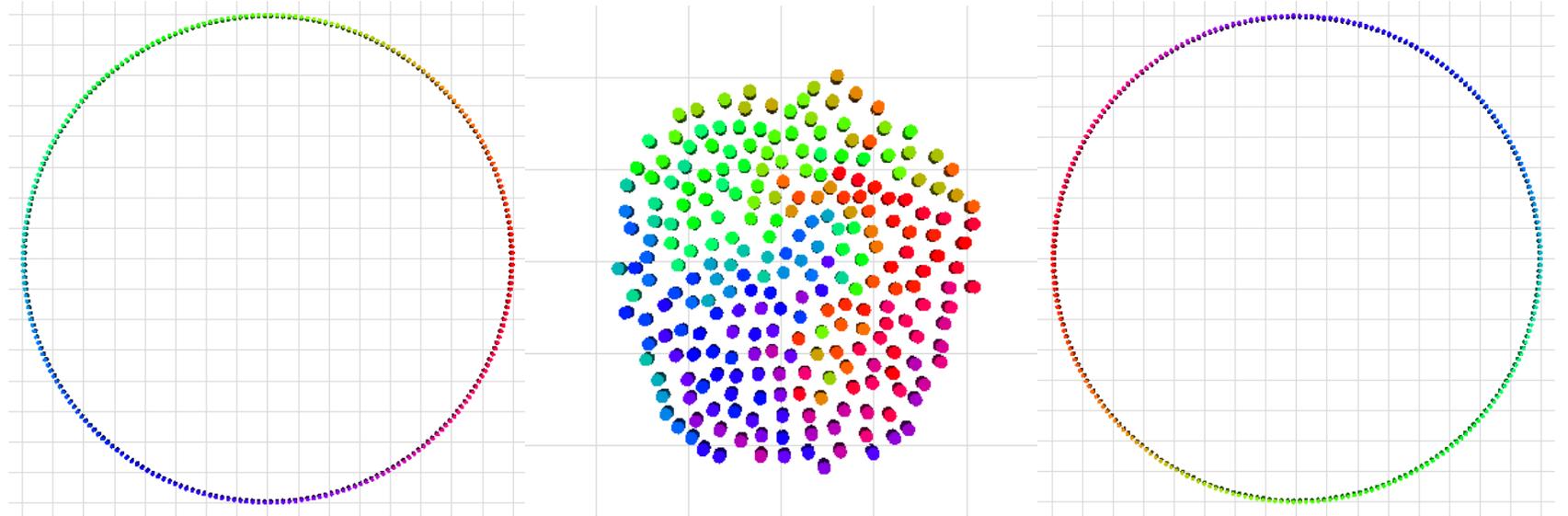


- **Self-Organized Pedestrian Crowd Dynamics-An applet simulating intersecting pedestrian streams**

**by Frederik Depta, Lijuan Liu and Nibras A Suleman**

**<http://itp.uni-frankfurt.de/~gros/JavaApplets/PedestrianCrowdDynamics/PedestrianApplet.html>**

# 相互速度障碍物碰撞避免模型 (RVO, Reciprocal Velocity Obstacles)



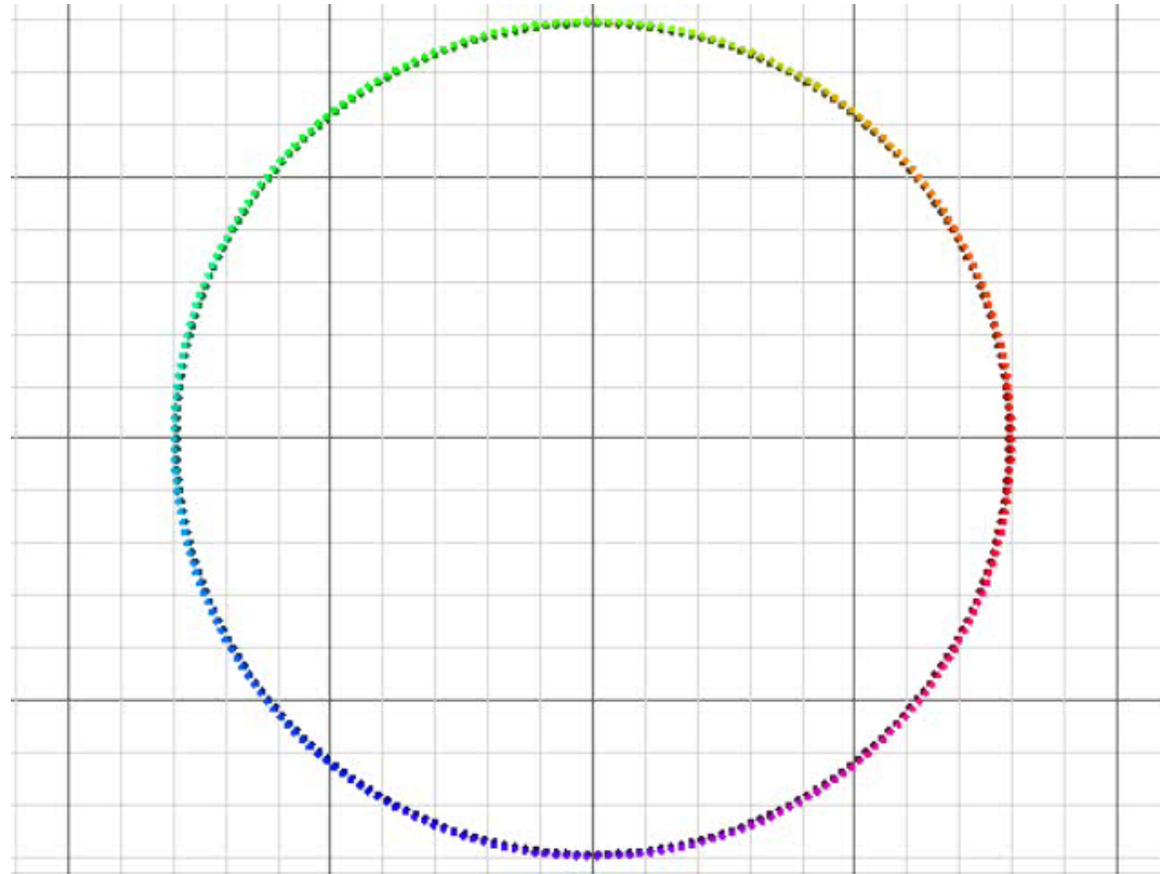
## □ 参考文献

Jur van den Berg, Ming Lin, Dinesh Manocha, “**Reciprocal Velocity Obstacles** for Real-Time Multi-Agent Navigation”, IEEE International Conference on Robotics and Automation (ICRA), 2008, pp. 1928-1935.

# 背景

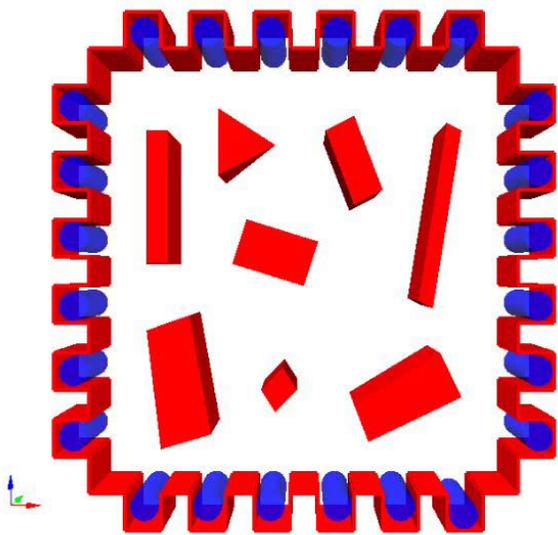
- 在群组动画中，一个关键问题是如何避免个体之间的碰撞。在现实世界中，当两个行人判断前行可能发生碰撞时，两个人**会同时调整自己的速度**以避免碰撞；
- 尽管碰撞避免是群体动画中常见的行为，但由于涉及到每个个体的运动能力、心理、外界环境等多种因素，这种相互作用复杂性高，合理的碰撞避免是一项极具挑战性的课题；
- **相互速度障碍物(RVO)**模型是一个很好的解决方案
  - 假设个体以**匀速前进**
  - 每个个体在保持与周围个体相对运动的同时，**在速度域中计算出可能导致碰撞的速度集合**，并对自身速度进行必要的调整。在调整过程中，碰撞避免的任务**同时分配给**相关的个体，使它们**相互协调完成**碰撞避免任务。

# 要解决什么问题？



一个园形的例子: 每个智能体(agent)走到园的正对面位置

# 多智能体(multi-agent)导航



- $N$ 个智能体共享一个环境
- 一个智能体从起始点运动到目标点, 相互之间不发生碰撞, 并避开障碍物

- 解耦处理法(Decoupled)
  - 每个智能体同时自主导航
  - 全局路径规划和局部碰撞避免分开处理
- 集中处理法(Centralized)
  - 在复合配置空间进行路径规划
    - 例子: 大型团体操

# 问题描述

- 独立导航
- 感知和行动是连续的
- 全局路径规划 vs. 局部导航
- 在每个周期：每个智能体能观察到其它智能体(知道其它智能体的位置和速度)
- 但是不知道它们准备做什么
- 智能体该如何做？

# 速度障碍物

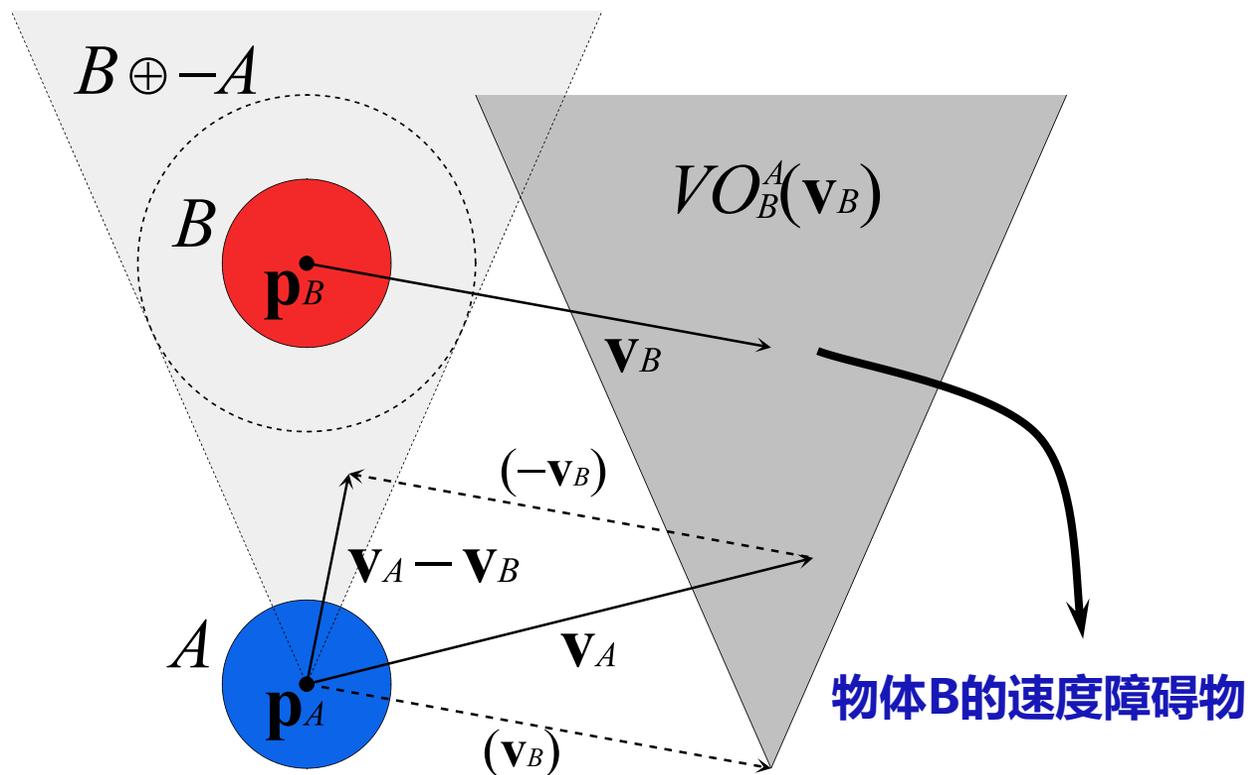
- 假设 $A$ 为平面上的一个智能体，它的位置 $\mathbf{p}_A$ ， $B$ 为一个移动的平面障碍物，其位置为 $\mathbf{p}_B$

- 则障碍物 $B$ 对智能体 $A$ 的**速度障碍物**

$VO_B^A(\mathbf{v}_B)$ 为

- 在某一时刻将与障碍物 $B$ (以速度 $\mathbf{v}_B$ 移动)碰撞的智能体 $A$ 的所有可能的速度 $\mathbf{v}_A$ 集合

- 在几何上，速度障碍物可以用右面的图来表示：



圆盘形障碍物 $B$ 对一个圆盘形智能体 $A$ 的速度障碍物 $VO_B^A(\mathbf{v}_B)$

# 速度障碍物

- 假设 $A \oplus B$ 表示 $A$ 和 $B$ 两个物体的Minkowski和,  $-A$ 表示物体 $A$ 在它参考点的镜像,

$$A \oplus B = \{ \mathbf{a} + \mathbf{b} \mid \mathbf{a} \in A, \mathbf{b} \in B \}, -A = \{ -\mathbf{a} \mid \mathbf{a} \in A \}$$

- $\gamma(\mathbf{p}, \mathbf{v}) = \{ \mathbf{p} + t\mathbf{v} \mid t > 0 \}$ 表示以 $\mathbf{p}$ 为起点, 方向为 $\mathbf{v}$ 的光线。

# 速度障碍物

- 如果以 $\mathbf{P}_A$ 为起始点，方向为物体A和B两个物体的相对速度 $(\mathbf{v}_A - \mathbf{v}_B)$ 的光线与B和-A的Minkowski和相交，则速度 $\mathbf{v}_A$ 位于B的速度障碍物中。

$$VO_B^A = \{ \mathbf{v}_A / \lambda(\mathbf{P}_A, \mathbf{v}_A - \mathbf{v}_B) \cap B \oplus -A \neq \emptyset \}$$

- 如果 $\mathbf{v}_A$ 属于上述集合，则在某一时刻物体A和B将碰撞；如果 $\mathbf{v}_A$ 不属于B的速度障碍物集合，则两个物体将不会碰撞；如果 $\mathbf{v}_A$ 位于速度障碍物的边界，则物体A在某一时刻与物体B接触。

# 速度障碍物的性质

- 对称性质

$$\mathbf{v}_A \in VO_B^A(\mathbf{v}_B) \Leftrightarrow \mathbf{v}_B \in VO_A^B(\mathbf{v}_A)$$

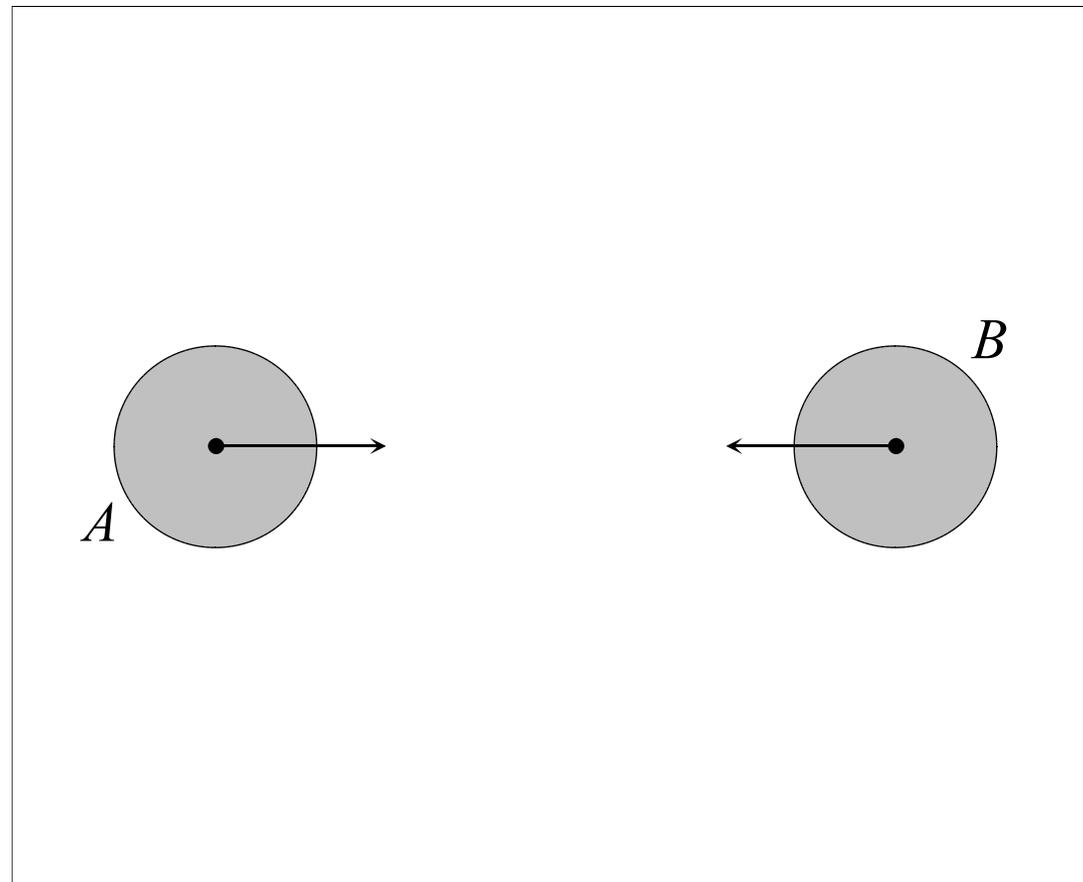
- 平移不变性

$$\mathbf{v}_A \in VO_B^A(\mathbf{v}_B) \Leftrightarrow \mathbf{v}_A + \mathbf{u} \in VO_B^A(\mathbf{v}_B + \mathbf{u})$$

# 速度障碍物的应用

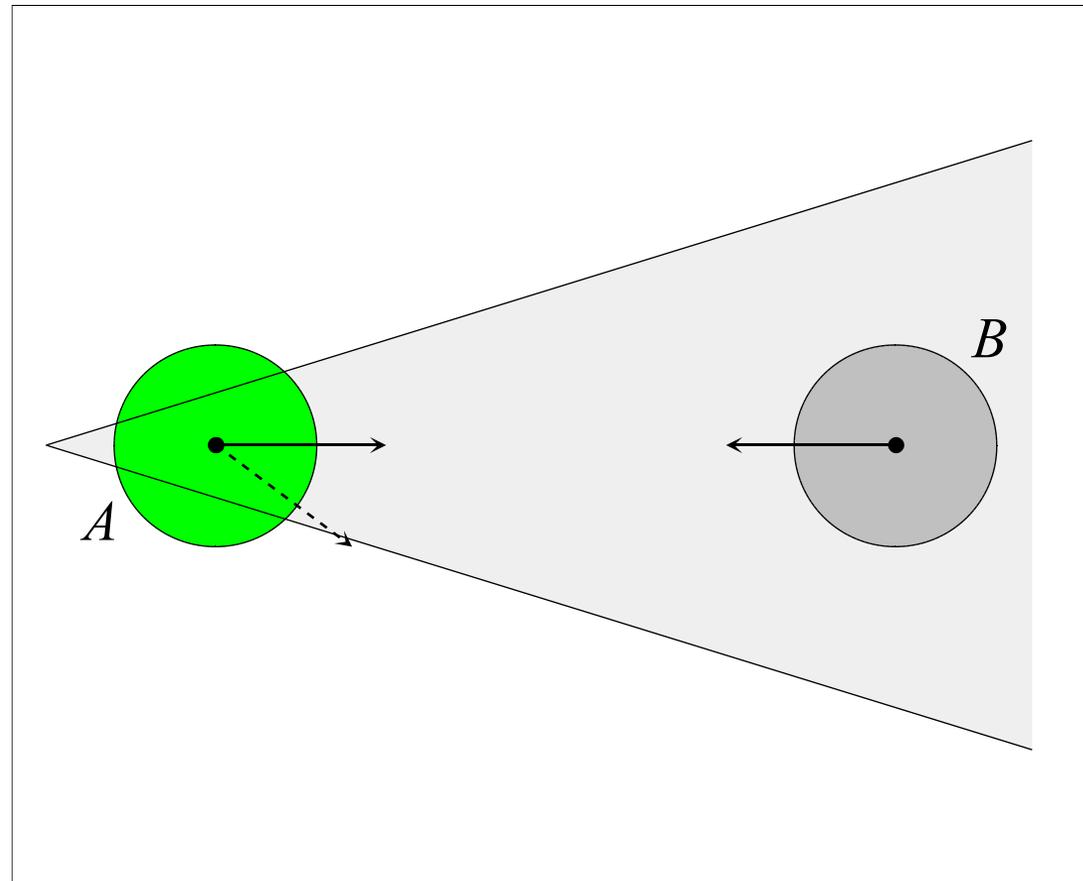
- 速度障碍物这一概念可用于移动物体的导航。在每个规划周期内，智能体选择一个位于移动物体速度障碍物外面的速度；
- 如果有很多自由的速度可选，则选择的速度为导引智能体到目标位置的**最合适**的速度；
- 多智能体的导航？
- 但是智能体不是被动地移动，而是**相互之间**做出反应；
- 结果：**振荡(Oscillation)!**

# 振荡



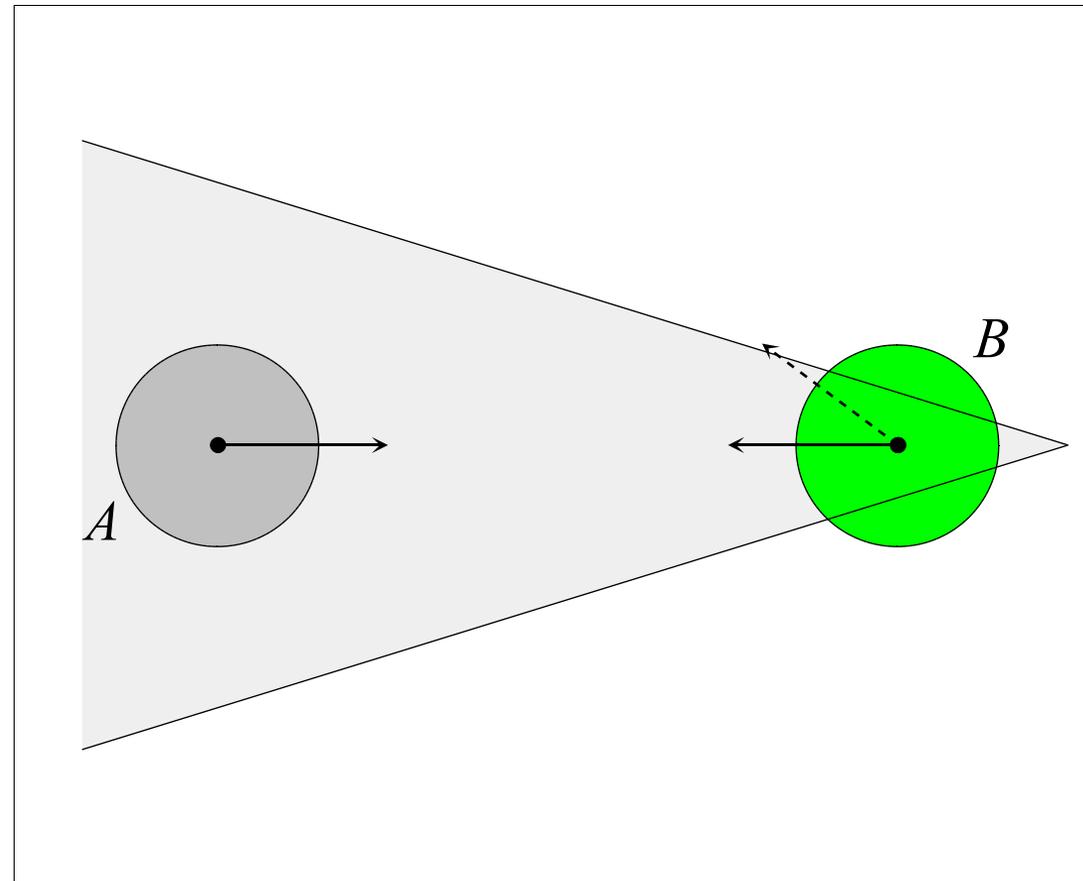
例子: 以相对方向运动的两个智能体

# 振荡



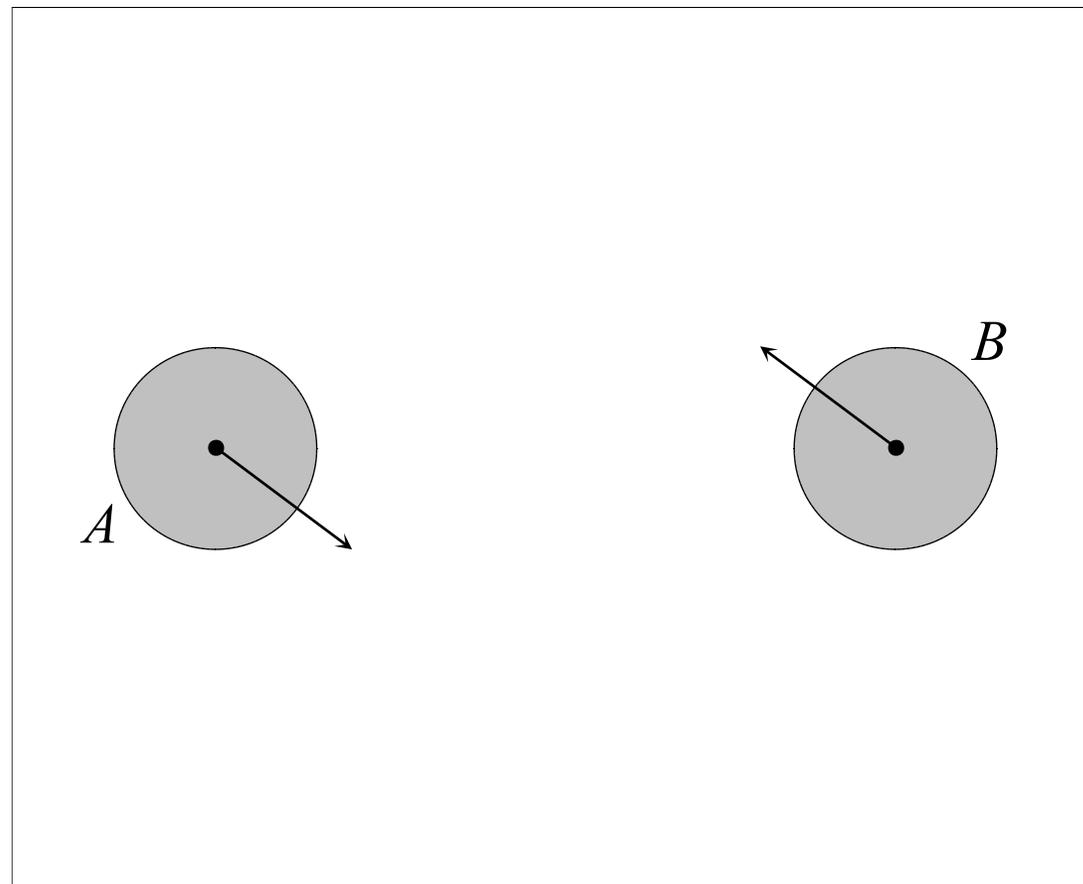
例子: 以相对方向运动的两个智能体

# 振荡



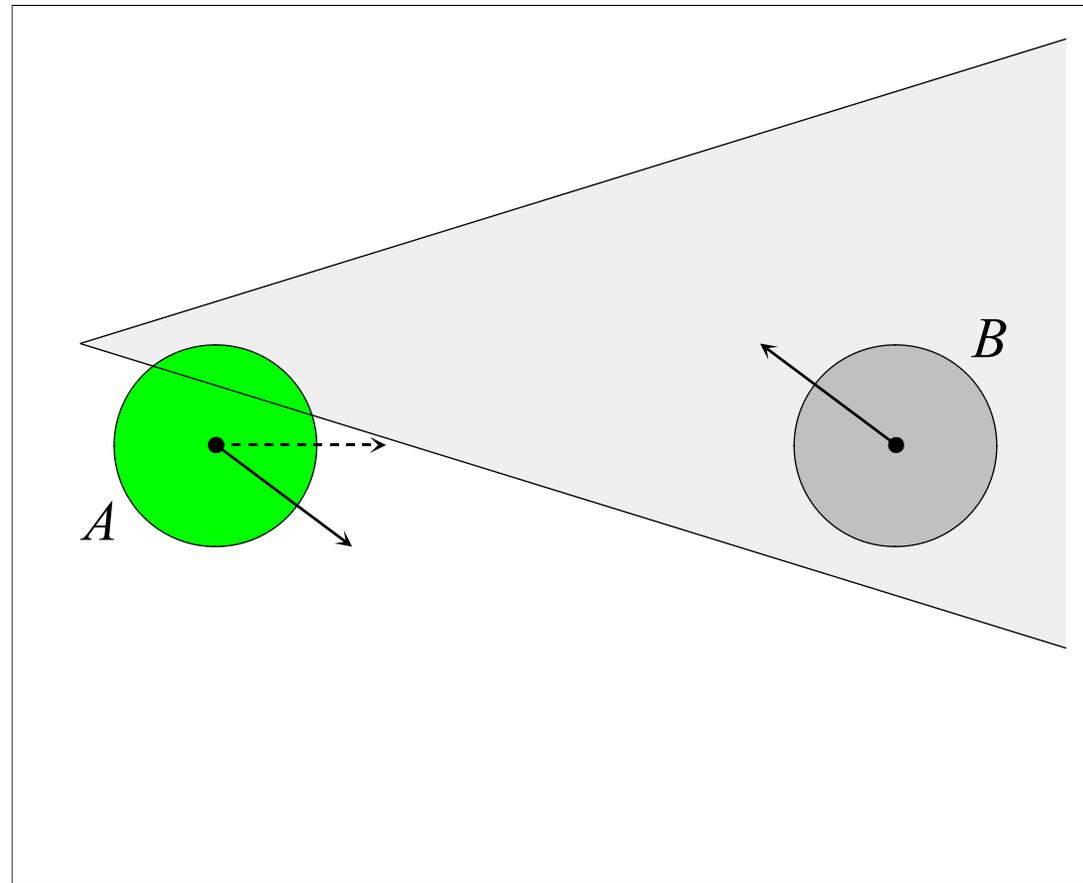
例子: 以相对方向运动的两个智能体

# 振荡



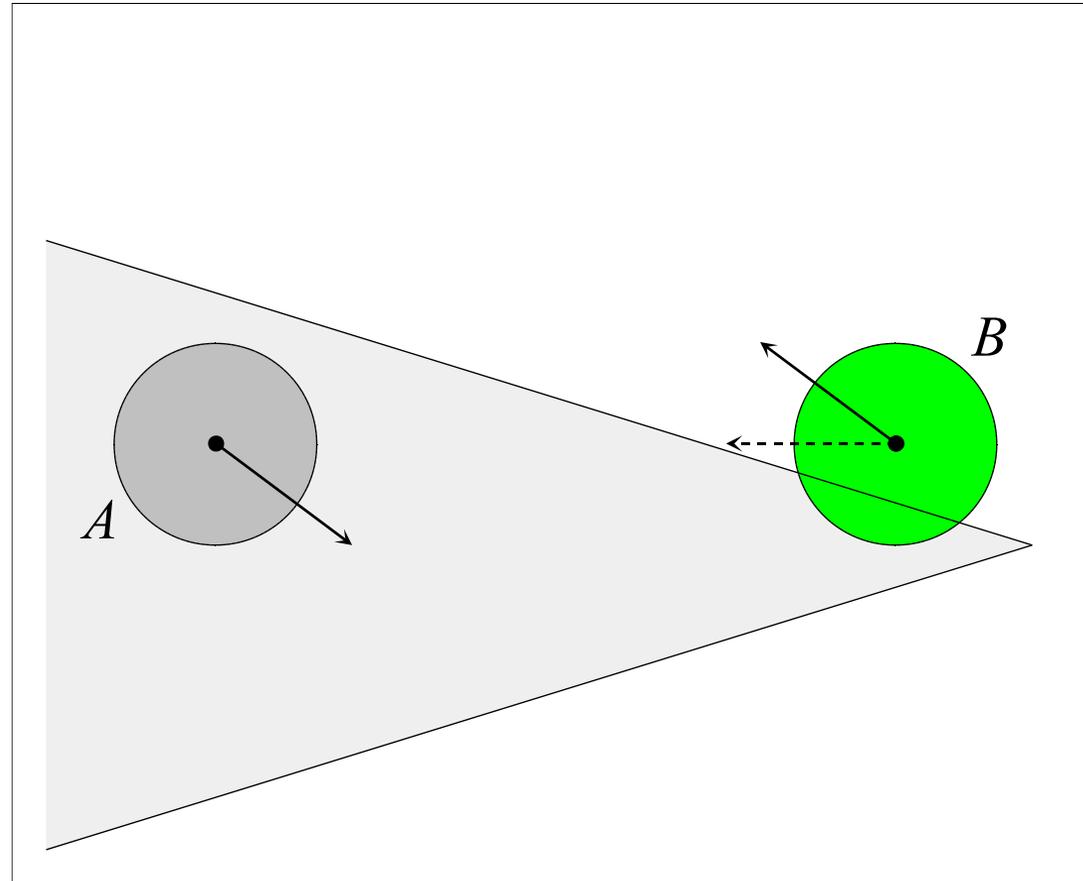
例子: 以相对方向运动的两个智能体

# 振荡



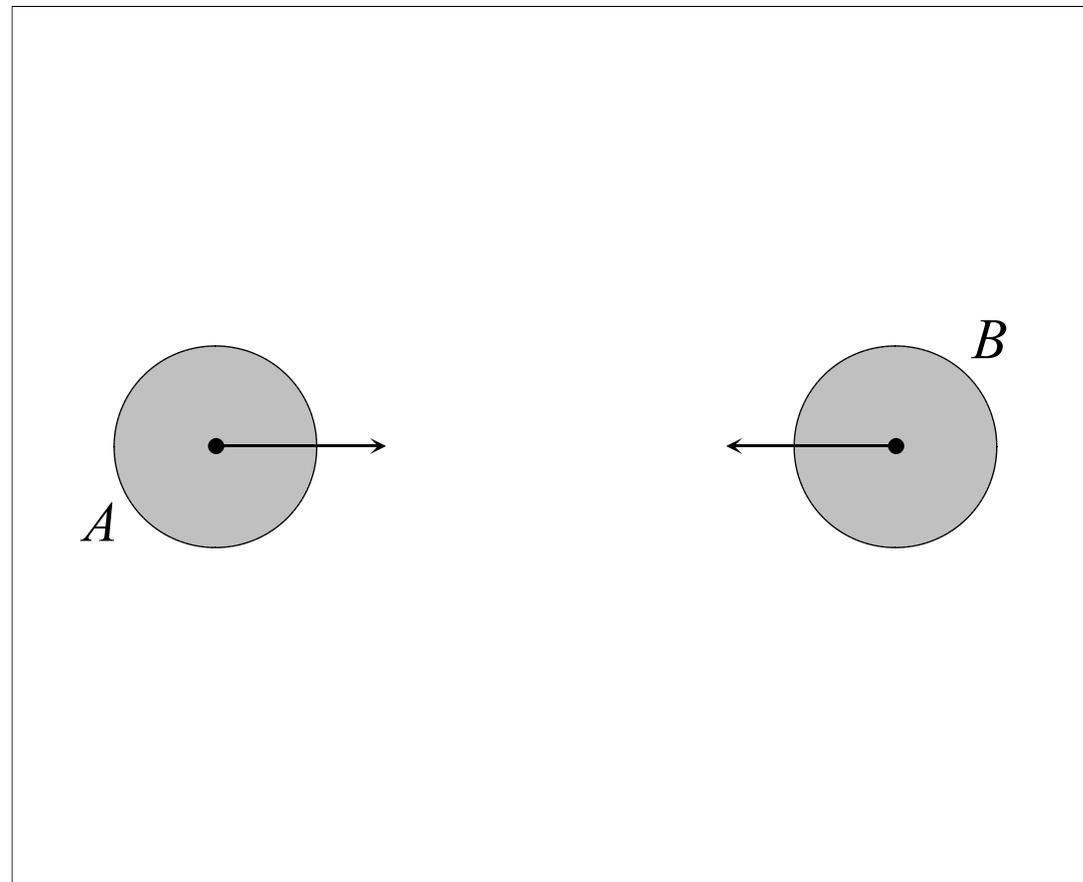
例子: 以相对方向运动的两个智能体

# 振荡



例子: 以相对方向运动的两个智能体

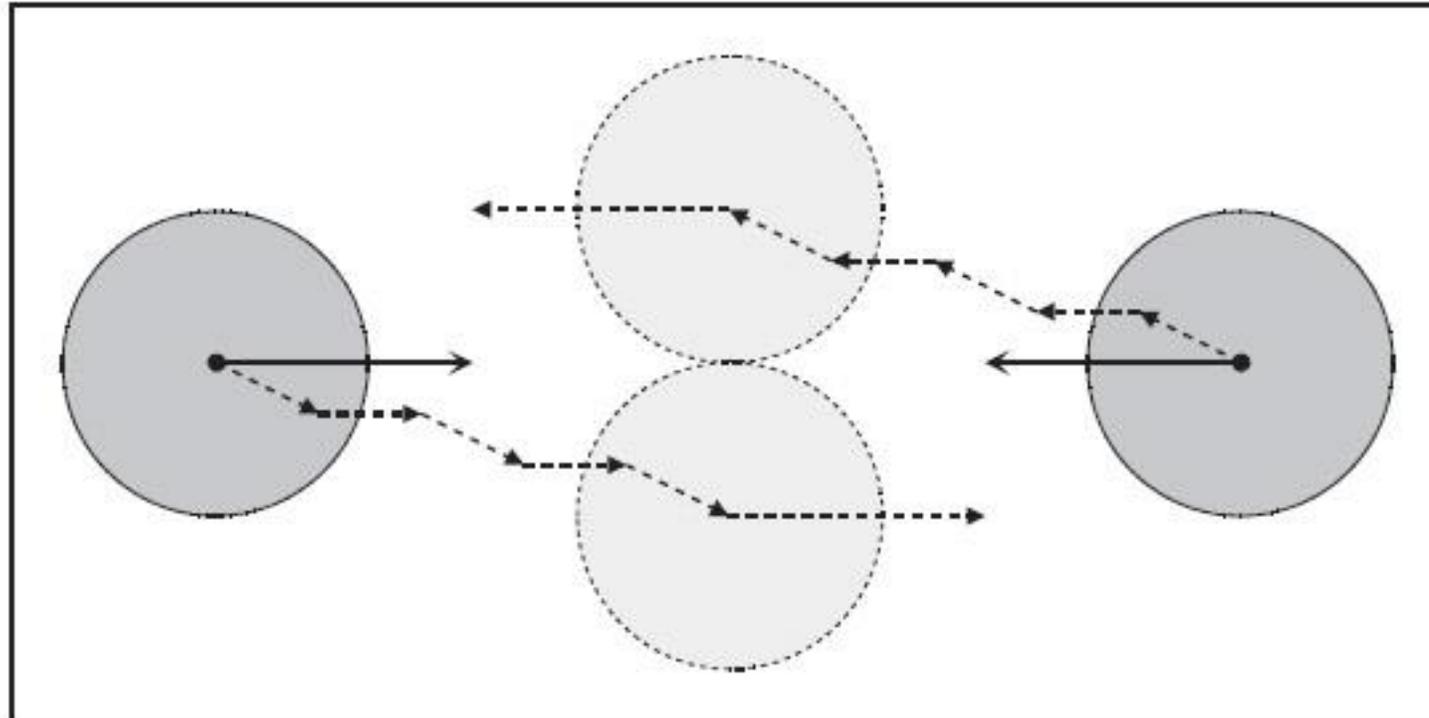
# 振荡



例子: 以相对方向运动的两个智能体

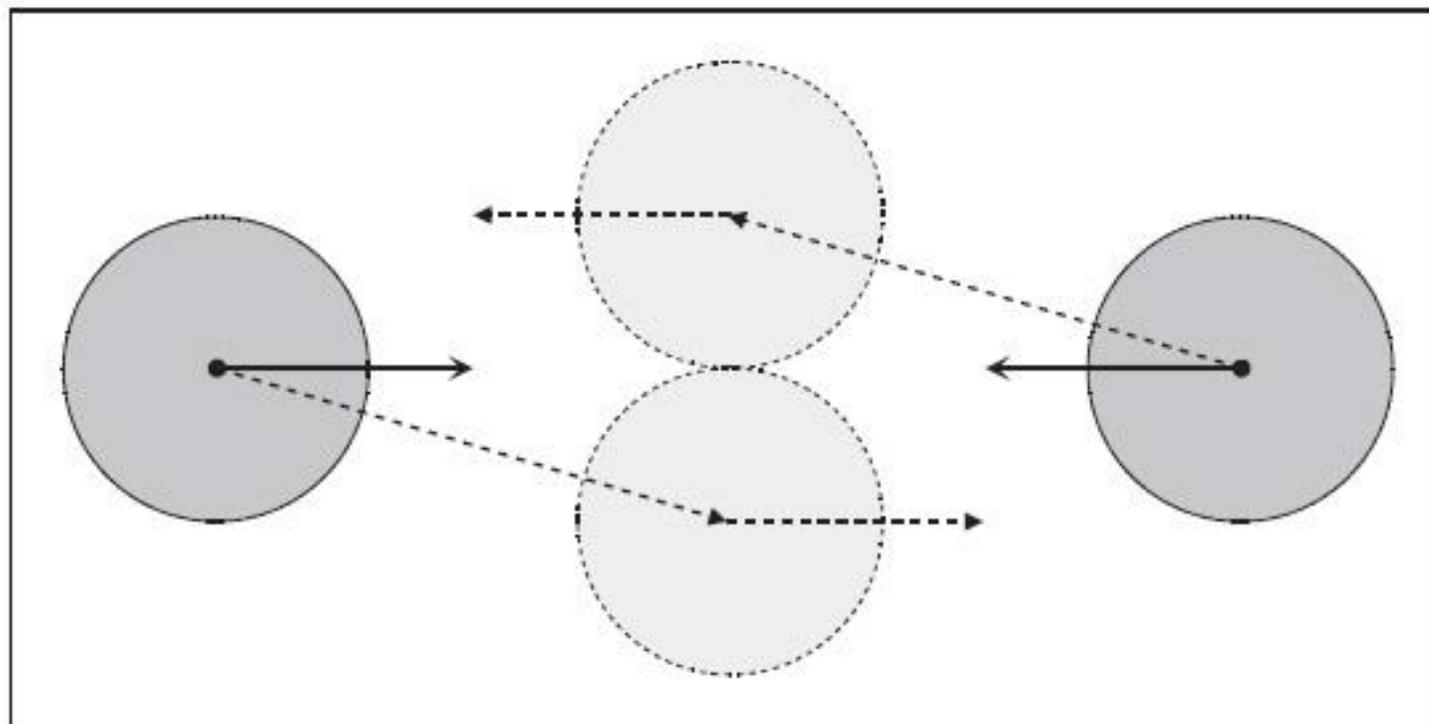


# 振荡导致的路径



例子: 以相对方向运动的两个智能体

# 更好的路径



例子: 以相对方向运动的两个智能体

# 相互速度障碍物方法

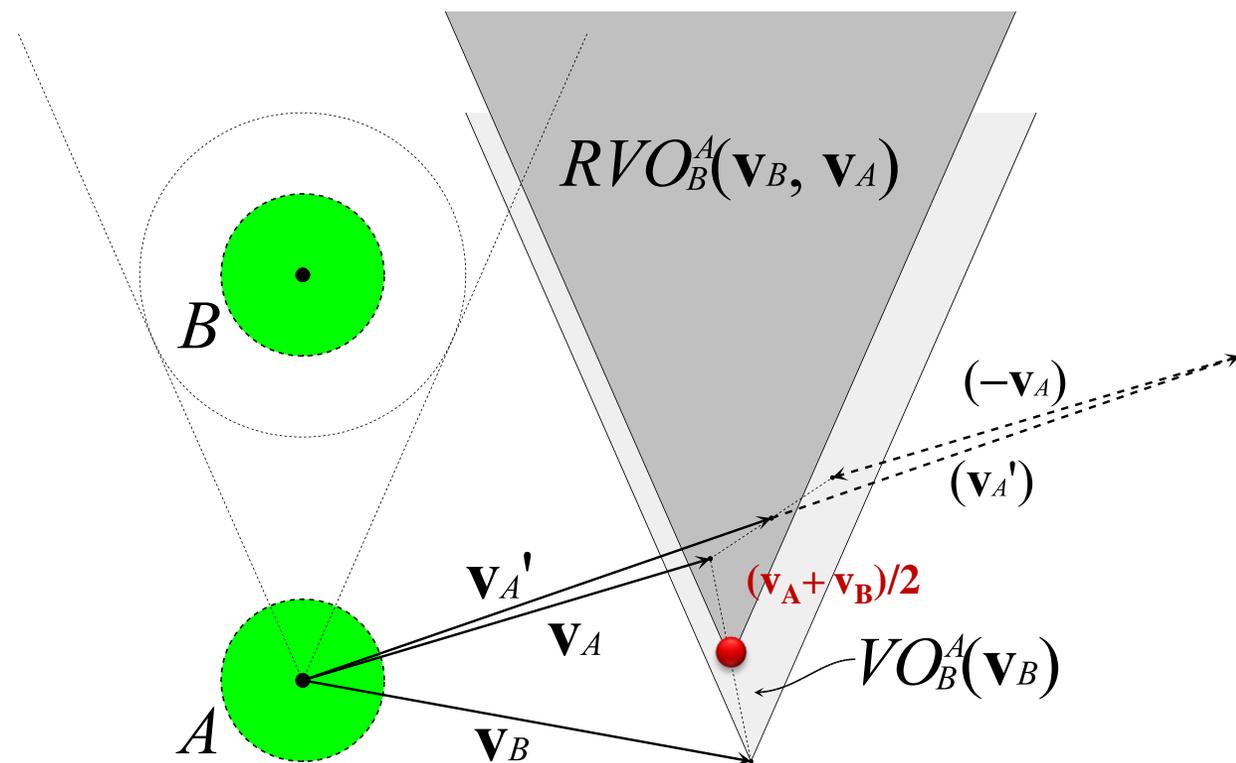
- 想办法避免振荡;
- **原因**: 智能体之间没有通讯或整体的协调;
- **简单的想法**: 不是选择位于速度障碍物外部的一个新的速度, 而是选择**当前速度**和一个位于速度障碍物外部的速度的**平均**;
- 形成**相互速度障碍物**(**RVO**, Reciprocal **V**elocity **O**bstacle)

# 相互速度障碍物

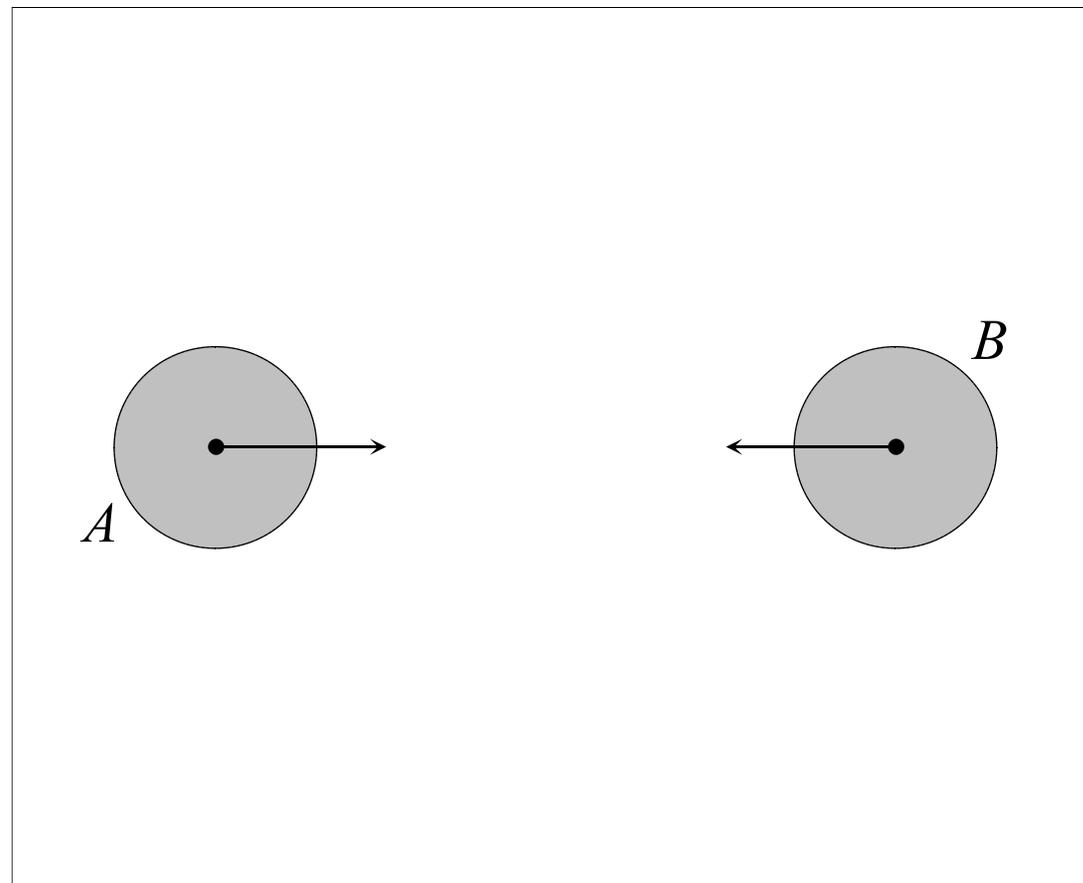
- $B$ 对 $A$ 的**相互**速度障碍物包含 $A$ 的所有以下速度的集合：该集合的成员为 $A$ 的当前速度 $\mathbf{v}_A$ 和 $B$ 的速度障碍物内的一个速度的平均：

$$RVO_B^A(\mathbf{v}_B, \mathbf{v}_A) = \left\{ \mathbf{v}_A' / 2 \mathbf{v}_A' - \mathbf{v}_A \in VO_B^A(\mathbf{v}_B) \right\}$$

- 几何上，可解释为把 $B$ 的速度障碍物 $VO_B^A(\mathbf{v}_B)$ 平移到顶点 $(\mathbf{v}_A + \mathbf{v}_B)/2$ ；

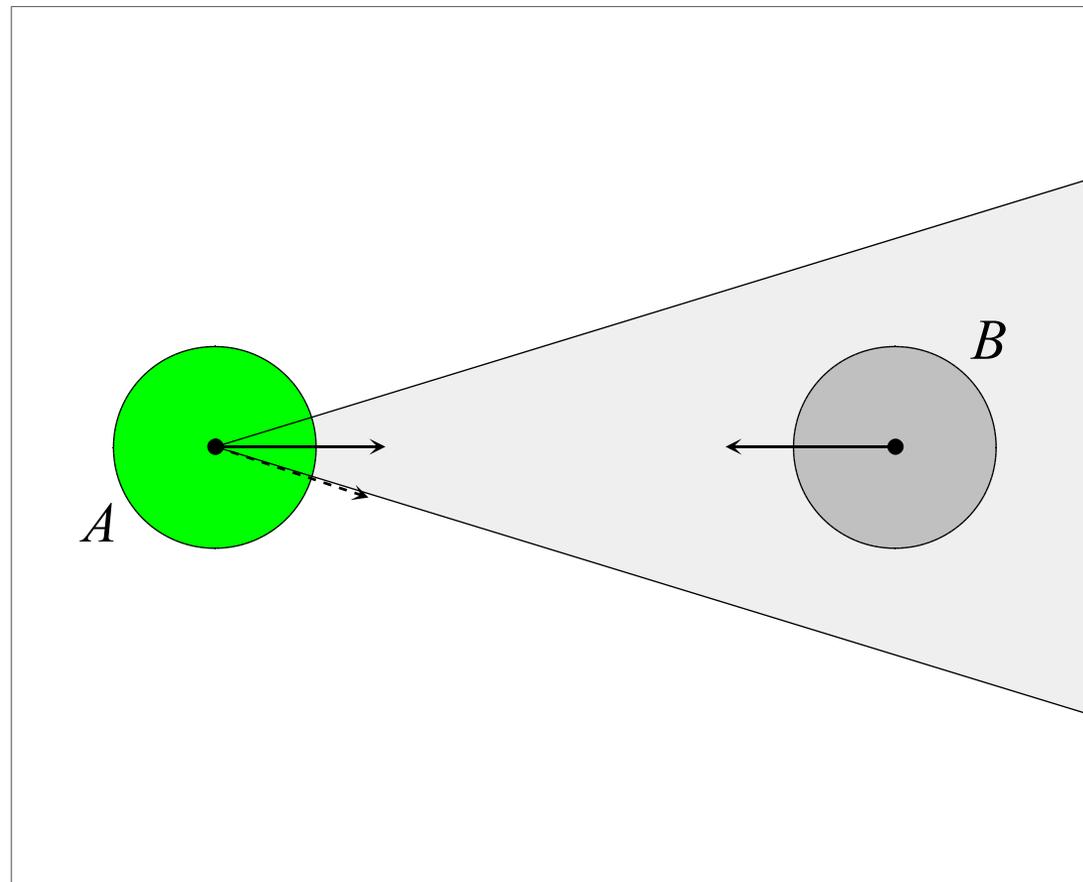


# 振荡?



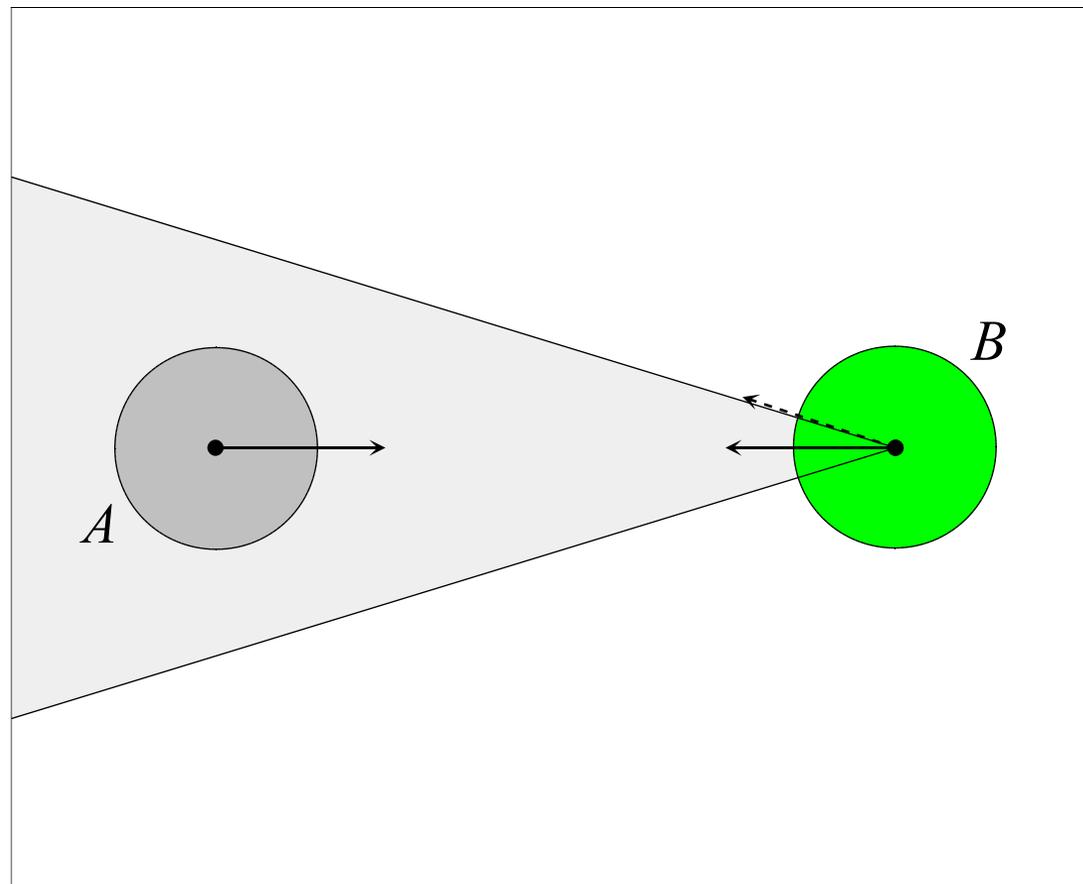
例子: 以相对方向运动的两个智能体

# 振荡?



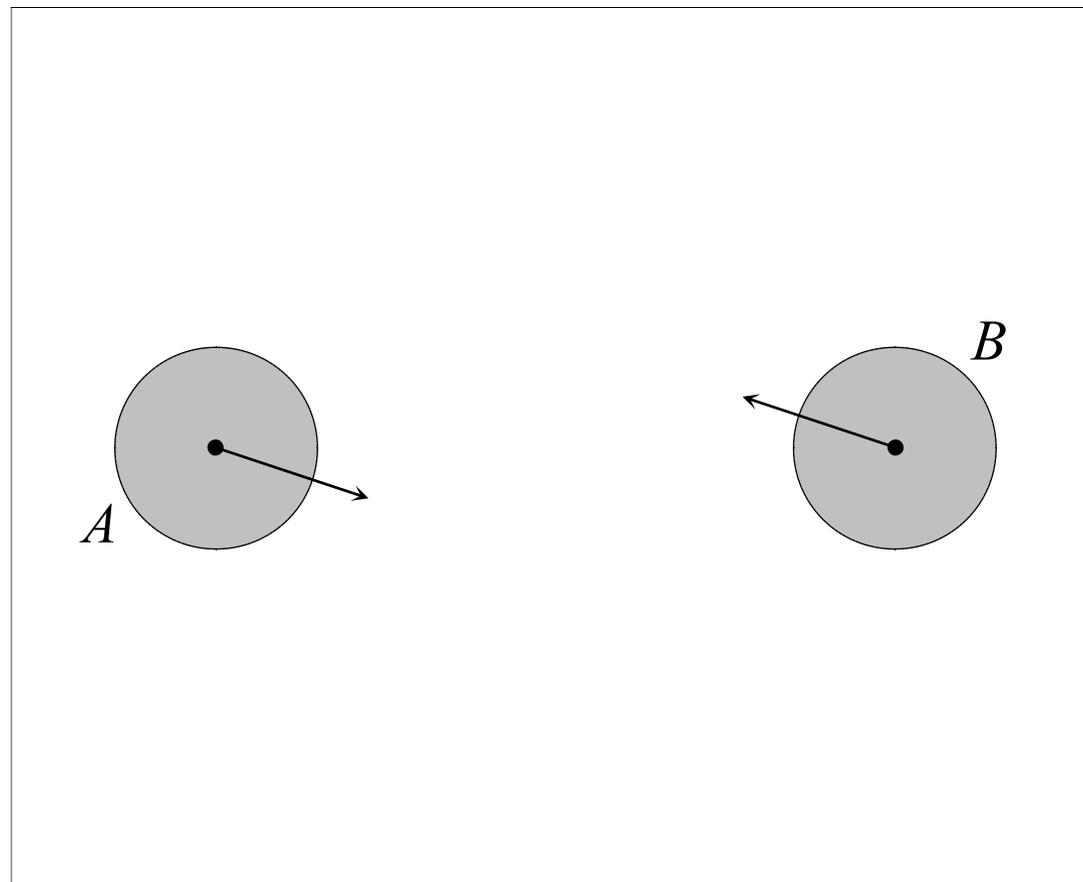
例子: 以相对方向运动的两个智能体

# 振荡?



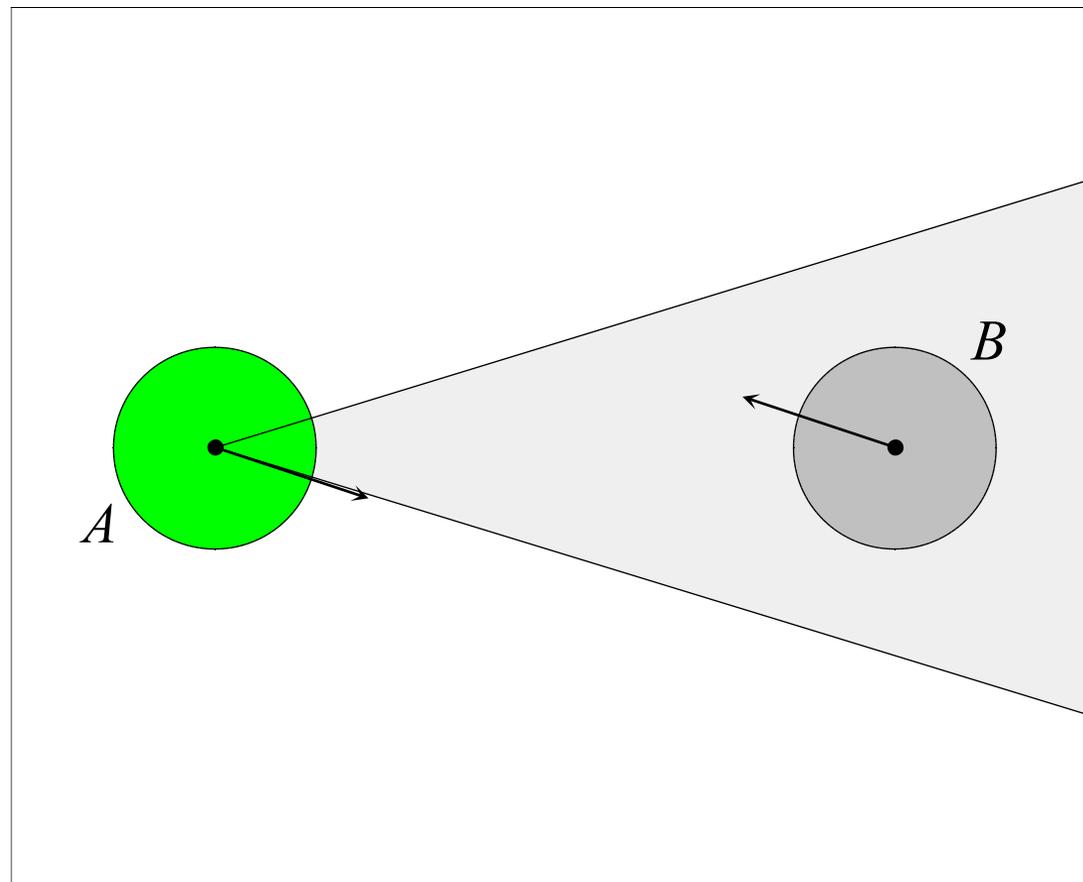
例子: 以相对方向运动的两个智能体

# 振荡?



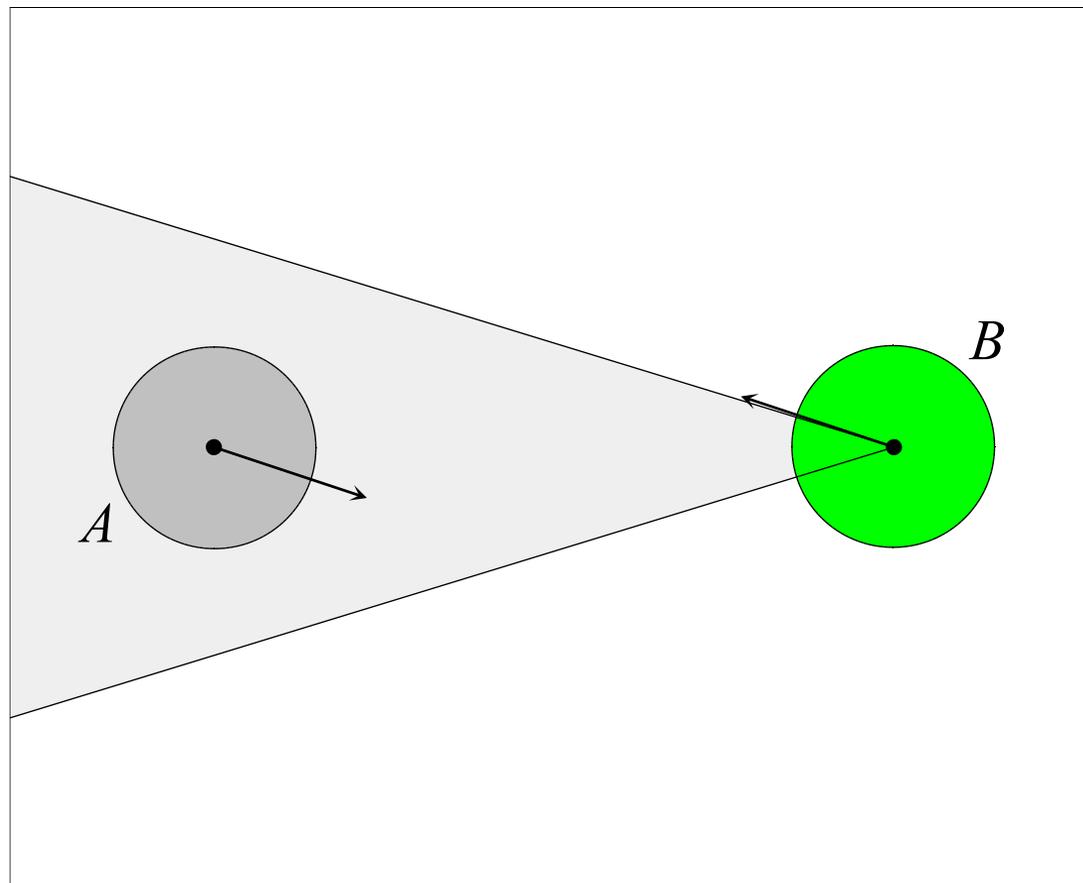
例子: 以相对方向运动的两个智能体

# 振荡?



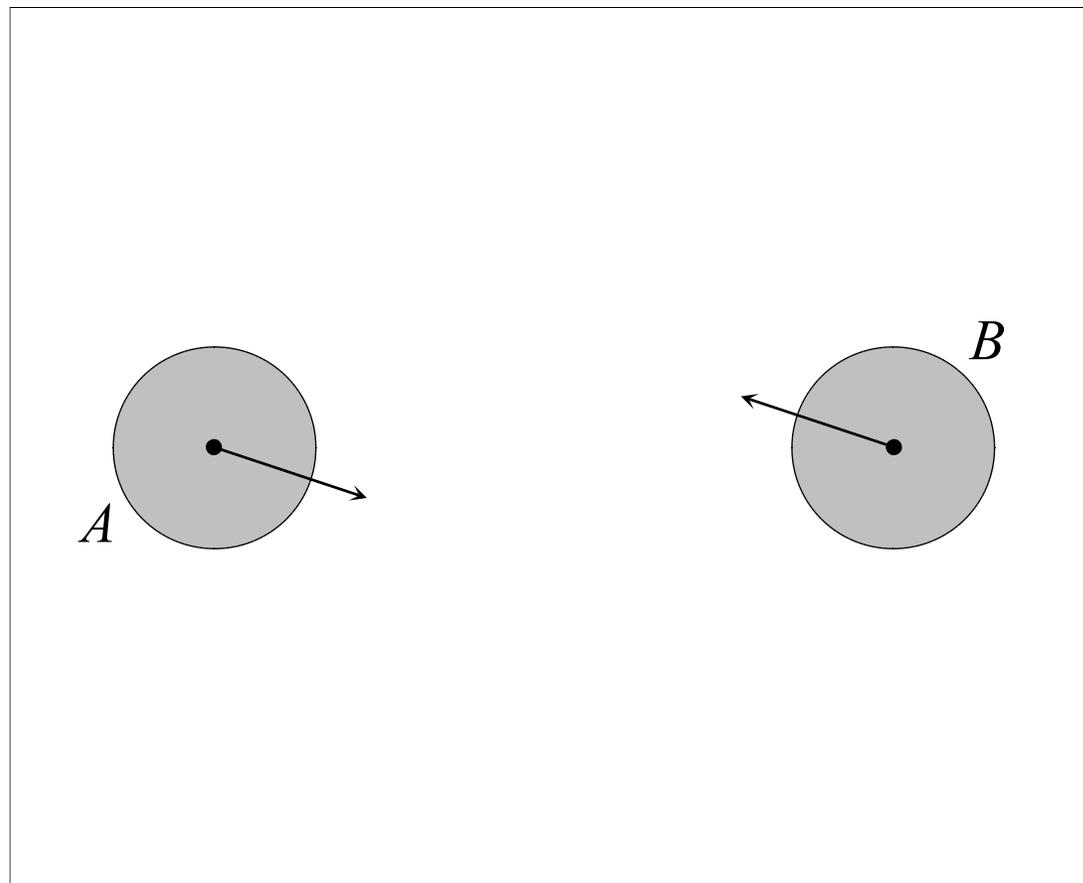
例子: 以相对方向运动的两个智能体

# 振荡?



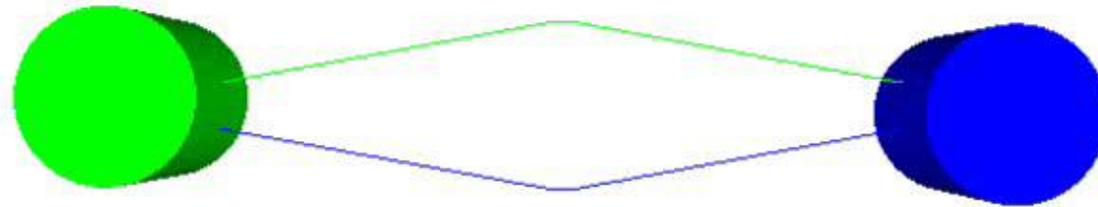
例子: 以相对方向运动的两个智能体

# 振荡?



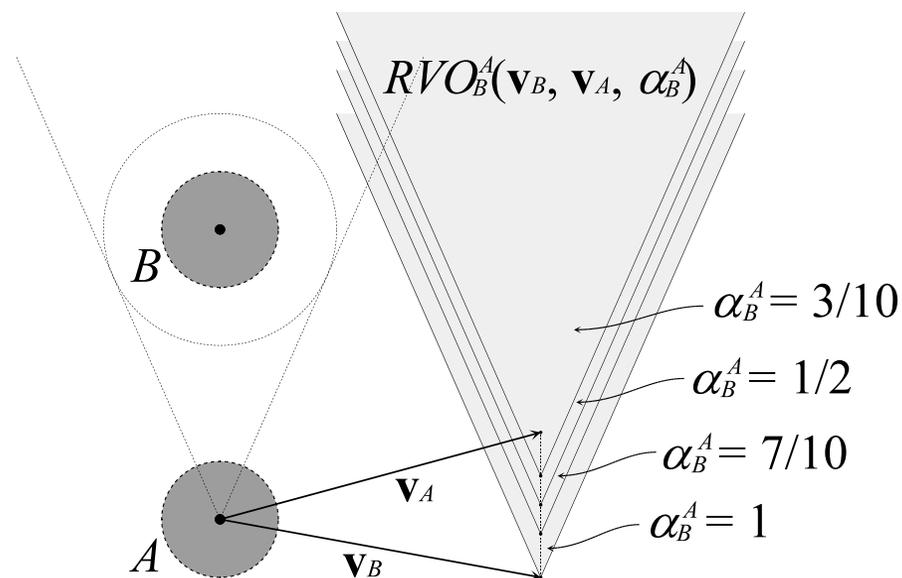
例子: 以相对方向运动的两个智能体

# 没有振荡而且是安全的



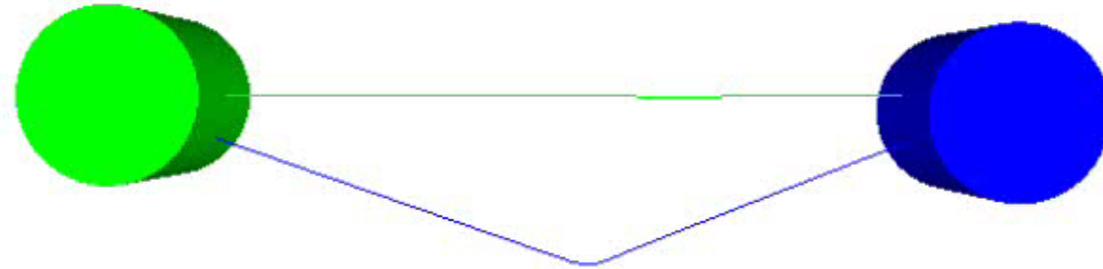
例子: 以相对方向运动的两个智能体

# 广义相互速度障碍物(RVOs)



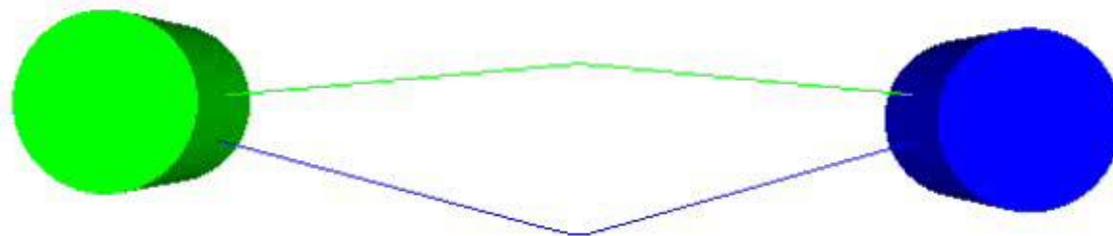
- 不采用50%-50% ， 而采用其它分布。
- 参数 $\alpha$ ;  $\alpha \rightarrow 0$ : A没有努力;  $\alpha \rightarrow 1$ : A尽力
- $RVO_B^A(\mathbf{v}_B, \mathbf{v}_A, \alpha) = \{\mathbf{v}'_A \mid (1/\alpha)\mathbf{v}'_A + (1 - 1/\alpha)\mathbf{v}_A \in VO_B^A(\mathbf{v}_B)\}$

# 广义相互速度障碍物(RVOs)



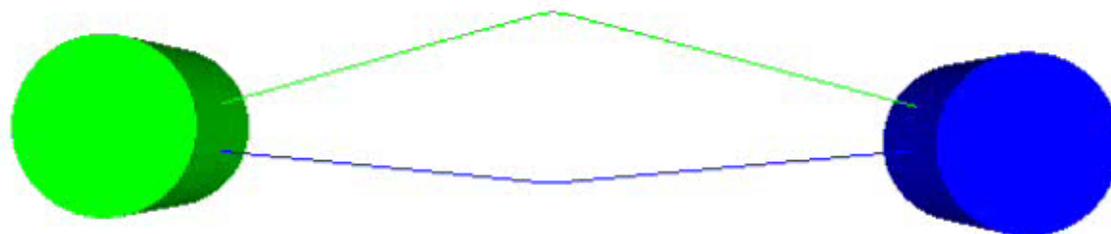
0% - 100%

# 广义相互速度障碍物(RVOs)



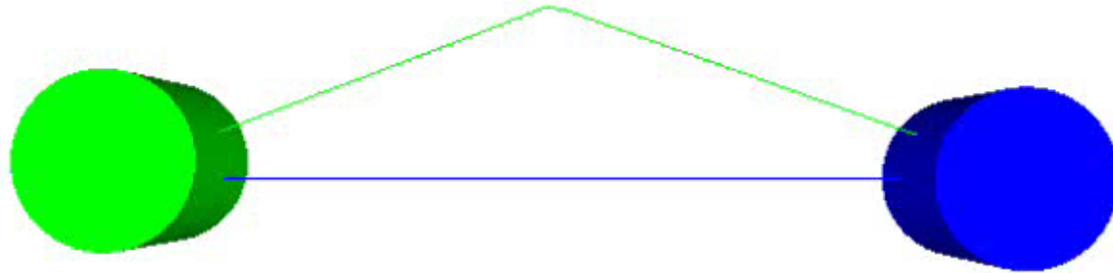
25% - 75%

# 广义相互速度障碍物(RVOs)



75% - 25%

# 广义相互速度障碍物(RVOs)

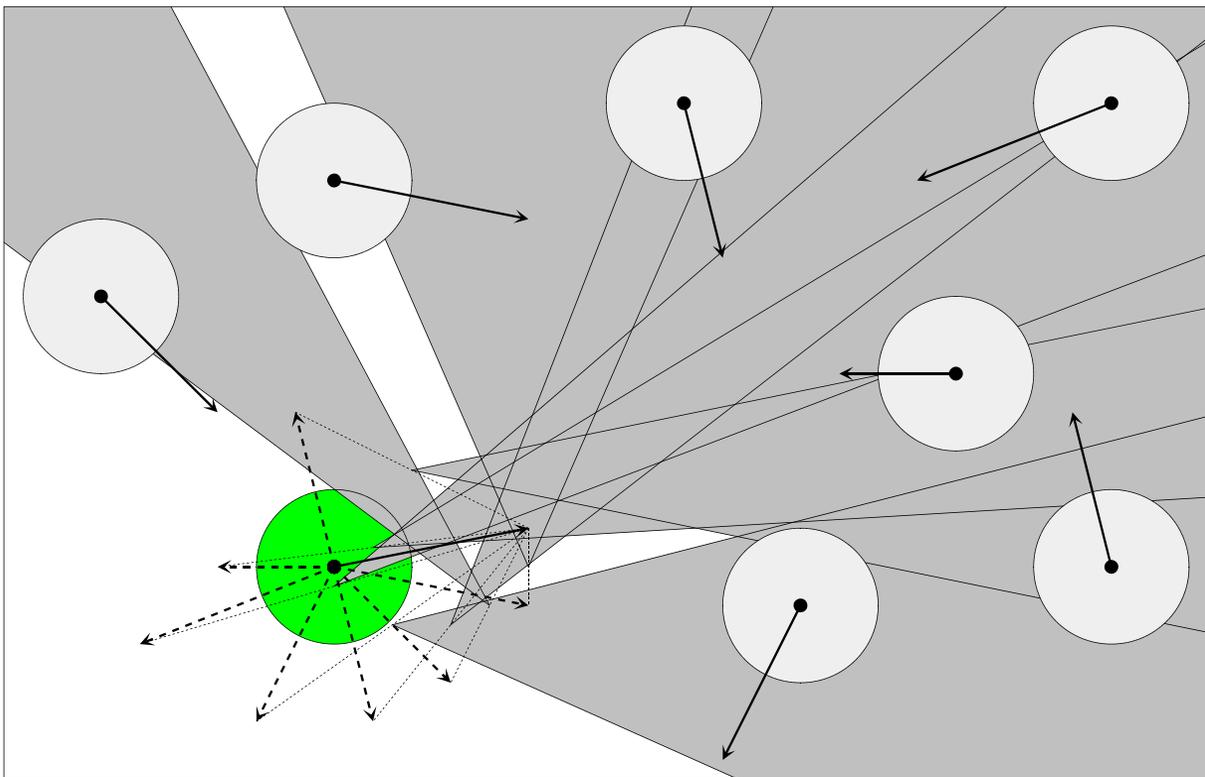


100% - 0%

# 多智能体的导航

- 设有 $N$ 个智能体:  $A_1, \dots, A_N$ , 它们的位置为  $p_1, \dots, p_N$ , 速度为  $\mathbf{v}_1, \dots, \mathbf{v}_N$ , 首选的速度为  $\mathbf{v}_1^{\text{pref}}, \dots, \mathbf{v}_N^{\text{pref}}$ , 目标位置为  $g_1, \dots, g_N$
- 时间步长为  $\Delta t$
- 在每个时间步, 对每个智能体:
  - 计算首选的速度
  - 选择新速度
  - 根据新的速度更新智能体的位置

# 选择新的速度



- 在相互速度障碍物的并集的外面，  
最接近首选的速度

- 环境可能变得很拥挤：没有有效的速度
- 解决方法：在RVO中选择速度，但对下面这些参数进行惩罚：
  - 预计到碰撞的时间
  - 到首选速度的距离
- 以最小的惩罚选择速度

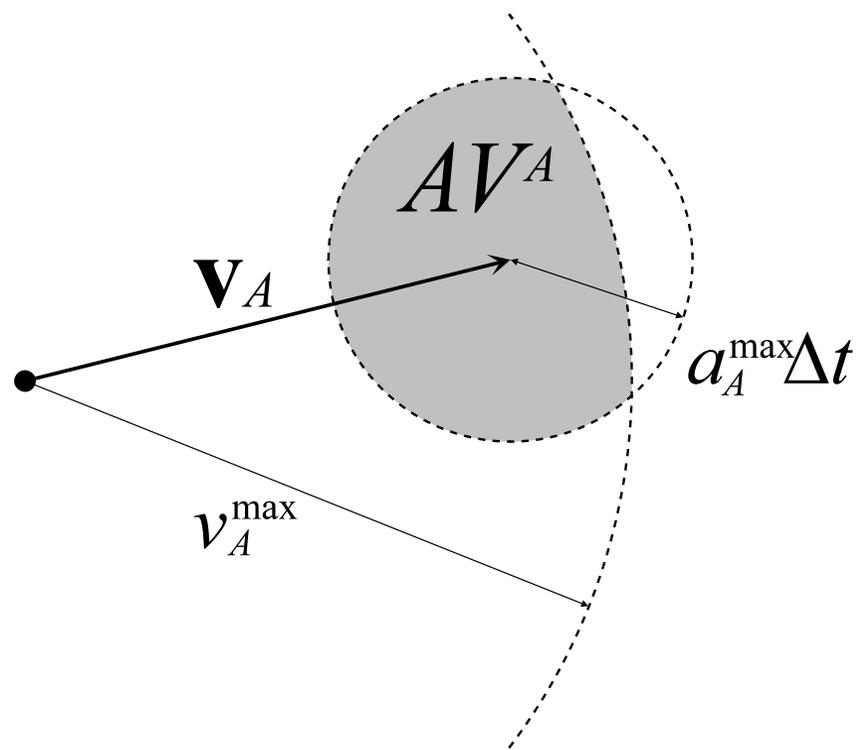
# 真实性

- 动力学约束

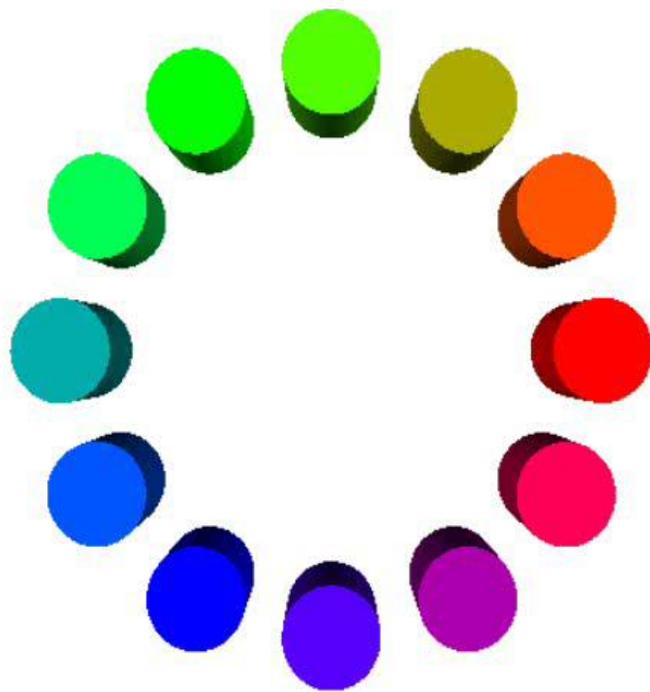
- 最大速度
- 最大加速度
- 其它...

- 邻居

- 圆形邻居
- 可见的邻居区域...

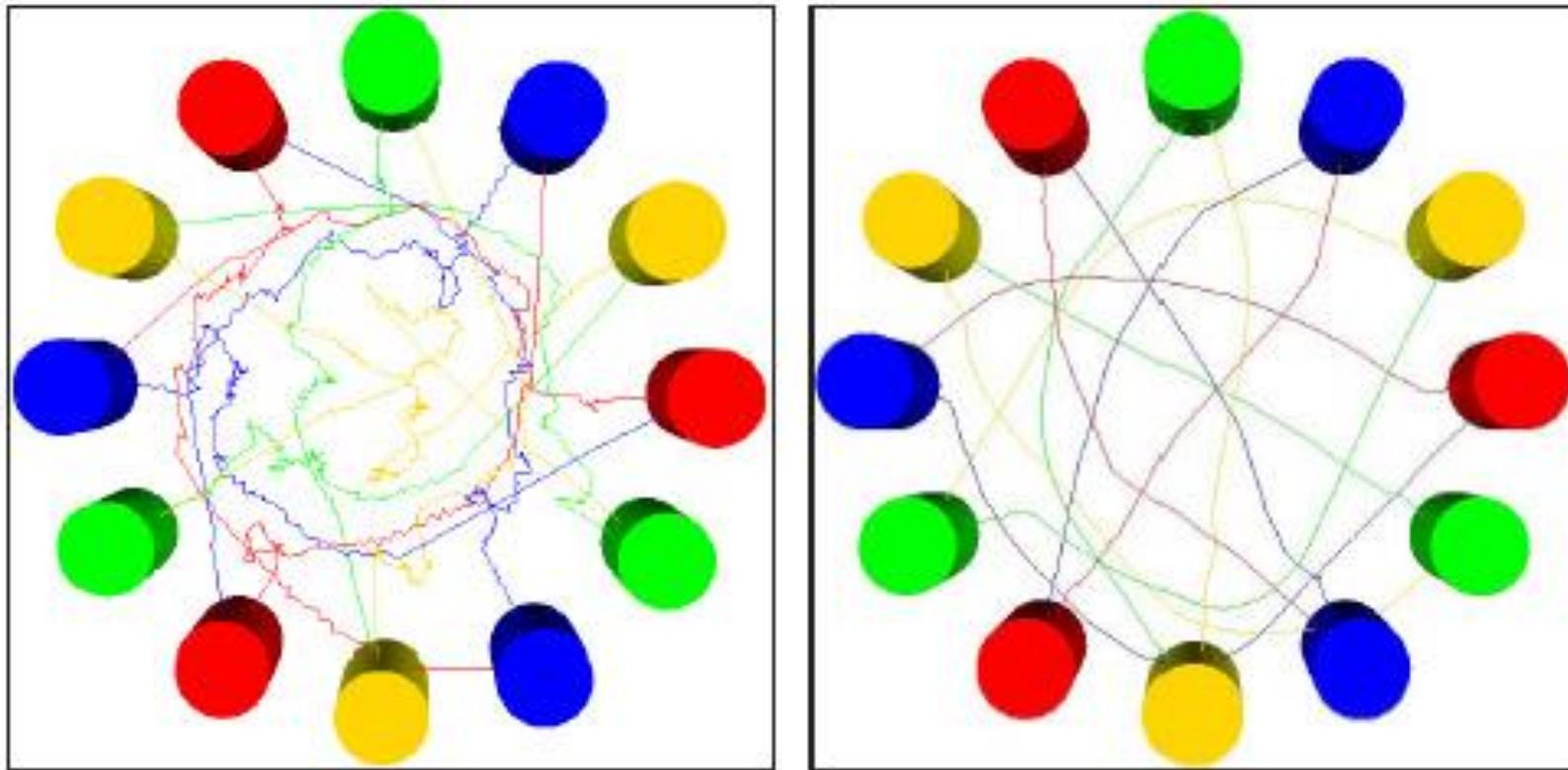


# 实验

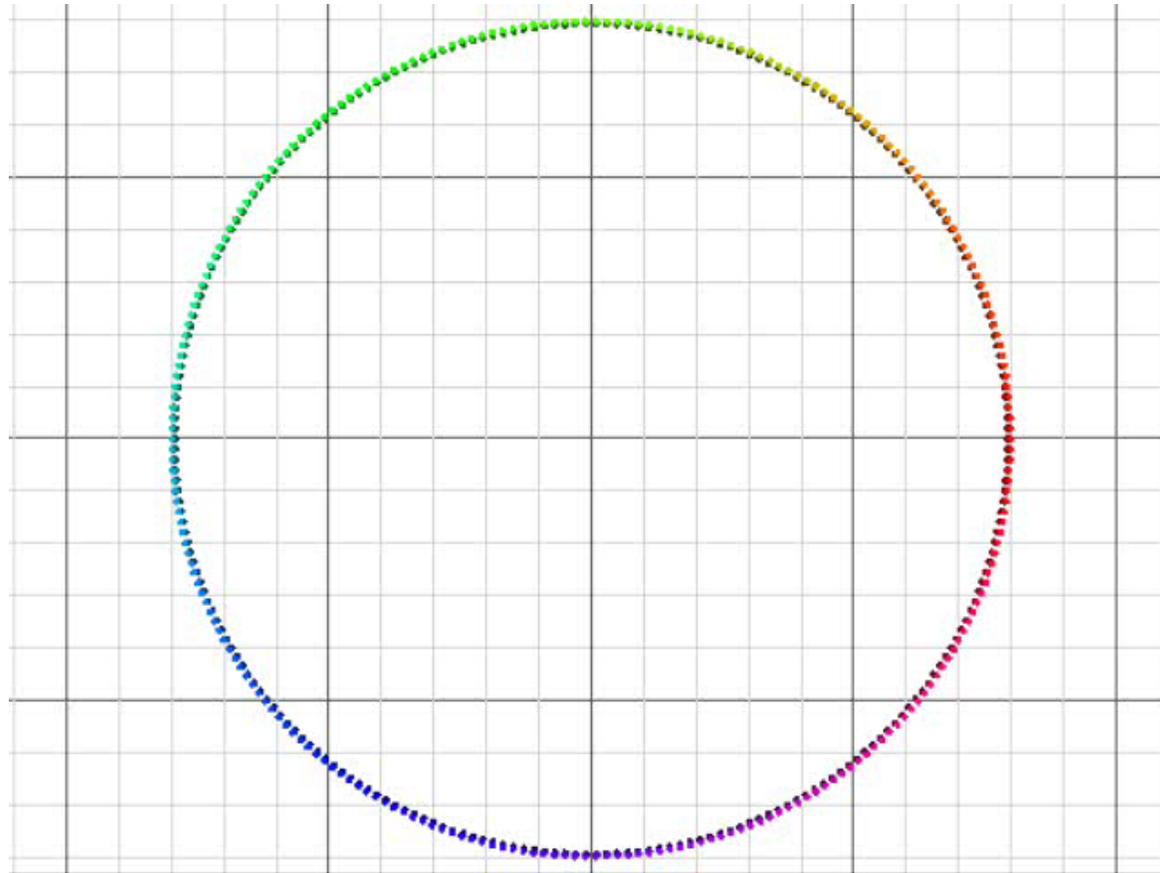


园: 每个智能体运动到园的正对面

# VO和RVO的对比

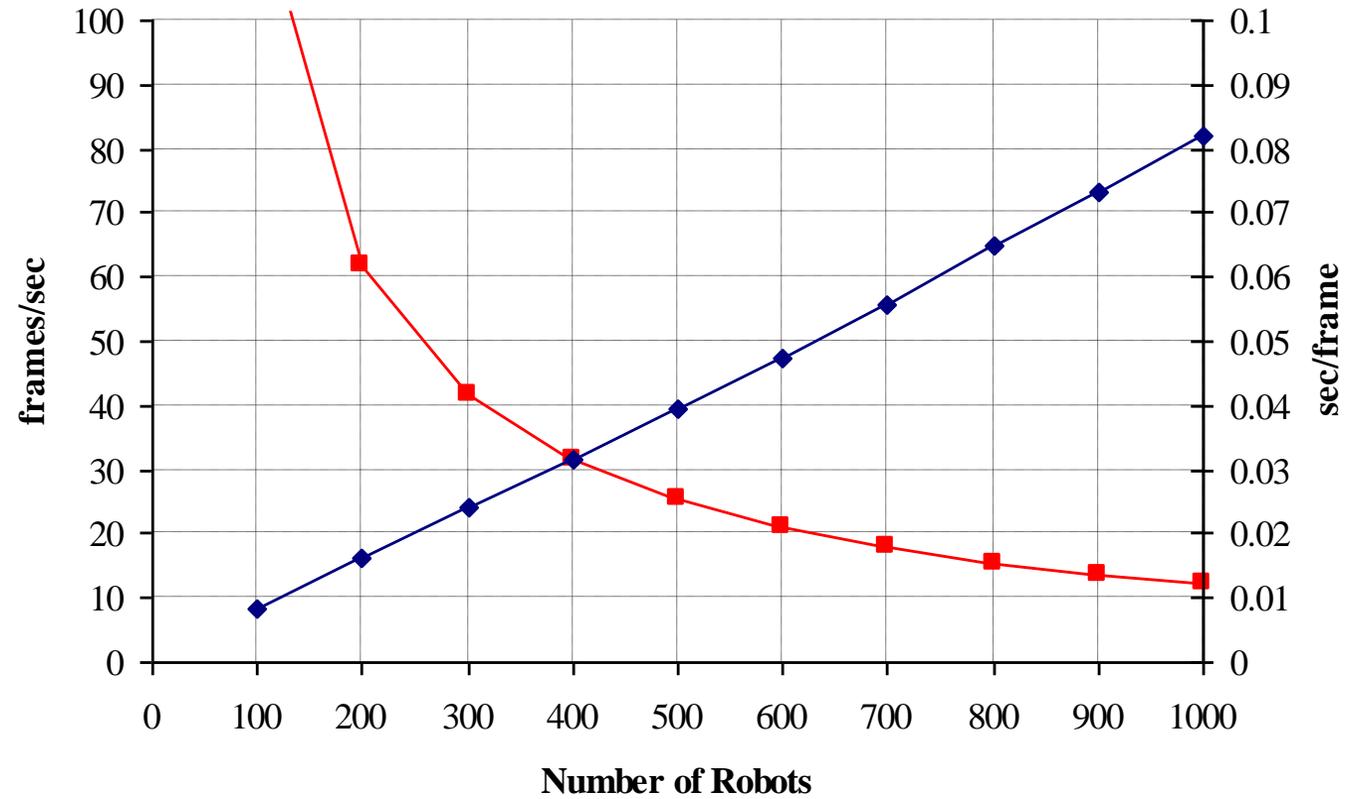


# 实验



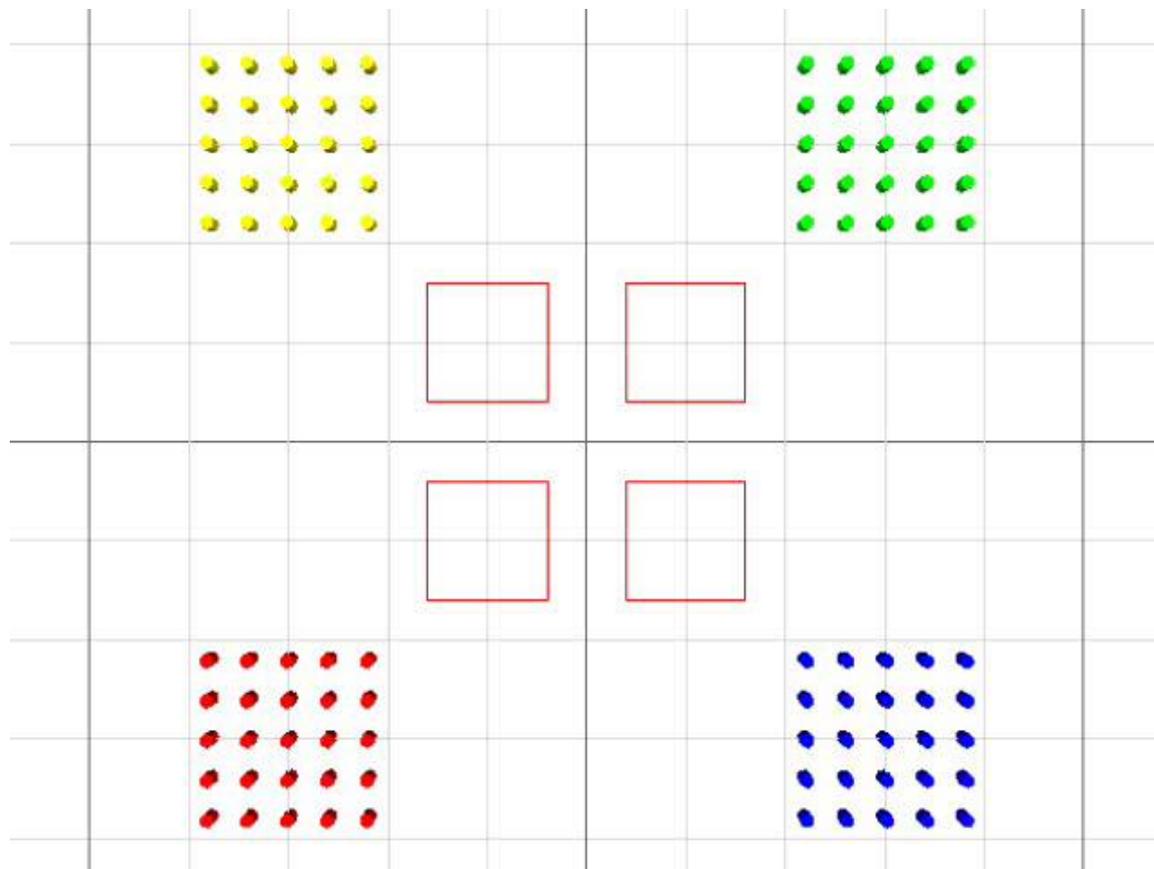
一个园形的例子: 每个智能体(agent)走到园的正对面位置

# 实验



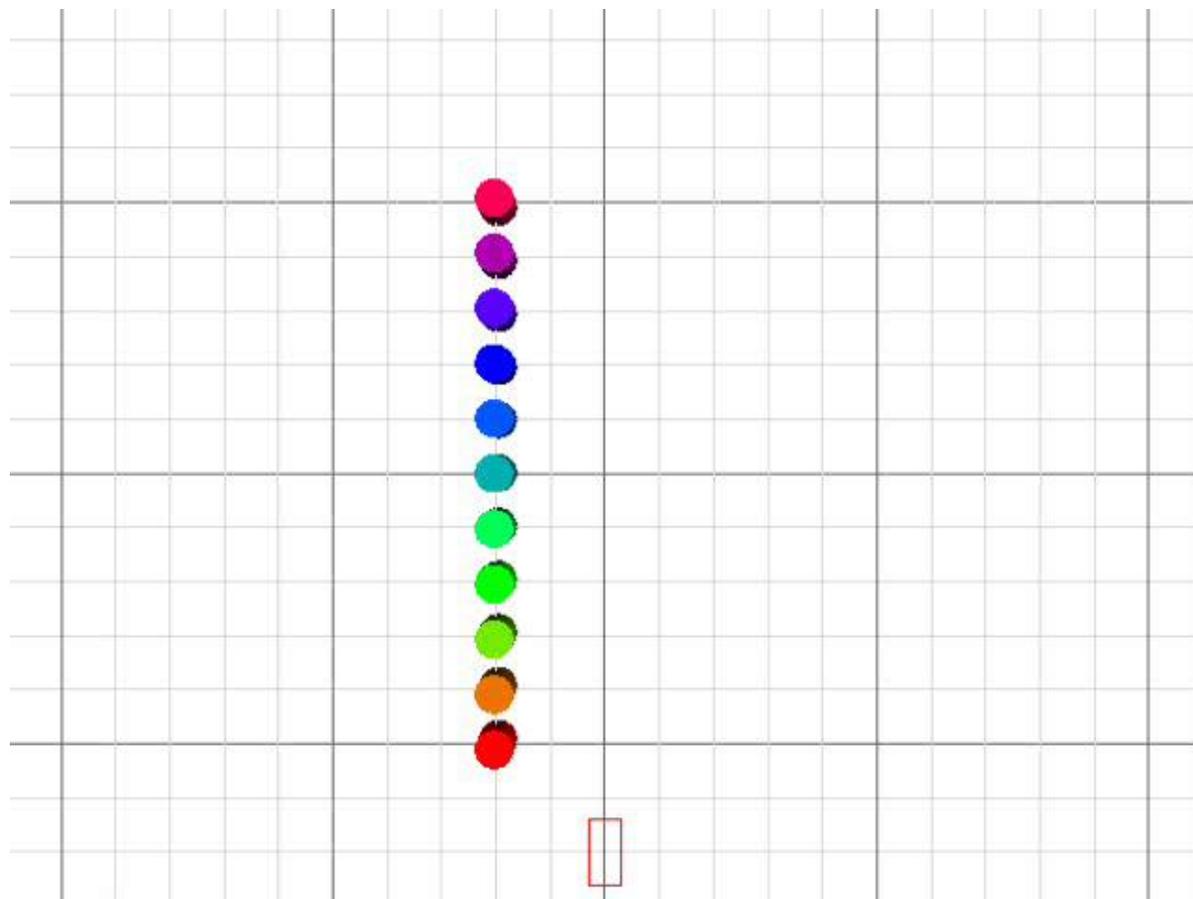
圆形例子实验

# 实验



静态障碍物: 窄的通道

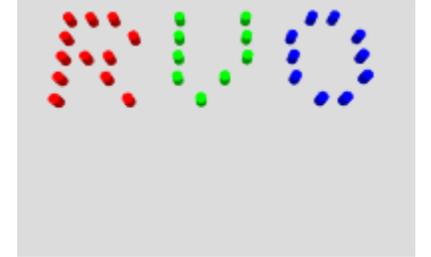
# 实验



高速运动障碍物: 小汽车

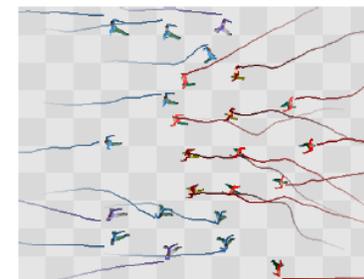
# RVO2 Library: Reciprocal Collision Avoidance for Real-Time Multi-Agent Simulation

- 免费下载网页: <http://gamma.cs.unc.edu/RVO2/>
- Latest Versions
  - RVO2 Library and RVO2-3D Library are **open-source software** as of May 4, 2016.
  - RVO2 Library for C++ v2.0.1, released October 26, 2010.
  - RVO2-3D Library for C++ v1.0, released May 13, 2011.
  - RVO2 Library for C# v2.0, released January 10, 2011.
- Introduction
  - Simulations of multiple agents sharing a common workspace have gained increasing attention recently, for various purposes such as crowd simulation, navigating a team of mobile robots, video games, studying natural flocking behavior, traffic engineering, architecture and design, emergency training simulations, etc. We present an algorithm for interactive navigation and planning of large numbers of agents in two-dimensional (crowded) environments. At runtime, each agent senses the environment independently and computes a collision-free motion based on the [optimal reciprocal collision avoidance](#) (ORCA) formalism. Our algorithm ensures that each agent exhibits no **oscillatory** behaviors.
  - *RVO2 Library for C++* is an implementation of our algorithm. It has a simple API for third-party applications. The user specifies static obstacles, agents, and the preferred velocities of the agents. The simulation is performed step-by-step via a simple call to the library. The simulation is fully accessible and manipulable during runtime. The library exploits multiple processors if they are available using [OpenMP](#) for efficient parallelization of the simulation.
  - *RVO2-3D Library for C++* is an implementation of our algorithm for applications in three dimensions without static obstacles. A C# implementation of our algorithm by Slan is available from [rvo-unity.chezslan.fr](http://rvo-unity.chezslan.fr). We are actively developing implementations in C and Java for future release.



# 连续人群(Continuum Crowds)

- 目前我们讨论的大都是**基于智能体**的(“agent-based”)方法。
- 对每个个体，独立计算其运动。
- 这类方法能刻画每个人的独特情况：
  - Visibility
  - Proximity of other pedestrians
  - Other local factors
- 每个个体可定义不同的模拟参数。



## 缺点:

- 难以一致地生成连续的运动
- 对每个个体进行全局的路径规划代价较大：
  - 大部分模型把局部碰撞检测与全局的路径规划分开处理。
  - 会导致冲突（全局vs局部）。

# 连续人群(Continuum Crowds)



- Treuille等人提出了基于连续体动力学的实时人群模型。在该模型中，**动态势能场**同时将全局导航与移动障碍物（例如其他人）整合在一起，**无需明确碰撞避免**即可有效地解决大规模人群的运动。仿真可以以**交互速率**运行，在各种条件下平稳流动，并自然展现了在真实人群中观察到的紧急现象。
- Treuille, A., Cooper, S. and Popović, Z., 2006. Continuum crowds. ACM Transactions on Graphics (TOG), 25(3), pp.1160-1168. (Google Citations: **1264**, 2024.10.14 )

# 方法流程

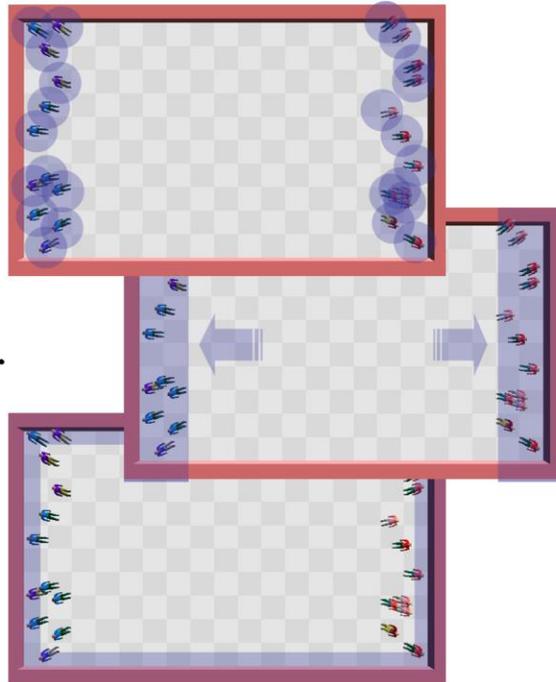
Compute a set of grids representing state.

*Density Grid:* Indicates people's static locations.

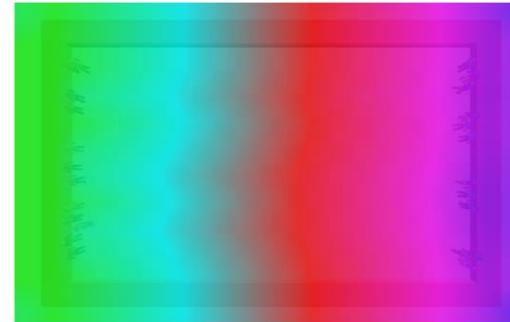
*Goal Grids:* Indicate people's desired locations.

*Boundary Grids:* Indicate impassible grid cells.

*Other grids...*



Combine grids into a set of potential fields.



Use potential fields to update people's positions.



# 方法流程——动态势能场模型

(A dynamic potential field model)

- 最优路径计算
- 密度和速度计算
- 控制方程(The Governing Equations)
  - 最大速度场(Maximum Speed Field)
  - 不舒服场(Discomfort Field)
  - 单位代价场(Unit Cost Field)
- 离散网格结构 ( Discretized grid structure )
  - 密度转换 ( Density conversion )
  - 单位代价计算 ( Unit cost computation )
  - 动态势能场构建(Dynamic Potential Field Construction)

# 控制方程(Governing Equations)

- **假设1:** 每个人尽力到达一个地理目标  $G \subseteq R^2$
- **假设2:** 人群尽可能按最大的速度运动。这可表达为一个最大速度场  $f$ , 使得在  $\mathbf{x}$  位置的一个人在沿  $\theta$  方向运动时, 其速度为:  $\dot{\mathbf{x}} = f(\mathbf{x}, \theta) \mathbf{n}_\theta$ 。其中,  $\mathbf{n}_\theta = (\cos\theta, \sin\theta)^T$  表示指向  $\theta$  方向的单位矢量。
- **假设3:** 存在一个不舒服场  $g$ , 如果  $g(\mathbf{x}') > g(\mathbf{x})$ , 则人们更倾向在  $\mathbf{x}$  而非  $\mathbf{x}'$ 。

# 控制方程(Governing Equations)

- **假设4:** 假设 $\Pi$ 为从 $\mathbf{x}$ 到目标某个点的所有路径的集合。假设速度场 $f$ 、不舒服场 $g$ 、目标 $G$ 固定, 在位置 $\mathbf{x}$ 的人将选择路径 $P \in \Pi$ , 并使得下面的能量极小化:

$$\underbrace{\alpha \int_P 1 ds}_{\text{Path Length}} + \underbrace{\beta \int_P 1 dt}_{\text{Time}} + \underbrace{\gamma \int_P g dt}_{\text{Discomfort}} .$$

其中 $\alpha$ 、 $\beta$ 、 $\gamma$ 为权因子,  $ds$ 表示沿弧长积分,  $dt$ 表示沿时间积分。因为 $ds=f dt$ , 故

$$\alpha \int_P 1 ds + \beta \int_P \frac{1}{f} ds + \gamma \int_P \frac{g}{f} ds \implies \int_P C ds, \quad \text{where } C \equiv \frac{\alpha f + \beta + \gamma g}{f}$$

**单位代价场(Unit cost field)**

# 最优路径计算

## (Optimal Path Computation)

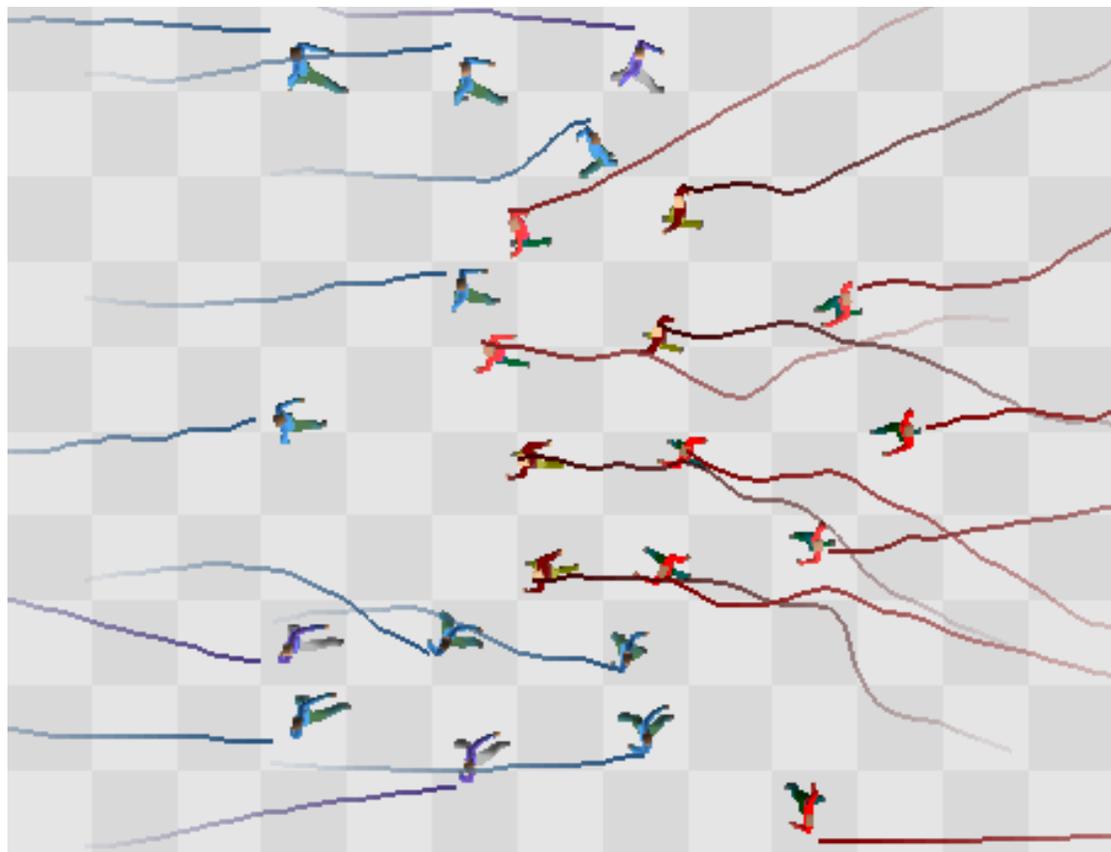
- 给定单位代价场(Unit cost field), 采用势能场方程来求解最优路径
- 假设我们有一个**动态梯度函数** $\phi: \mathbf{R}^2 \rightarrow \mathbf{R}$ , 其值等于到达目标最佳路径的代价。
- 直观地, 对于每个人, 最优策略是沿着这个函数梯度相反的方向运动, 因为这样可以最快地降低路径的代价。
- **动态梯度函数**可采用如下方法来定义:
  - 在目标处:  $\phi=0$ ,
  - 在其它地方满足Eikonal方程,  $\|\nabla\phi(\mathbf{x})\| = C$ ,
- 这表示, 每个人沿着梯度相反的方向运动, 其速度为 $\dot{\mathbf{x}} = f(\mathbf{x}, \theta)\mathbf{n}_\theta$ , 也即:

$$\dot{\mathbf{x}} = -f(\mathbf{x}, \theta) \frac{\nabla\phi(\mathbf{x})}{\|\nabla\phi(\mathbf{x})\|},$$

# 算法流程

- 对于每个时间步:
  - 把crowd转化为密度场;
  - 对于每个group:
    - 构造单位代价场  $C$ ;
    - 构造势能场  $\phi$  及其梯度  $\nabla\phi$
    - 更新每个人的位置
  - 强制保证人之间的最短距离

# 结果



连续人群(蓝色)与Reynold智能体人群互动(红色)



一支大部队实时穿过山区地形去追逐一支较小的部队

# DEMO

---

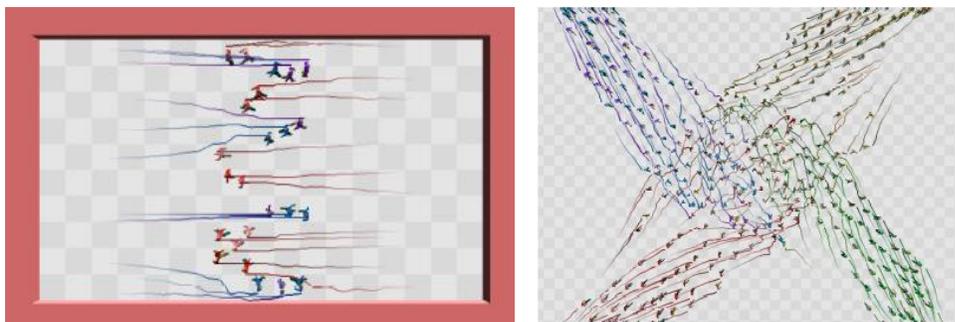
## Continuum Crowds

Adrien Treuille  
Seth Cooper  
Zoran Popović

# 优缺点

## 优点

- 可消除个体的运动冲突现象
- 运动路径更光滑
- 可方便地整合到基于智能体的方法中
- 可刻画一些常见的人群现象，如
  - 人行道的形成
  - 湍流堵塞期间的短时涡旋



## 缺点

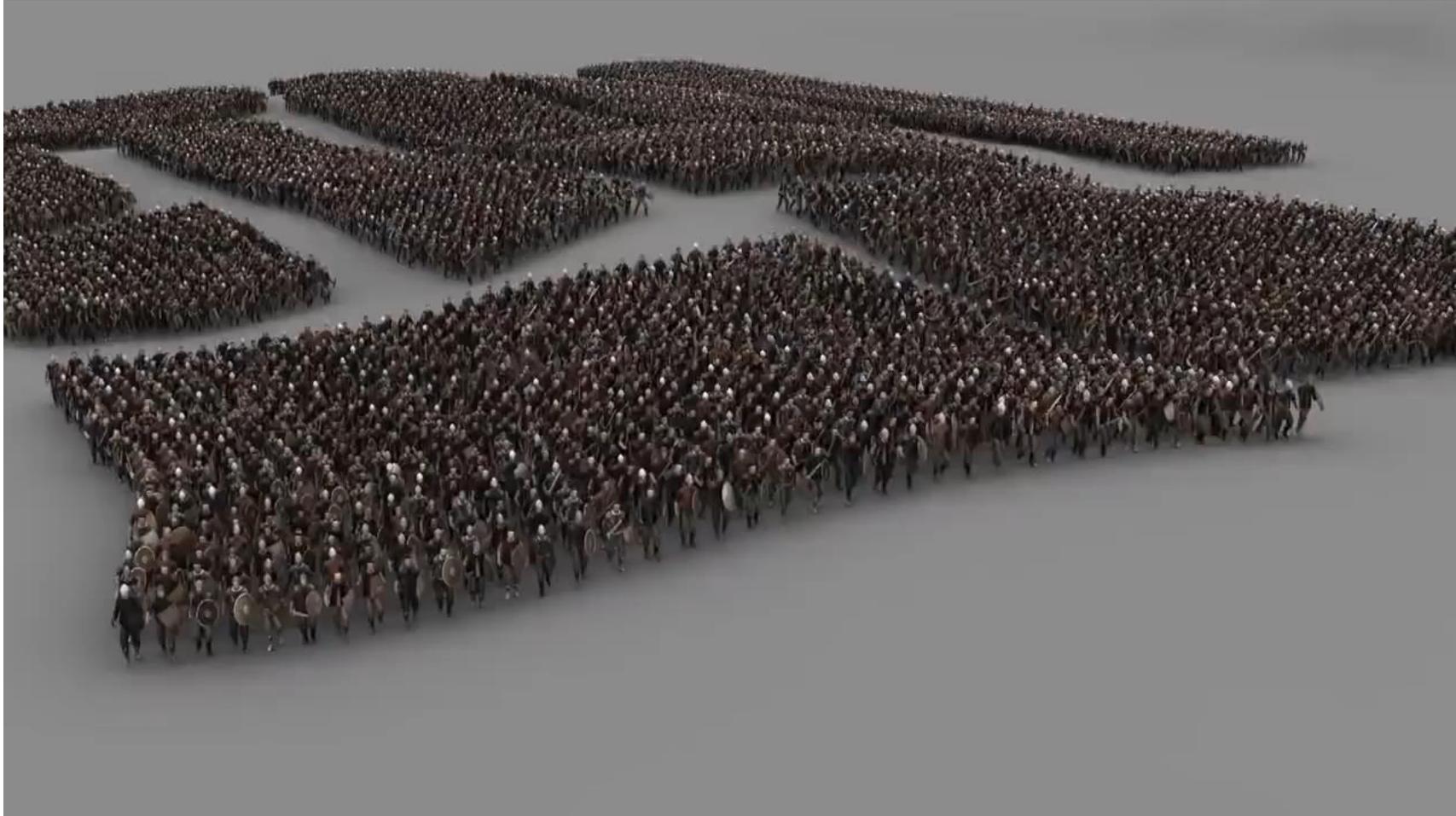
- 不适用于未知的环境
  - 该方法假设人群知道动态的真实环境
- 在不考虑惯性的情况下改变方向
- 与完全基于Agent的方法相比，缺少灵活性和个体差异性

# 总结

- 人群仿真是一个比较活跃的研究方向。人群的运动涉及环境约束、人之间的交互、智能路径规划等。
- 在影视特效、交通仿真、人群紧急疏散、**无人驾驶**等场景具有较大应用。
- 每种方法都有其适用范围。
- 要模拟真实人群的行为还是具有很大的挑战性。

Alahi, Alexandre, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. "Social Istm: Human trajectory prediction in crowded spaces." *CVPR 2016*, pp. 961-971.

# Massive DEMO



**The End**