

# Компьютерное моделирование

## Агентное моделирование

Кафедра ИВТ и ПМ

2018

# План

## Прошлые темы

Агентное моделирование

Системы с множеством агентов

Реализация модели

Моделирование поведения мурьвьёв

Моделирование движения бактерии

## Программы для моделирования

# Outline

## Прошлые темы

Агентное моделирование

Системы с множеством агентов

Реализация модели

Моделирование поведения мурьвёв

Моделирование движения бактерии

## Программы для моделирования

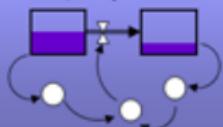
## Прошлые темы

- ▶ Виды моделирования времени?
- ▶ Виды моделирования пространства?
- ▶ Если пространственное положение не важно?
- ▶ Как определяется цепь Маркова?
- ▶ Для чего может использоваться цепь Маркова?

# Имитационное моделирование

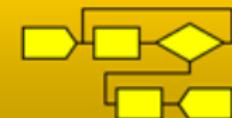
## Системная динамика

Связанные переменные,  
Накопители, Обратные связи



## Дискретно-событийное

Заявки, Ресурсы, Процессы  
(последовательности операций)



Индивидуальные свойства  
и правила поведения.  
Прямое или косвенное  
взаимодействие



## Агентное моделирование

# Outline

Прошлые темы

## Агентное моделирование

Системы с множеством агентов

Реализация модели

Моделирование поведения мурьвёв

Моделирование движения бактерии

Программы для моделирования

## Агентное моделирование vs моделирование случайных событий

- ▶ Моделирование дискретных событий сосредоточено на процессах
- ▶ Агентное моделирование рассматривает отдельные части системы - агентов.

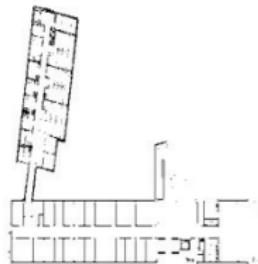
# Агентное моделирование

**Агентное моделирование** (agent-based model ,ABM) — метод имитационного моделирования, исследующий поведение децентрализованных агентов и то, как такое поведение определяет поведение всей системы в целом.

# Агентное моделирование

- ▶ Моделирование движения пешеходов, автомобилей
- ▶ Поведение колонии муравьёв, других насекомых, бактерий
- ▶ моделирование групп животных, в том числе в условиях конкуренции и оценка их влияния на окружающую среду
- ▶ моделирование потребительского поведения
- ▶ моделирование рынка ценных бумаг (моделирование действий трейдеров)
- ▶ Моделирование движения массовки в GCI
- ▶ Моделирование ботов в играх
- ▶ ...

# Агентное моделирование



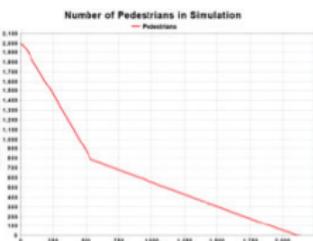
(a)



(b)

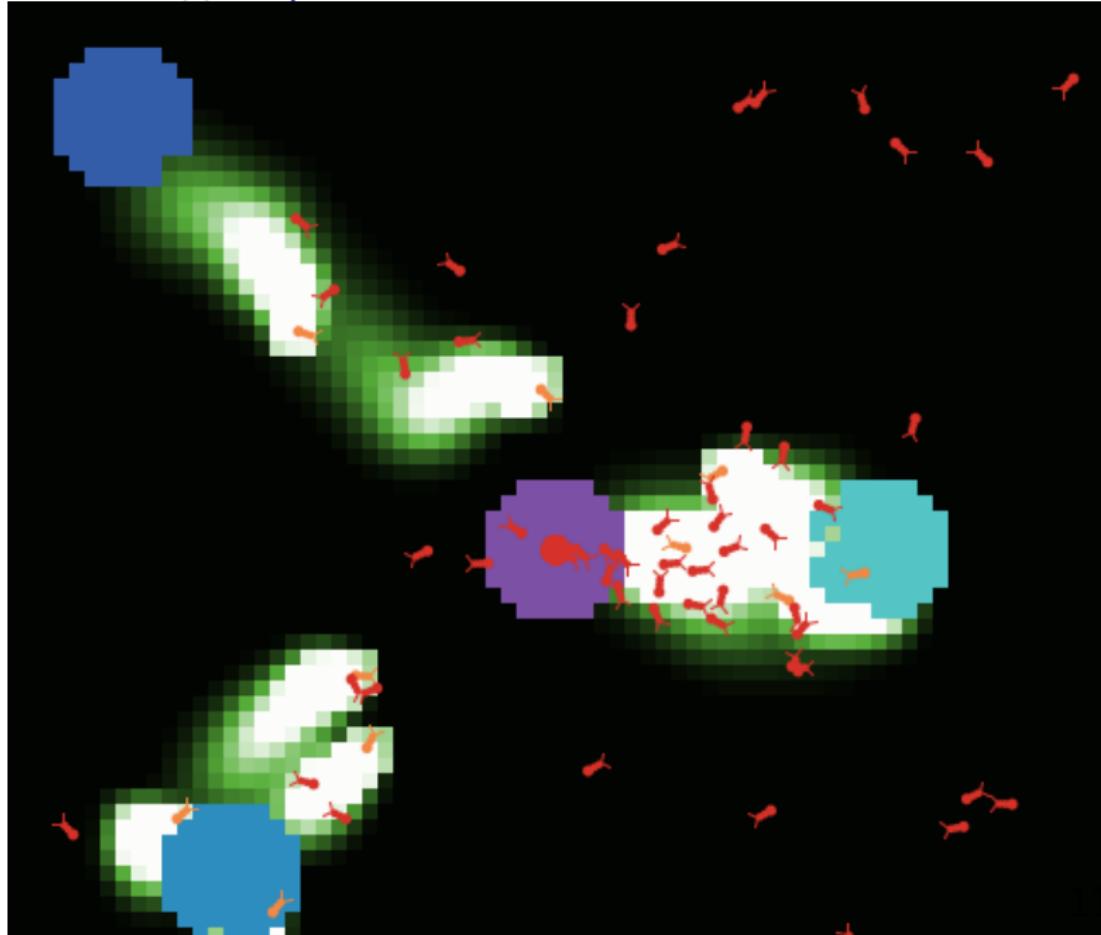


(c)

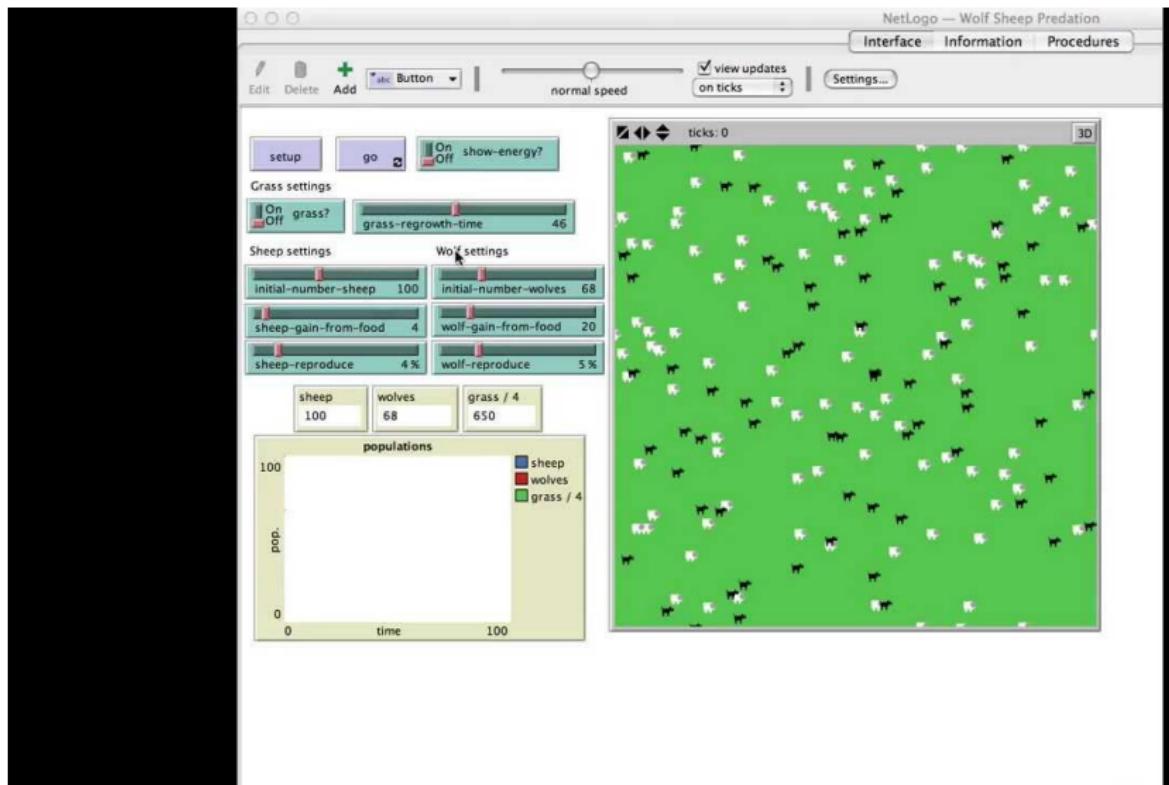


(d)

## Агентное моделирование



# Агентное моделирование



# Агентное моделирование



cinefex

## Агентное моделирование



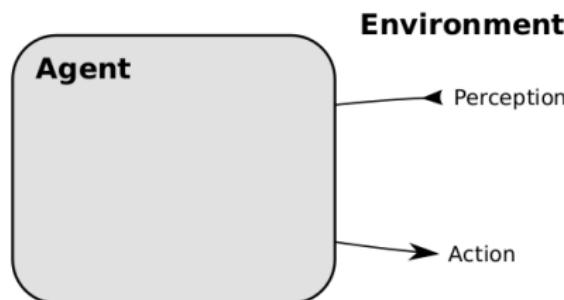
## Агентное моделирование

Сложное поведение системы можно описать моделируя отдельные её части - агентов (в общем случае разнородных), поведение которых гораздо проще моделировать.

Относительная простота не в последнюю очередь достигается за счёт инкапсуляции и принципа сокрытия.

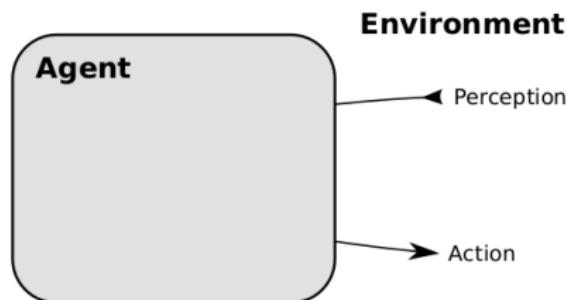
# Агент

- ▶ Агент автономен
- ▶ Агент может взаимодействовать с окружающей средой
- ▶ С точки зрения среды агент это чёрный ящик



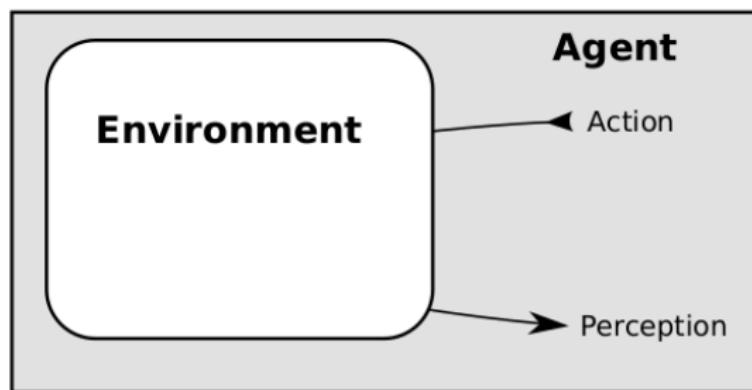
# Агент

- ▶ Обладает некоторой информацией о среде (perception)
- ▶ но не обязательно всей. иногда информация может быть верной с некоторой вероятностью
- ▶ Может воздействовать (action) на среду или на других агентов



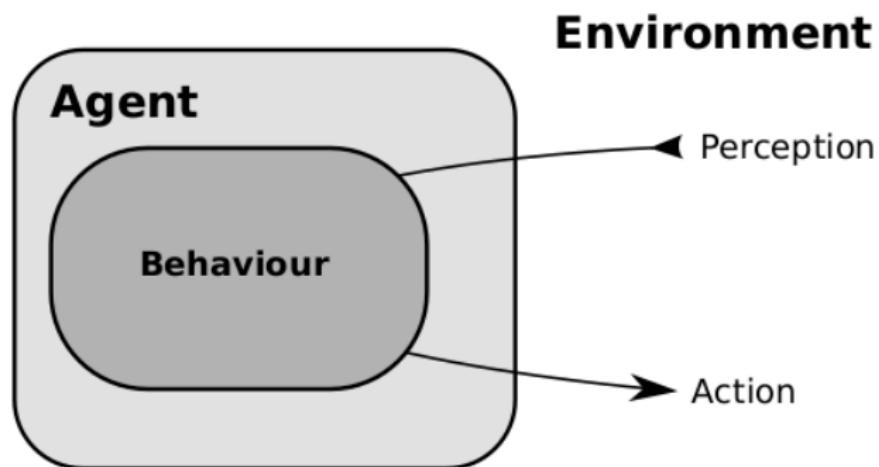
## Агент

- ▶ С точки зрения среды агент - черный ящик



# Простой агент

Simple Reflex Agent



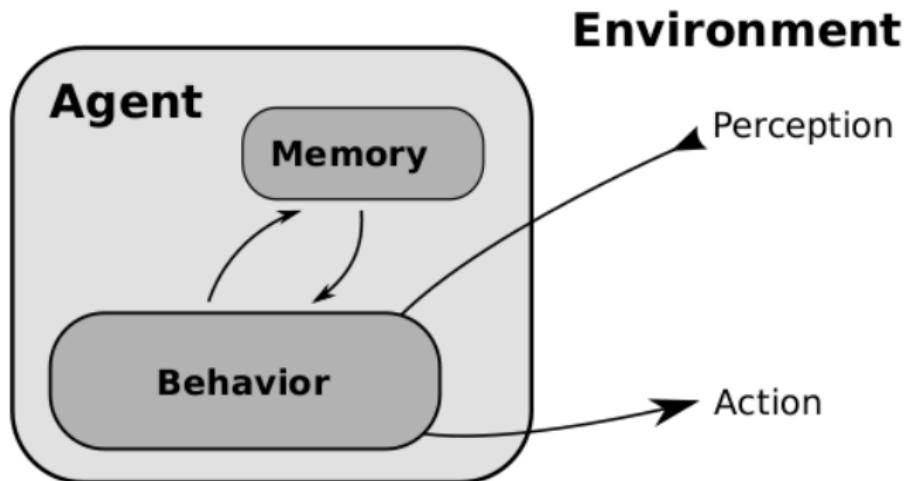
## Простой агент

- ▶ Поведение обычно строится на основе набора правил
- ▶ Можно рассматривать как функцию:

$$PERCEPTION \rightarrow ACTION$$

- ▶ Правила могут быть вероятностными

## Агент с памятью



## Агент с памятью

- ▶ Имеет состояние (память)
- ▶ Поведение зависит от состояния
- ▶ Способен "обучаться"
- ▶ Можно рассматривать как функцию:

$$PERCEPTION \times STATE \rightarrow ACTION \times STATE$$

## Пример. Трейдер

```
def behavior( price, state ):
    lastTxPrice, cash, stocks = state
    if (price > lastTxPrice) and (price - lastTxPrice ) > RL:
        n = floor( stocks * Cs )
        return SELL( n, price )
    elif (price < lastTxPrice) and (lastTxPrice - price) > RL:
        n = floor( ( cash * Cb ) / price )
        return BUY( n, price )
    else:
        return NOP
```

# Outline

Прошлые темы

Агентное моделирование

Системы с множеством агентов

Реализация модели

Моделирование поведения мурьвёв

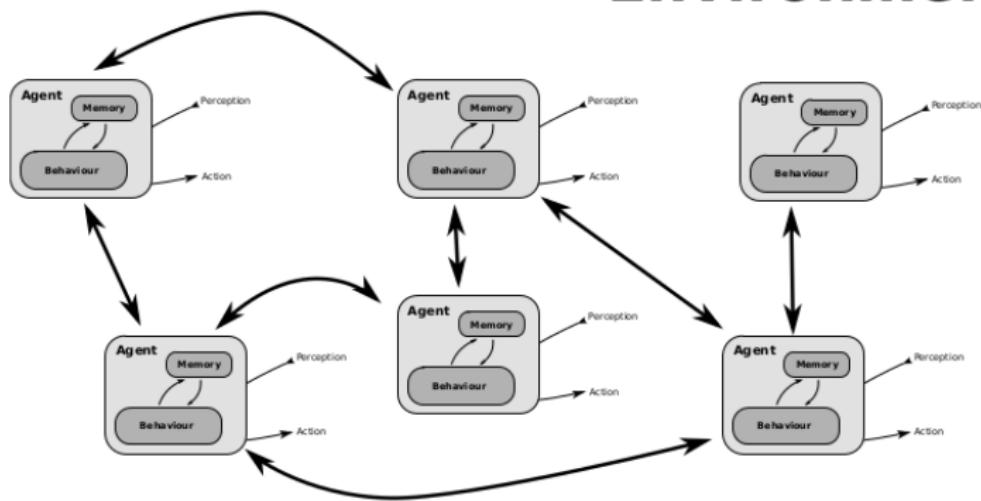
Моделирование движения бактерии

Программы для моделирования

## Системы с множеством агентов

- ▶ агенты могут быть одного вида или нескольких
- ▶ агенты могут взаимодействовать друг с другом напрямую

## Environment



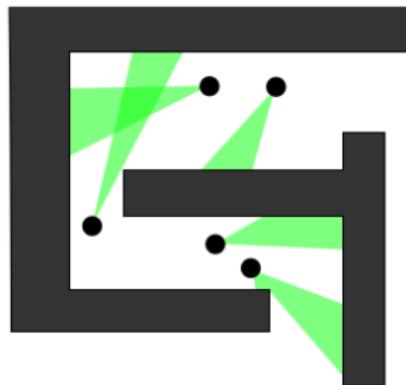
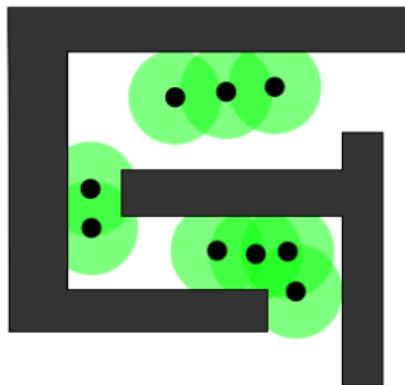
## Агенты и пространство

В некоторых моделях положение агентов в пространстве не играет роли. Например, если агенты могут взаимодействовать вне зависимости от расстояния.

В некоторых моделях пространство нужно включить в модель

- ▶ Агенты могут иметь своё расположение в пространстве (двумерном, трёхмерном и т.д.)
- ▶ Агенты могут перемещаться
- ▶ Поведение агентов может зависеть от их положения в пространстве
- ▶ Связи между агентами так же могут зависеть от их взаимного расположения и способа восприятия окружающей среды

# Агенты и пространство



Какие из агентов связаны?

# Outline

Прошлые темы

**Агентное моделирование**

Системы с множеством агентов

**Реализация модели**

Моделирование поведения мурьвёв

Моделирование движения бактерии

Программы для моделирования

## Рализация

- ▶ класс - тип агента
- ▶ объект - агент
- ▶ поля - состояние агента
- ▶ методы - поведение

```
class FooAgent {  
    private final long ID = 223;  
    private int count = 2;  
    private double ratio = 1.5;  
  
    public Action behaviour( Perception p ) {  
        //...  
    }  
}
```

## Асинхронный алгоритм

```
agents, env = initialize()
t = t_init
while t < t_max:
    for agent in agents:
        p = computePerceptionFor(agent, env, agents)
        action = agent.behaviour(p)
        updateEnvironment( env, action )
        increment(t)
```

## Синхронный алгоритм

```
agents, env = initialize()
t = t_init
while t < t_max:
    ps = computeAllPerceptions(env, agents)
    actions = allBehaviours( agents, ps )
    updateEnvironment( env, actions )
    increment(t)
```

# Синхронный алгоритм

```
agents, env = initialize()
t = t_init
while t < t_max:
    ps = computeAllPerceptions(env, agents)
    actions = allBehaviours( agents, ps )
    updateEnvironment( env, actions )
    increment(t)
```

В чём отличие от асинхронного?

## Пространство. Подход Лагранжа

- ▶ агентам известно о своём положении в пространстве

```
agents = [  
    Agent(id=1, posX=8.2, posY=0.5, ... ),  
    Agent(id=2, posX=9.1, posY=2.7, ... ),  
    Agent(id=3, posX=4.6, posY=1.8, ... ),  
    ...  
]
```

# Пространство. Подход Эйлера

- ▶ Пространство разбито на ячейки
- ▶ Каждая ячейка может содержать несколько агентов

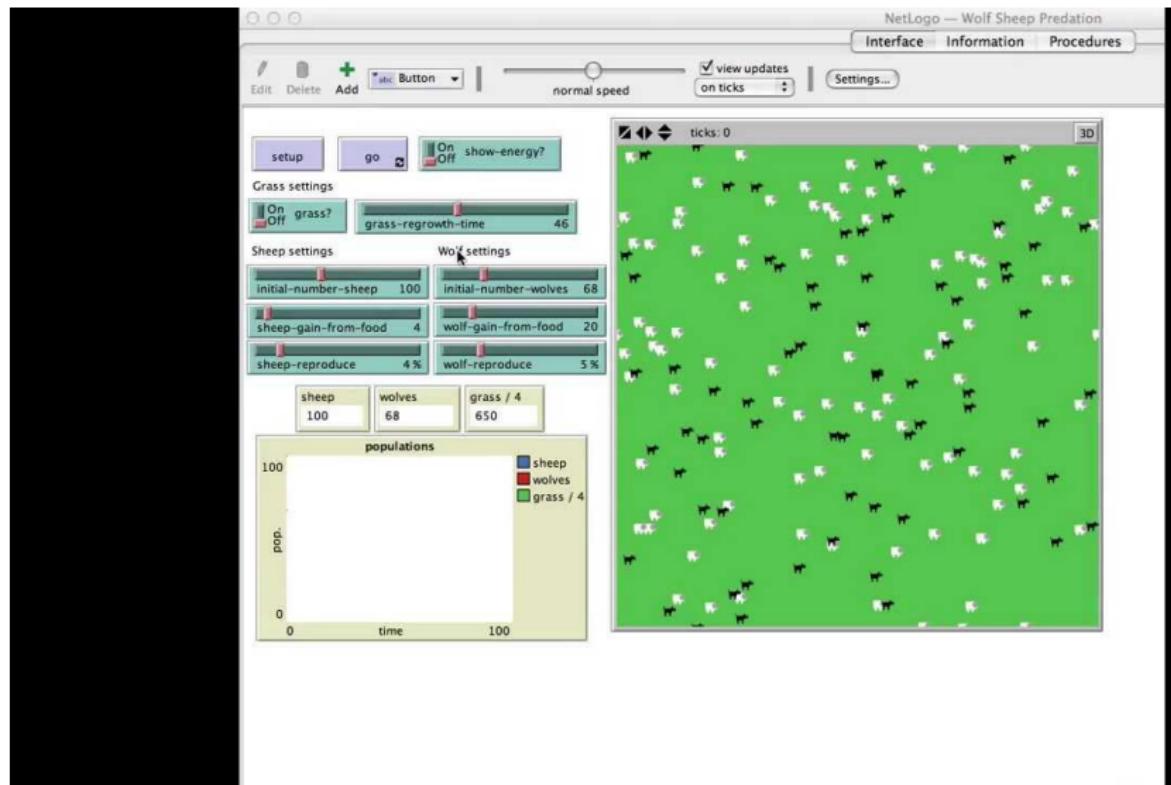
## Преимущества

- ▶ Легко определить взаимодействующих агентов (они находятся в соседних ячейках)
- ▶ возможно параллельное вычисление

## Недостатки

- ▶ Меньшая точность расположения в пространстве

# Подход Эйлера



# Outline

Прошлые темы

**Агентное моделирование**

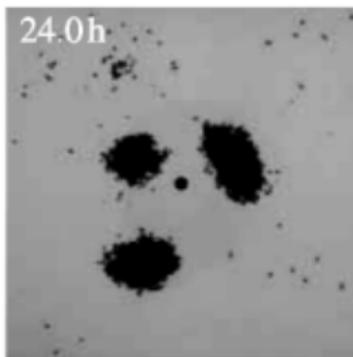
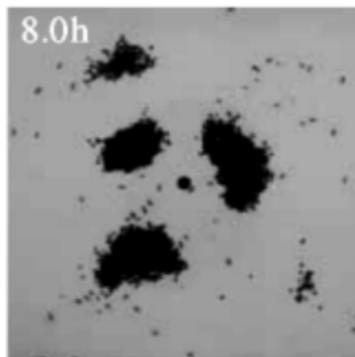
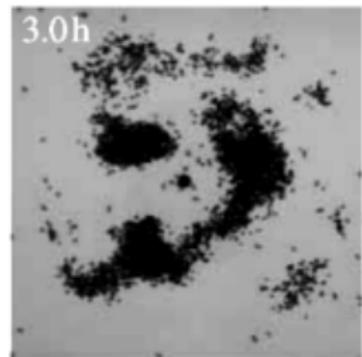
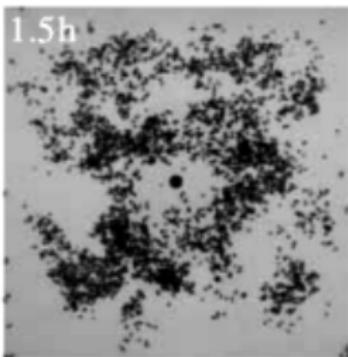
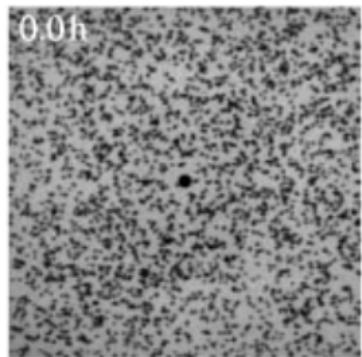
Системы с множеством агентов

Реализация модели

**Моделирование поведения мурьвёв**

Моделирование движения бактерии

Программы для моделирования



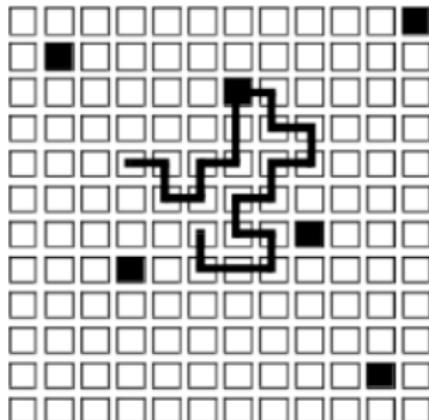
Сбор муравьями трупов в кучи

Jost et al., J. R. Soc. Interface, 2007

- ▶ Как это происходит?
- ▶ Это разум роя (swarm intelligence)?
- ▶ Какая простейшая модель, повторяет такое поведение?

## Deneubourg's Model (1991)

- ▶ пространство - равномерная сетка из квадратных ячеек
- ▶ муравей может двигаться в 4 стороны
- ▶ движения муравья - случайное блуждание
- ▶ может переползать труп
- ▶ асинхронный алгоритм обработки агентов



## Поведение муравья

- ▶ с вероятностью  $P_p$  муравей поднимает труп, если он находится в маленькой кучке или один
- ▶ с вероятностью  $P_d$  муравей оставляет труп в большой куче
- ▶ Память муравья  $M$ :
  - ▶ муравей помнит  $n$  последних посещённых клеток: был ли в клетке труп или нет
  - ▶  $M(i) = 1$ , для клетки с трупом
  - ▶  $M(i) = 0$ , для пустой клетки
  - ▶  $i = 1..n$
  - ▶ массив  $M$  обновляется каждый шаг муравья (каждый шаг времени)

## Поведение муравья

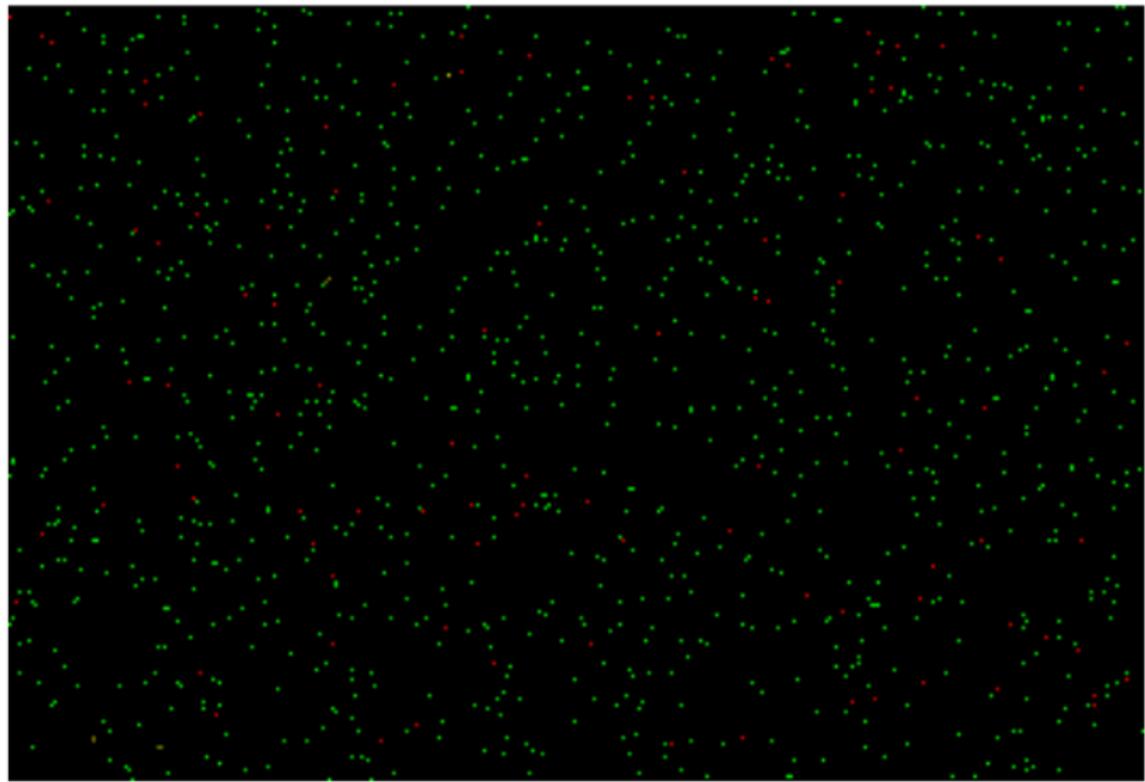
$$f = \sum_{i=0}^n M(i)$$

$$P_d = \left( \frac{k_1}{k_1 + f} \right)^2$$

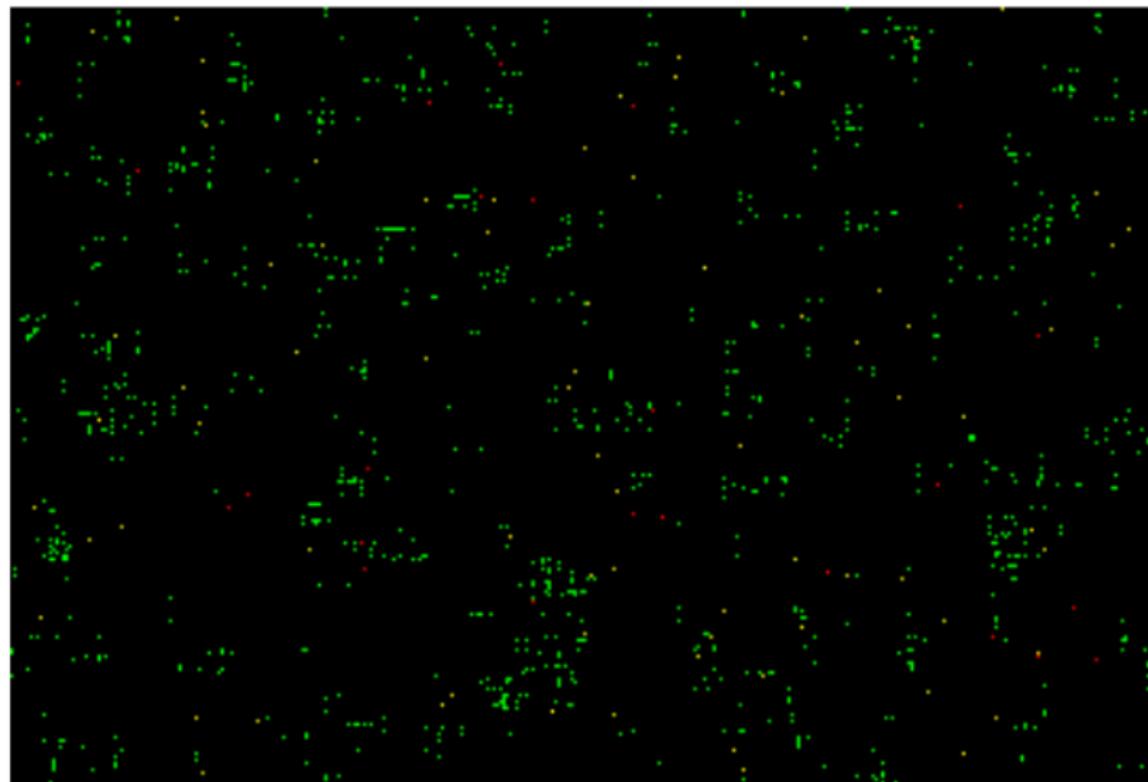
$$P_p = \left( \frac{f}{k_2 + f} \right)^2$$

$k_1$  и  $k_2$  - параметры модели

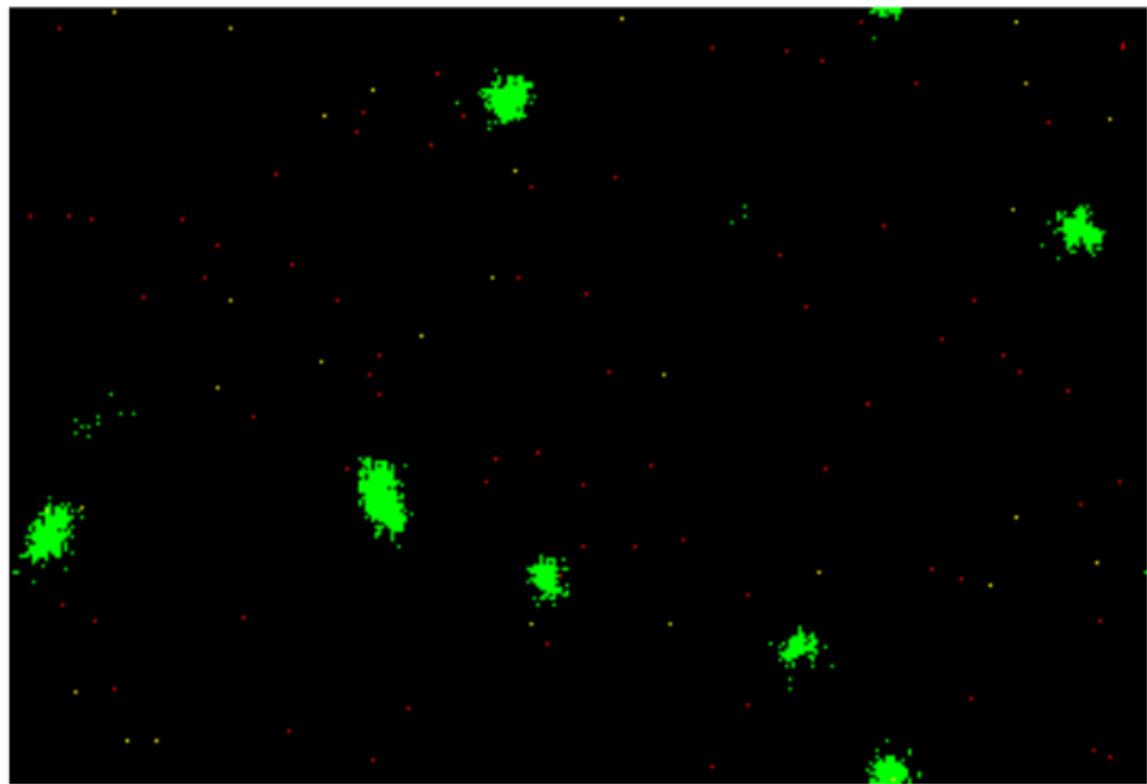
## Deneubourg's Model. Пример. I



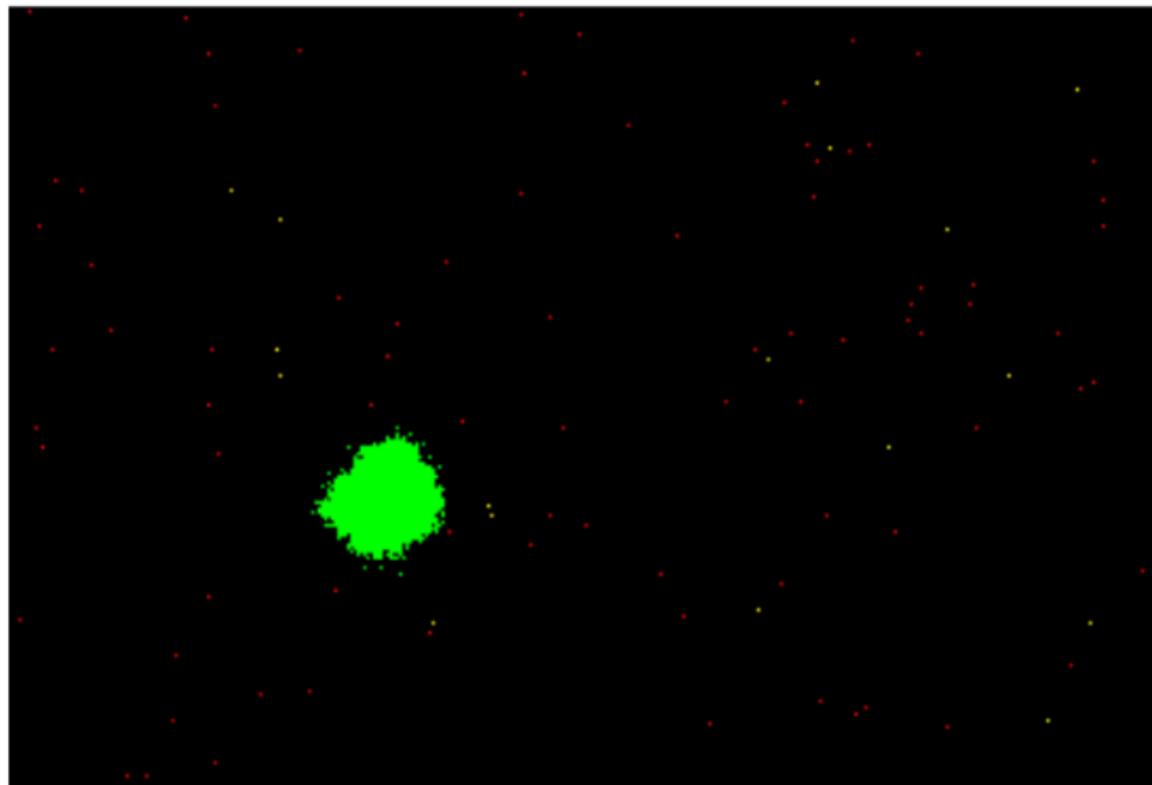
## Deneubourg's Model. Пример. II



## Deneubourg's Model. Пример. III



## Deneubourg's Model. Пример. IV



## Deneubourg's Model. Выводы

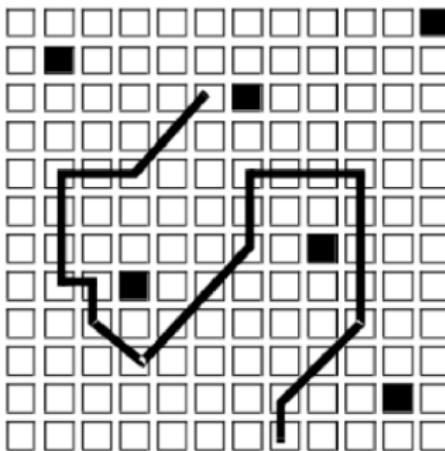
- ▶ Модель проста
- ▶ Модель соответствует наблюдаемому поведению муравьёв
- ▶ Но требует от муравья хорошей памяти...

# Unige Model (2000)

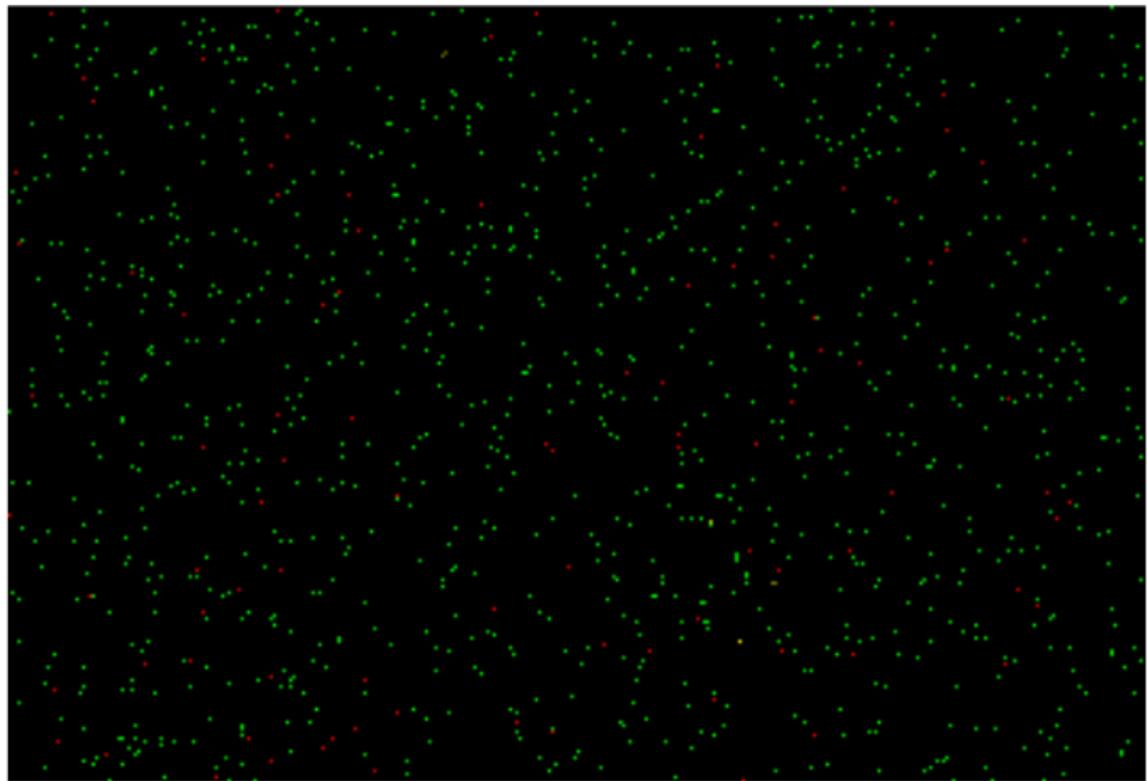
- ▶ пространство - набор квадратных ячеек
- ▶ муравей может двигаться в 8-ми направлениях
- ▶ движения муравья - случайное блуждание по большой территории (за счёт большего числа возможных направлений)
- ▶ асинхронный алгоритм обработки агентов

## Поведение муравья

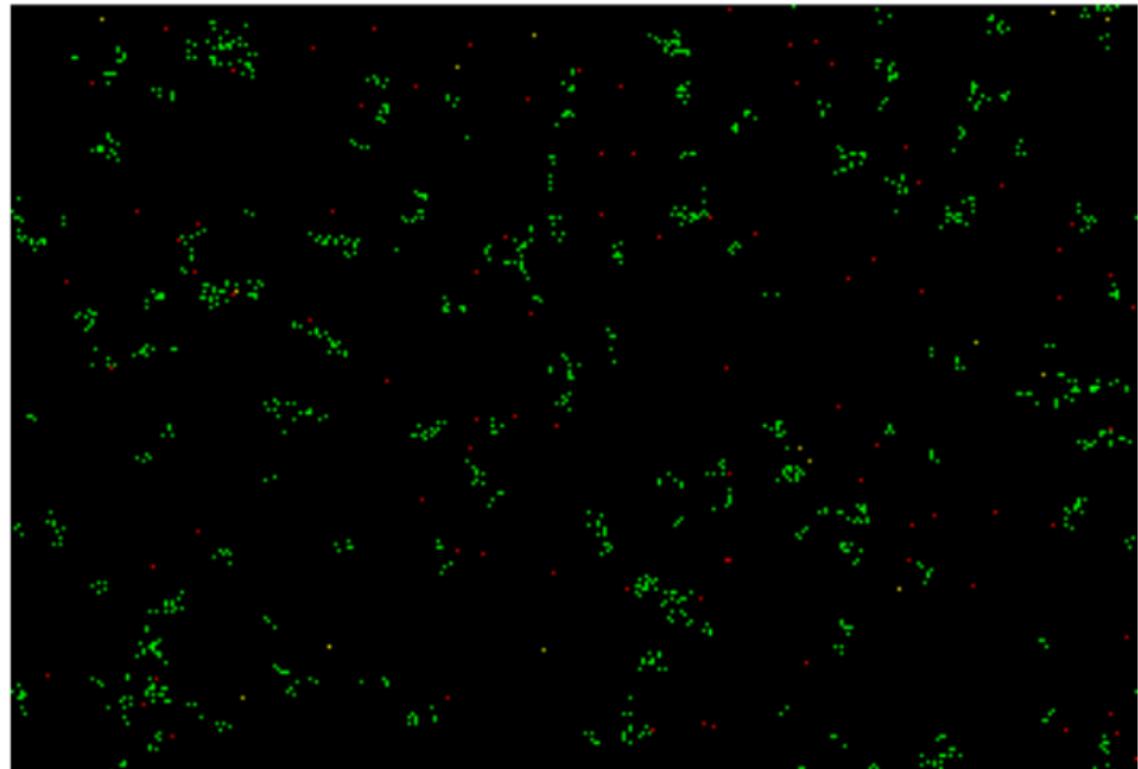
- ▶ Муравьи обходят препятствия (других муравьёв, трупы и др.)
- ▶ Муравей всегда подбирает труп если ничего не несёт
- ▶ Муравей всегда бросает труп, когда видит другой труп



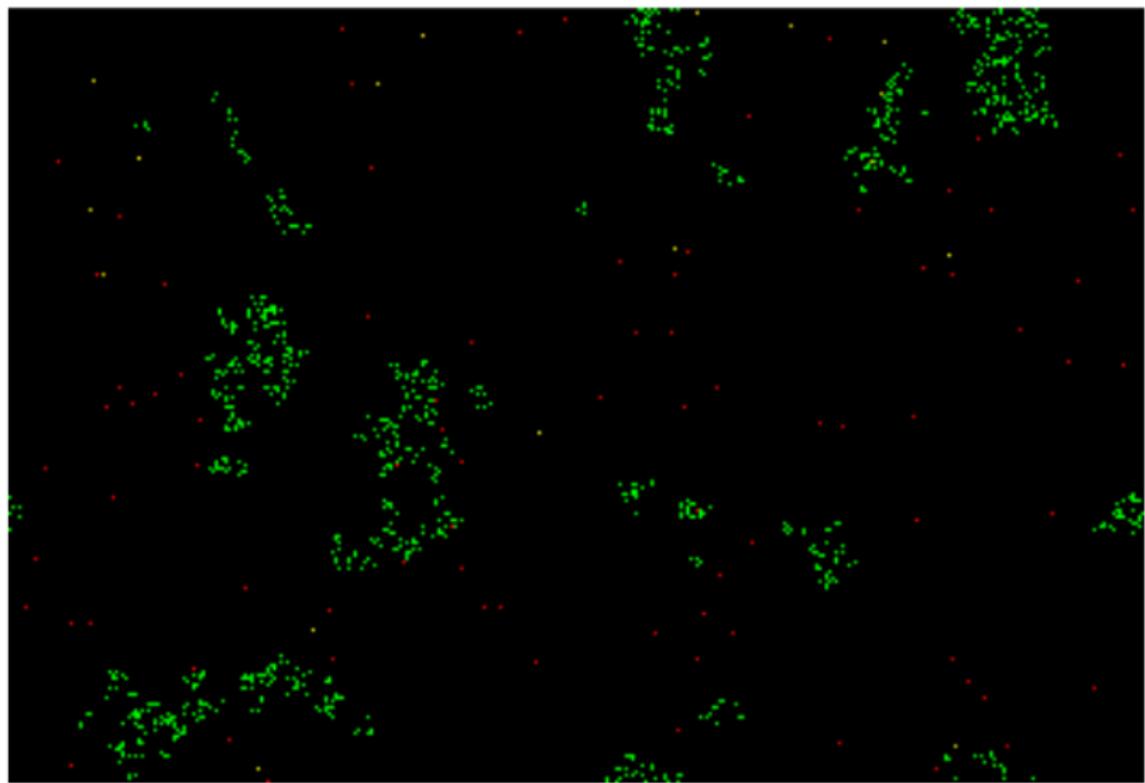
## Пример 2. I



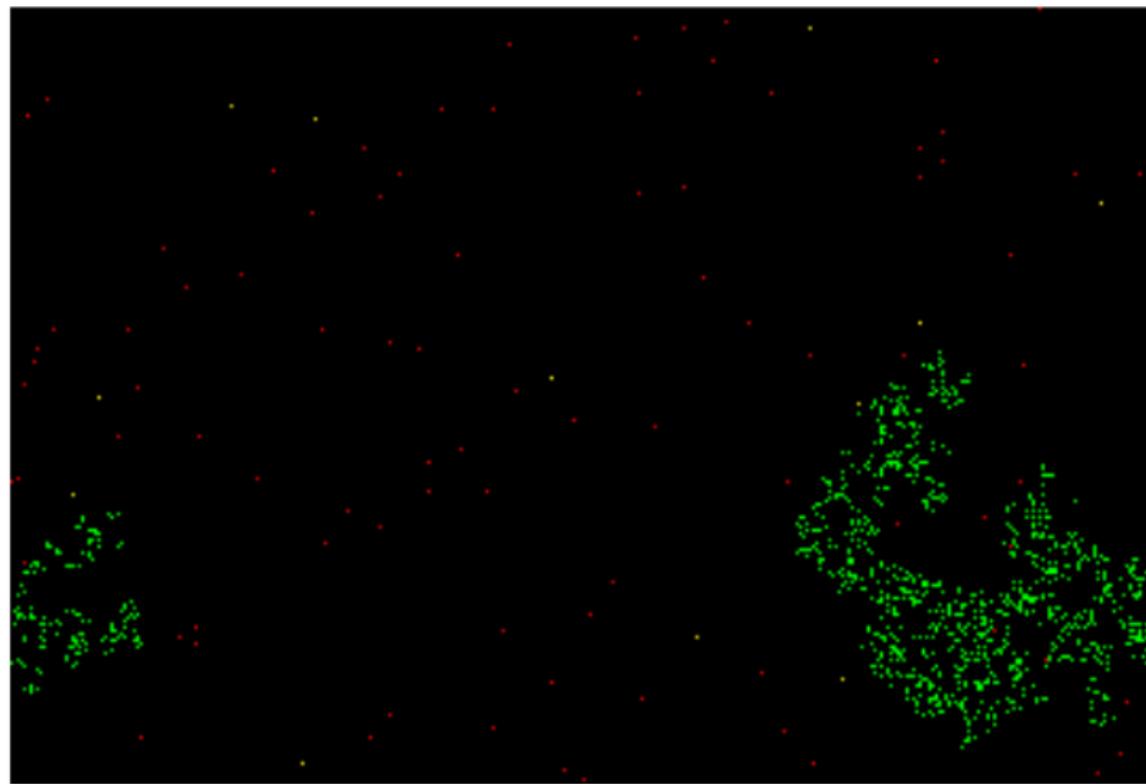
## Пример 2. II



## Пример 2. III



## Пример 2. IV



- ▶ Вероятность добавления или удаления трупа из кучи одинакова
- ▶ Для муравья нет разницы, большая перед ним куча или нет

- ▶ Вероятность добавления или удаления трупа из кучи одинакова
- ▶ Для муравья нет разницы, большая перед ним куча или нет

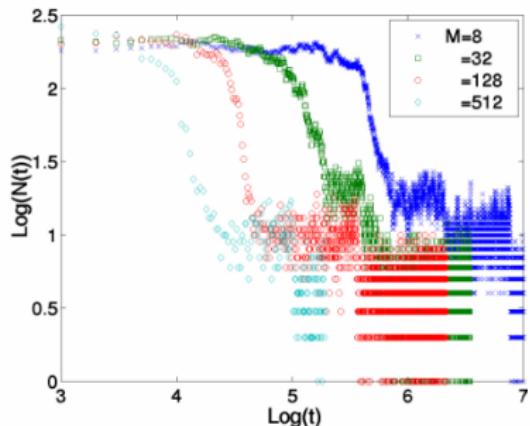
Почему образуются большие кучи?

- ▶ Вероятность добавления или удаления трупа из кучи одинакова
- ▶ Для муравья нет разницы, большая перед ним куча или нет

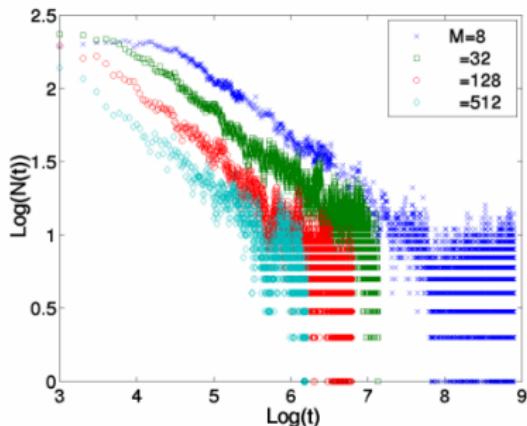
Почему образуются большие кучи?

- ▶ Если муравей подобрал одиноко лежащей труп, то вокруг него уже не возникнет куча
- ▶ Муравьи оставляют трупы только вокруг других трупов

# Сравнение моделей



Deneubourg, with 8 directions

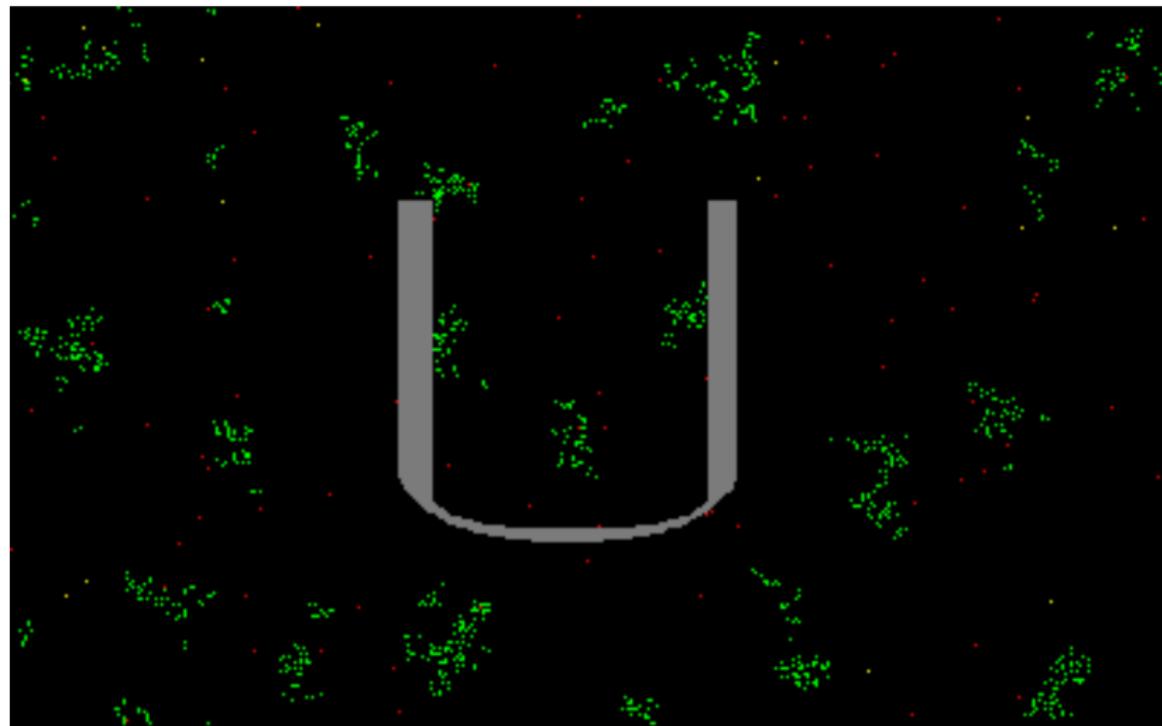


Unige

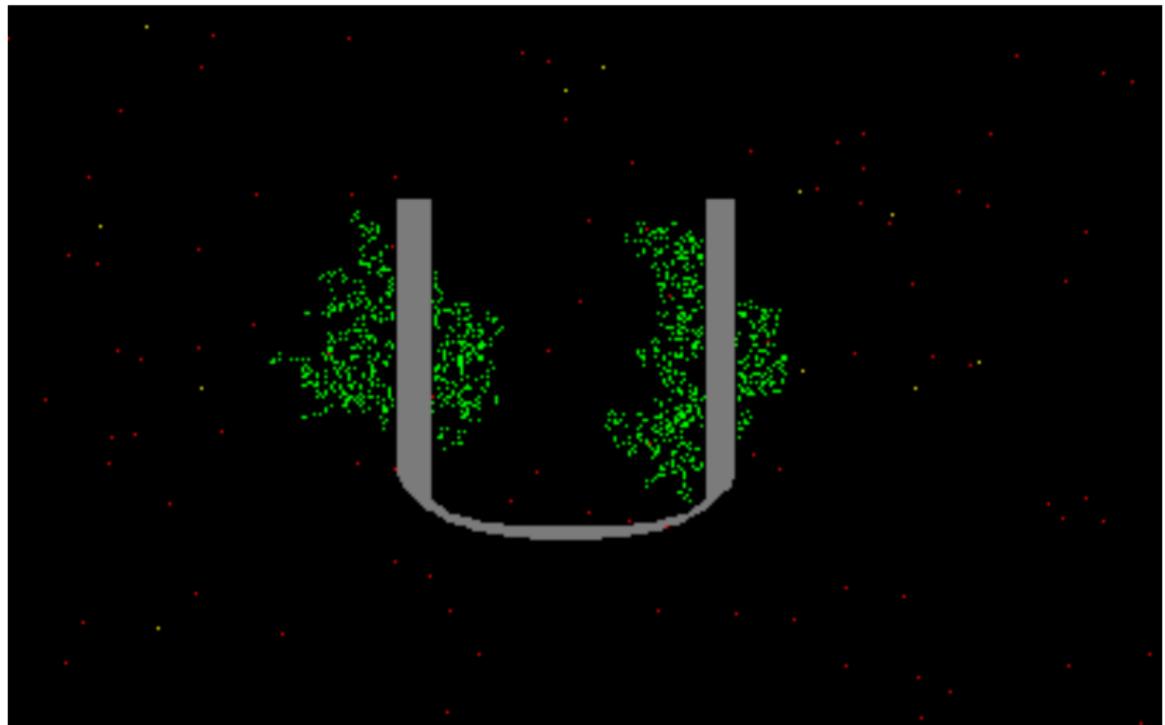
$N$  - число куч

- ▶ В модели Deneubourg время за которое достигается равновесие меньше примерно в 10 раз.
- ▶ Поведение муравьёв не коллективное
- ▶ Один муравей способен выполнить ту же работу, только за большее время
- ▶ Создание больших куч - статистический эффект
- ▶ "Умные" муравьи делают работу быстрее

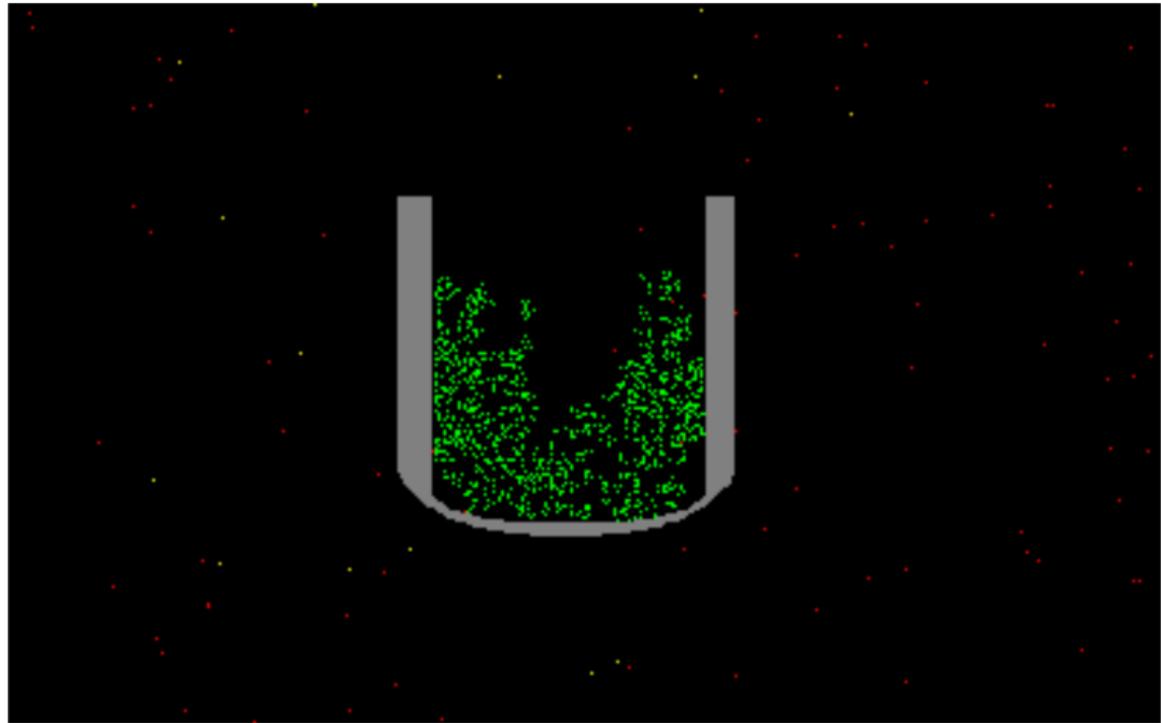
## Пример 3. Поле с препятствиями. I



## Пример 3. Поле с препятствиями. II



### Пример 3. Поле с препятствиями. III



# Outline

Прошлые темы

## Агентное моделирование

Системы с множеством агентов

Реализация модели

Моделирование поведения мурьвёв

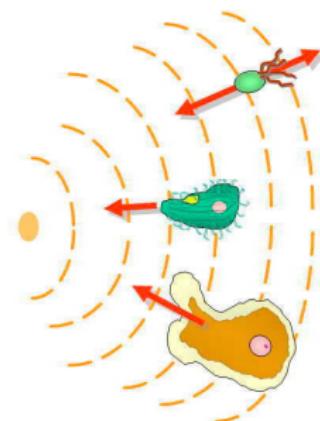
Моделирование движения бактерии

Программы для моделирования

# Хемотаксис

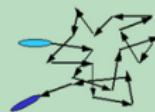
Хемотаксис — двигательная реакция микроорганизмов на химический раздражитель.

Бактерии способны двигаться по направлению к аттрактантам (зачастую питательным веществам) и от репеллентов (например, токсинов)

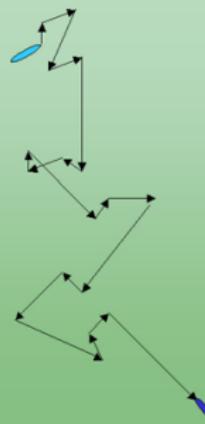


# Хемотаксис

Bacterial movement



**Random walk**  
(No stimulus)

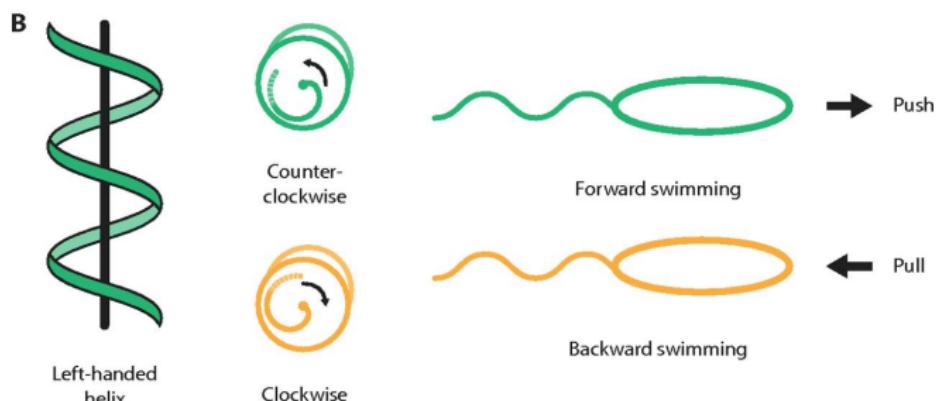
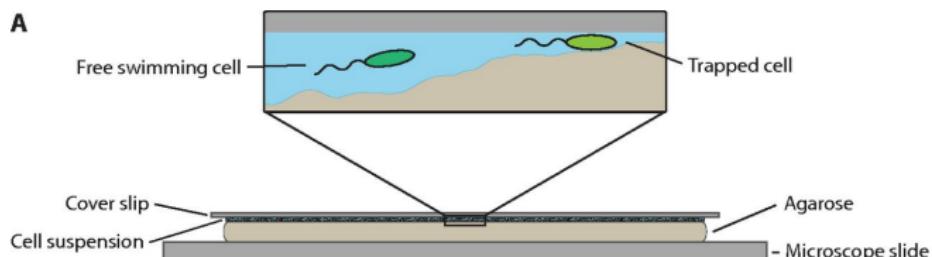


**Chemotaxis**  
(Positive Stimulus)

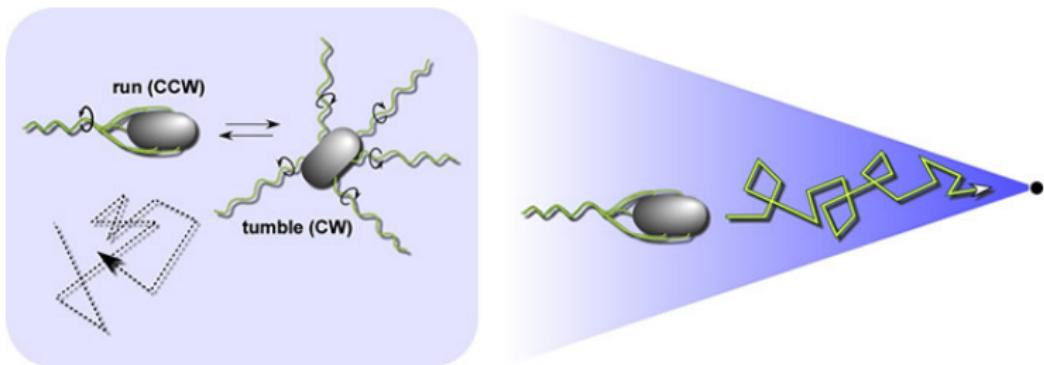
Brudersohn  
Wikimedia Commons

# Движение

Бактерии способны двигаться за счёт вращения микроскопических нитей - филаментов.



# Движение



## Хемотаксис

- ▶ Бактерии за счёт своих размеров и строения не могут измерить градиент концентрации вещества прямо на месте.
- ▶ Бактерия способна запомнить концентрацию вещества которая была некоторое время назад и сравнить с текущей концентрацией.
- ▶ Движение - случайное, но с учётом изменения концентрации вещества.

# Модель

- ▶ Моделирование пространство согласно подходу Эйлера.
- ▶ Пространство двумерная сетка
- ▶ в каждой клетке  $(x,y)$ 
  - ▶ список бактерий
  - ▶ концентрация вещества  $\rho_{xy}$
- ▶ Состояние  $(d_i, m_i)$  бактерии:
  - ▶  $d_i$  - последнее направление (N, S, E, W)
  - ▶  $m_i$  - последняя концентрация вещества

## Поведение

- ▶ Бактерии известно последняя концентрация  $m$ ;
- ▶ Бактерия измеряет концентрацию  $\rho_{x,y}$  в текущей позиции  $(x, y)$

Параметры модели:

- ▶  $p_i$  - вероятность поворота если концентрация увеличилась
- ▶  $p_d$  - вероятность поворота если концентрация уменьшилась

$$p_d > p_i$$

## Функция поведения

```
def behaviour( rho, m_i, d_i ):
    if rho <= m_i:
        p = p_d
    else:
        p = p_i
    if random() <= p:
        return rho, randomDirection()
    else:
        return rho, d_i
```

# Среда

Диффузия рассматриваемого вещества:

$$\frac{\partial \rho_{x,y}(t)}{\partial t} = D \nabla^2 \rho_{x,y}(t)$$

$D$  - коэффициент диффузии

$\nabla = \frac{\partial}{\partial x} \vec{i} + \frac{\partial}{\partial y} \vec{i} + \frac{\partial}{\partial z} \vec{i}$  - оператор набла

# Outline

Прошлые темы

Агентное моделирование

Системы с множеством агентов

Реализация модели

Моделирование поведения мурьвёв

Моделирование движения бактерии

Программы для моделирования

# Программы для моделирования

- ▶ AnyLogic

Используется во многих отраслях и исследователями.

Бесплатная версия ограничивает время моделирования.

Кроссплатформенная.

- ▶ NetLogo

Широко используется в образовании. Используется исследователями. Есть веб версия. Кроссплатформенная

- ▶ StarLogo

Использованы материалы курса Simulation and modeling of  
natural processes  
[coursera.org/learn/modeling-simulation-natural-processes/](https://coursera.org/learn/modeling-simulation-natural-processes/)

# Ссылки

Материалы курса

[github.com/ivtipm/computer-simulation](https://github.com/ivtipm/computer-simulation)