

# Навигация в синтетических пространствах

Студент гр. 5057/2  
Руцкий Владимир

23.10.2010

# Задача навигации

- Дано:
  1. Рабочее пространство  $W$
  2. Интеллектуальный агент (или несколько)
  3. Начальная  $A$  и конечная  $B$  точки из  $W$
- Требуется:

Найти путь для агента из  $A$  в  $B$

Сложность решения и разработка эффективного алгоритма зависят от:

- характеристик рабочего пространства,
- характеристик интеллектуальных агентов,
- требований к пути

# Требования к пути

- Найти кратчайший путь
- Найти путь близкий к кратчайшему
- Найти (кратчайший) путь, проходящий через определённый набор точек
- Также возможно:
  - Найти объект обладающий определёнными свойствами в пространстве - точка В не дана, её надо найти
  - Найти путь для движения набора агентов в формации (движение отряда)

# Рабочее пространство

- Размерность и топология пространства
  - 2D, 3D, d3D
- Взвешенность
  - Различная стоимость прохода через различные участки
- Постоянность пространства в процессе решения задачи:
  - статическое пространство
  - динамическое пространство
    - движущиеся препятствия

# Интеллектуальные агенты

- Сведения агента о пространстве
  - всё пространство
  - локальная область
- Количество агентов
- Коммуникация агентов
  - Разделение между агентами информации о пространстве
- Ограничения памяти агента
  - Может ли агент хранить информацию о всём исследованном пространстве
- Число степеней свободы агента
  - Может ли агент поворачиваться или менять свою форму

# Дополнительные ограничения

- Геометрические
  - Агент имеет определённую геометрическую форму
- Кинематические
  - Поворота агента занимает время
  - Невозможность поворота в некоторых случаях
    - Например, автомобиль может поворачиваться только двигаясь

# Сложность задачи

- В большинстве реальных постановок - NP-полные или NP-трудные
  - 2D, полигональные препятствия - полиномиальная сложность
  - 3D, препятствия - тетраэдры - NP-трудная

# Упрощение задач навигации

- Сведение к меньшим размерностям
  - 3D -> 2D / d3D
- Снятие геометрических ограничений
  - агент - материальная точка



# Характеристики рассматриваемых задач

Рассмотрим:

- 2D
- Полигональные препятствия
- Один агент
- Цель: найти близкий к кратчайшему путь из A в B

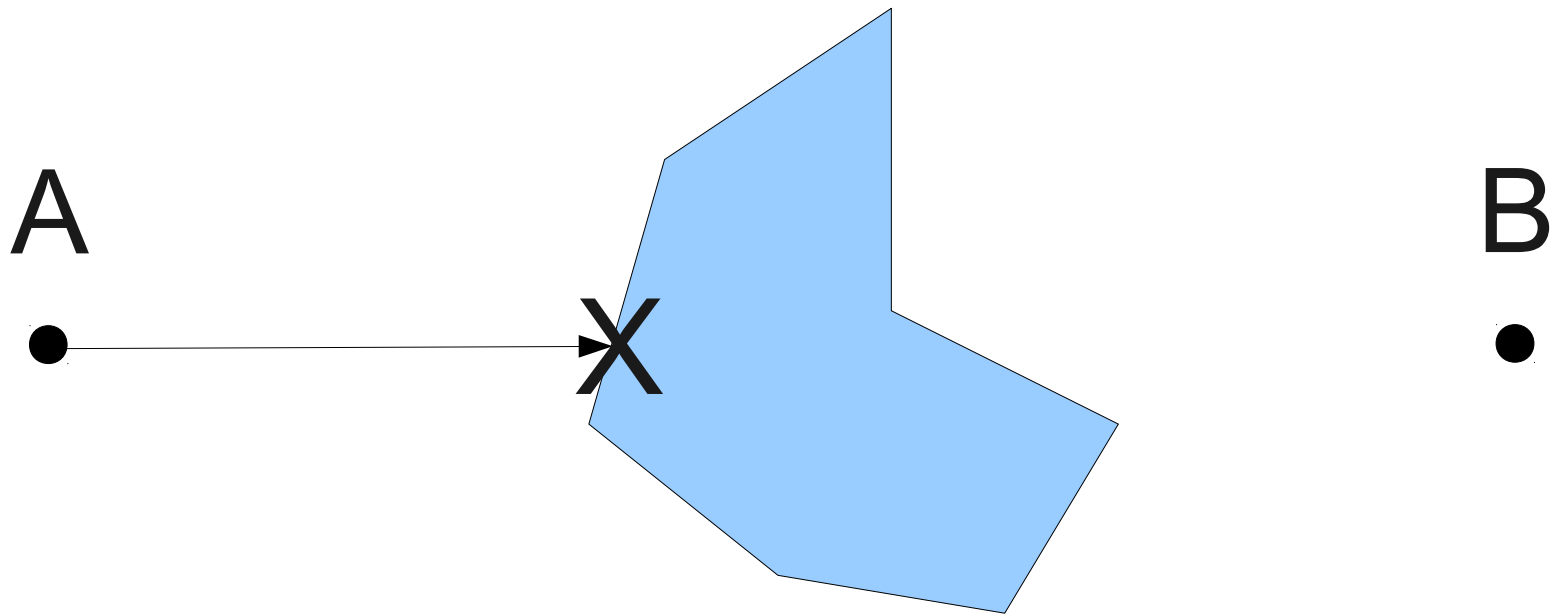
# Решение задачи.

## Методы, ориентированные на агента

- Агент не обладает знаниями о всём пространстве
- Агент «видит» небольшую окрестность рядом с собой
  - Тактильная информация о пространстве - движение до столкновения

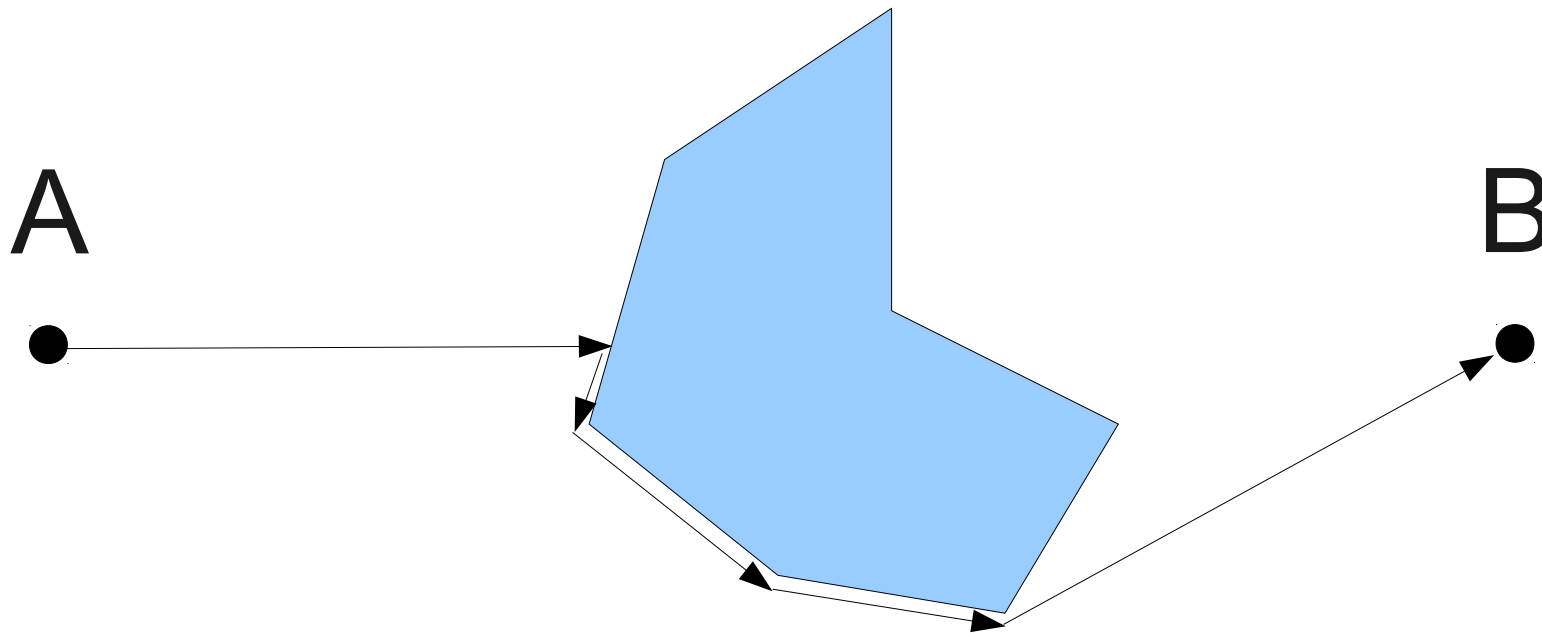
# GoStraight

- Работает в отсутствие препятствий
- Используется как процедура в более сложных алгоритмах



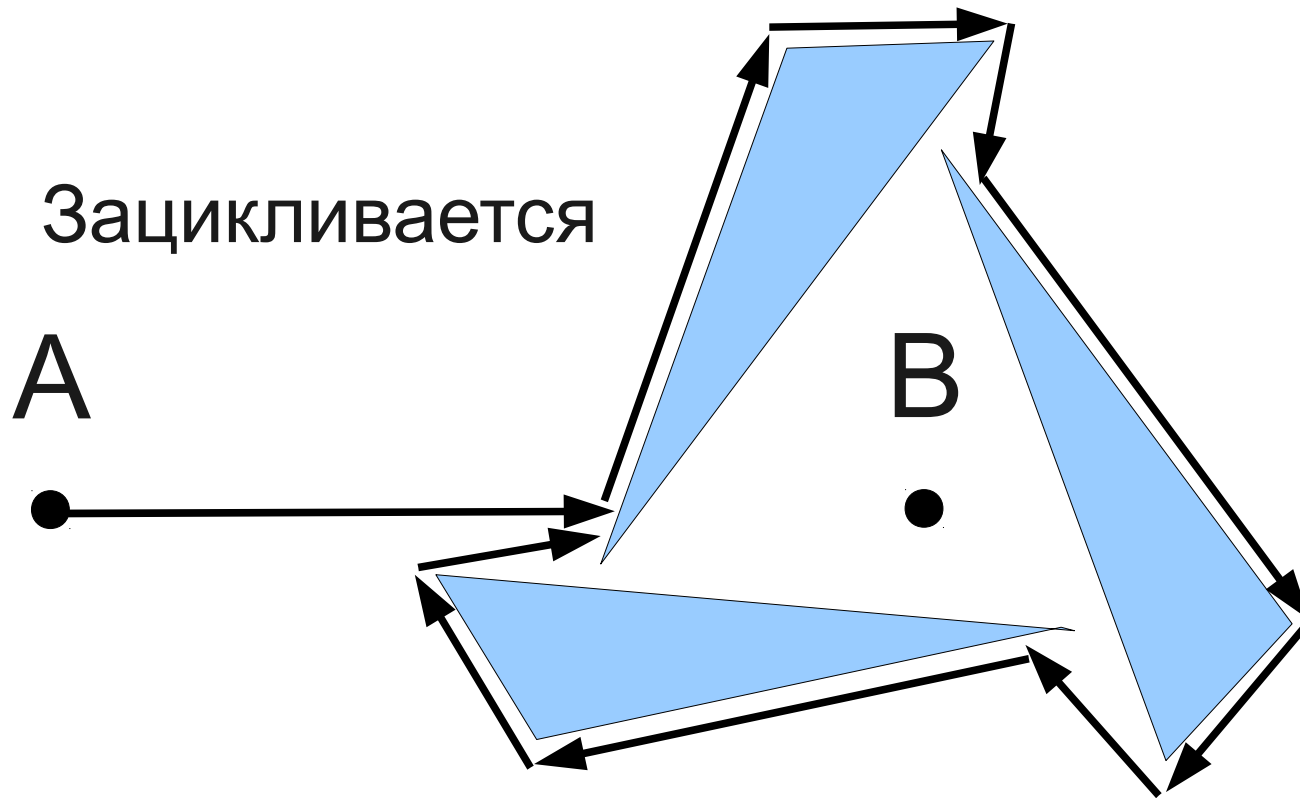
# LHT/RHT (1)

- Left (Right) Hand Traverse - обход препятствия по левой (правой) руке:
  - Идём к цели до встречи с препятствием
  - Обходим препятствие с правой стороны



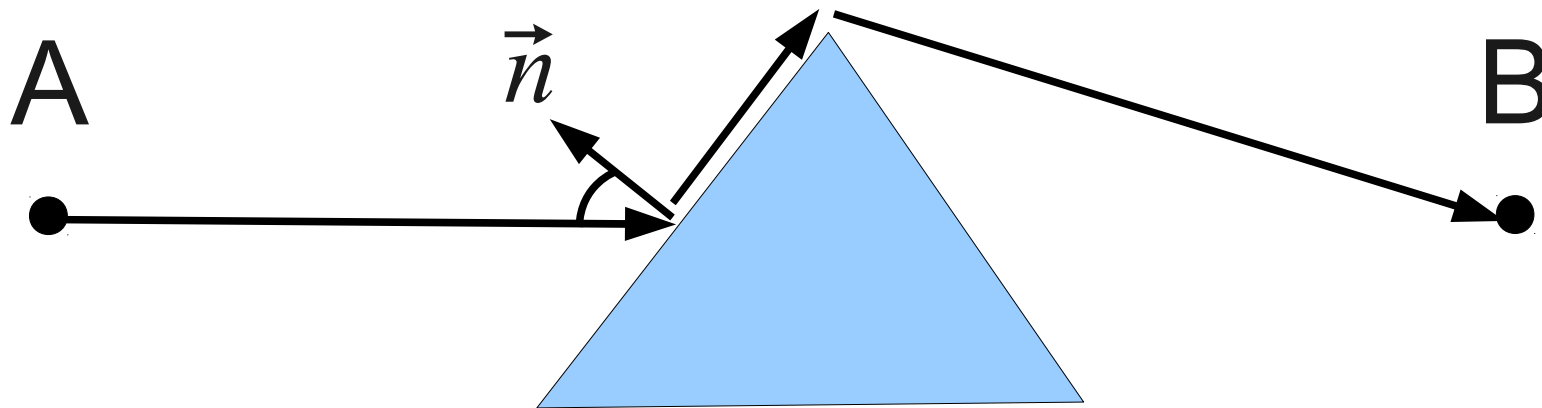
## LHT/RHT (2)

- Не всегда работает



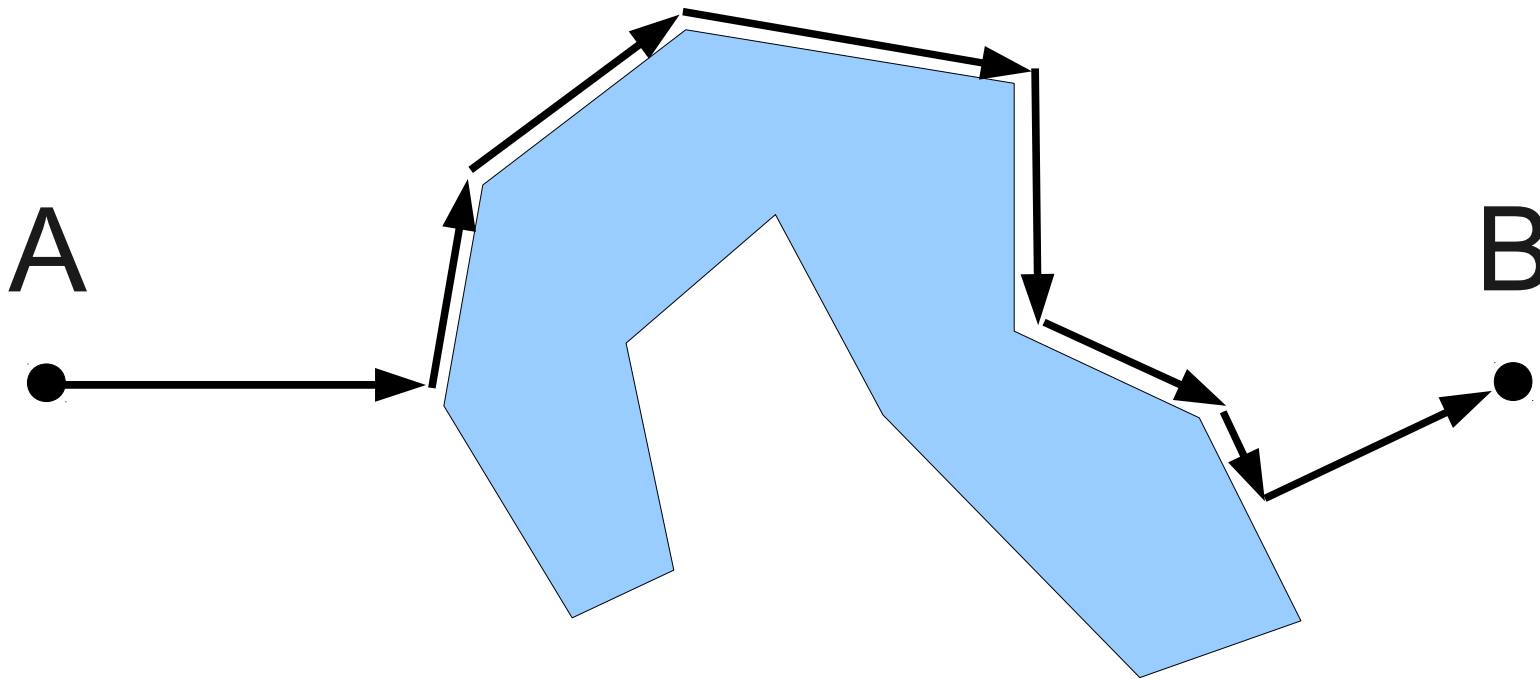
# СТ

- Conditional Traverse - аналогично LHT/RHT, но с выбором направления обхода
  - Например, по вектору нормали к препятствию
- Позволяет выбрать чуть более реалистичный обход



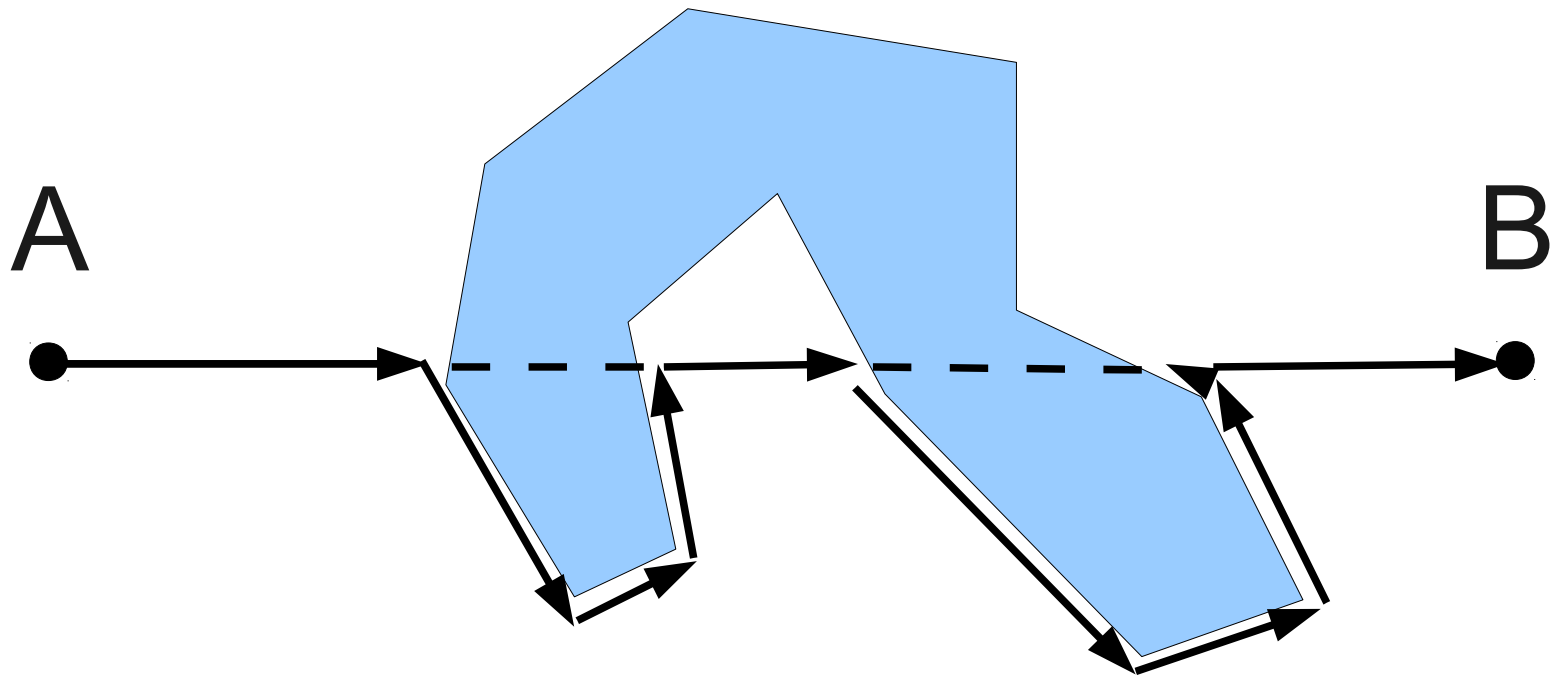
# BUG1

- Обход препятствия до точки, ближайшей к цели
- Гарантированно находит путь



# BUG2

- Обход препятствия до точки выхода луча из препятствия
- Гарантированно находит путь





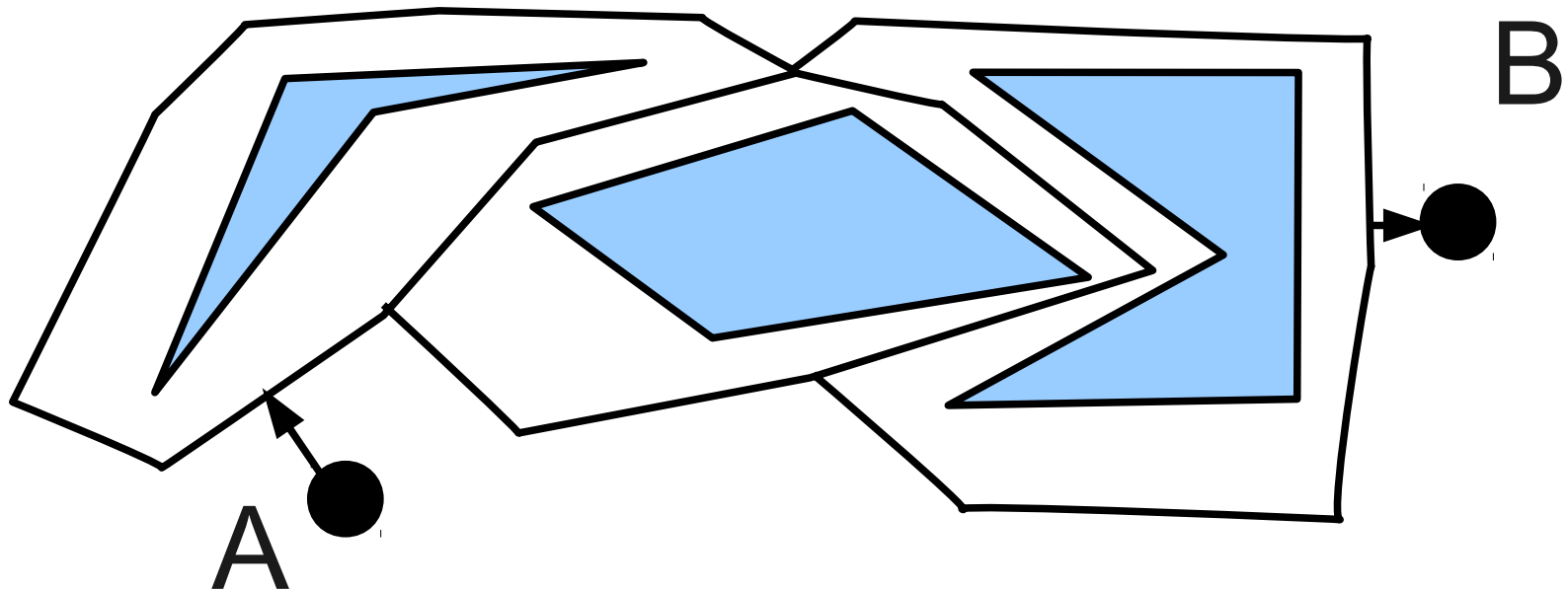
# Решение задачи. Методы, ориентированные на пространство

- Агенту известно всё пространство
- Решение:
  1. Построить граф, моделирующий свойства достижимости
  2. Найти путь в графе
  3. Восстановить по пути в графе путь в пространстве
  4. По возможности сократить путь (если он приближенный)

# Road Graph

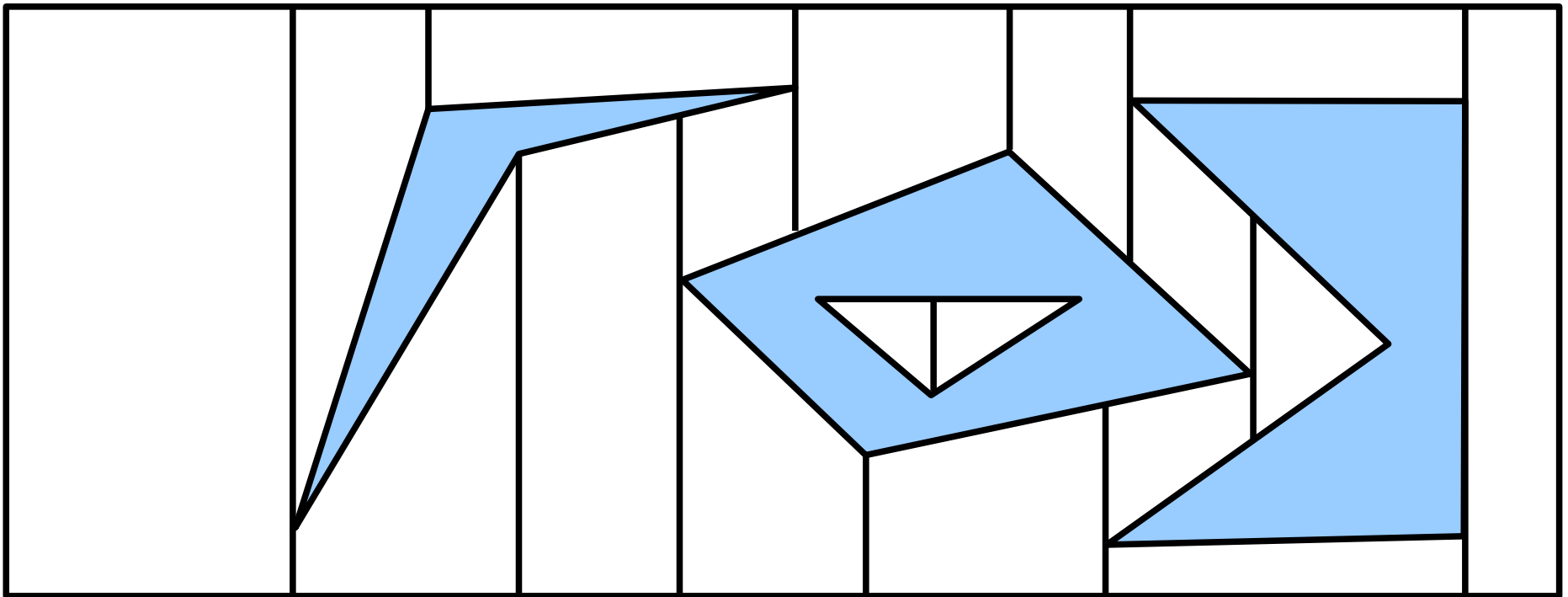
- Построим между препятствиями граф дорог
- Добавим в граф путь между A и ближайшей «дорогой», путь между B и ближайшей «дорогой»
- Найдём путь между A и B в построенном графе.

Как построить граф дорог?



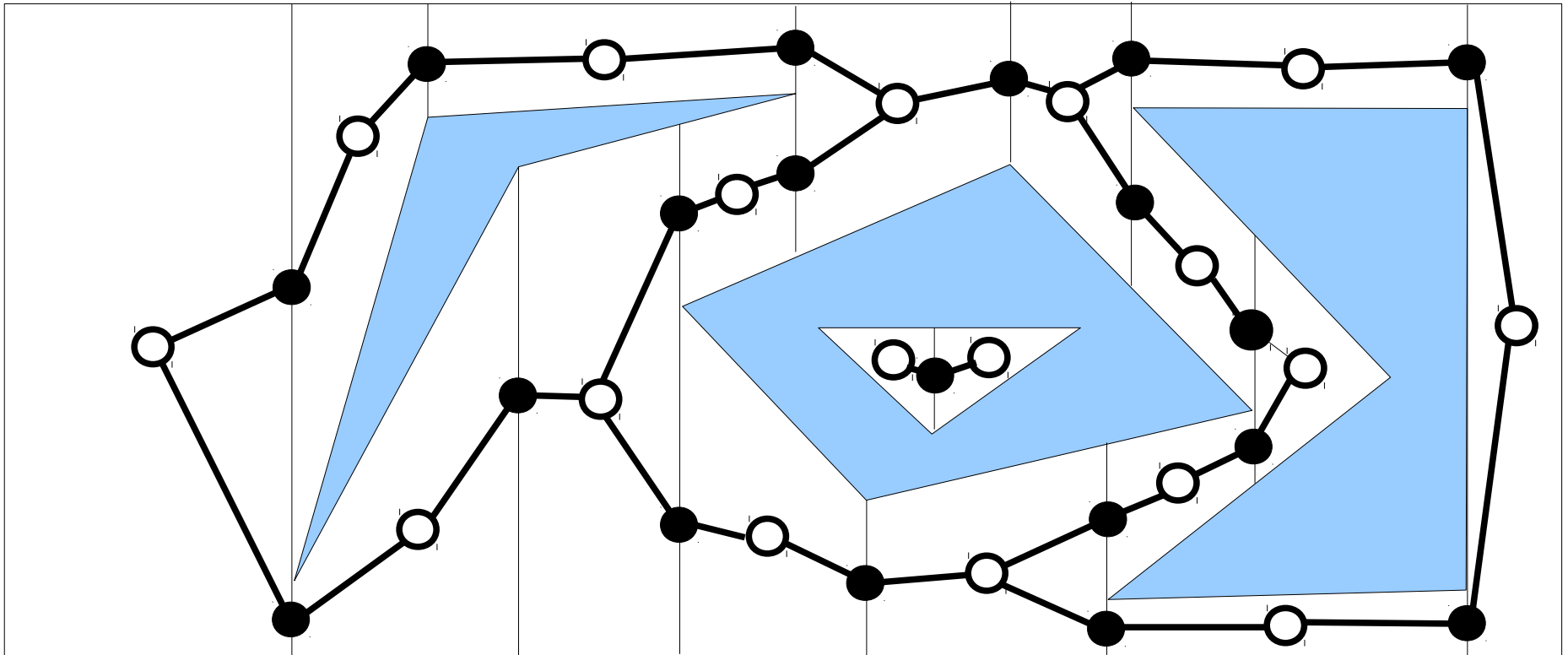
# Road Graph. Трапециодальная карта (1)

- Построим трапециодальную карты по рёбрам препятствий и удалим трапеции, лежащие внутри препятствий
  - Основания трапеций - вертикальные отрезки, содержащие вершины препятствий
  - Боковые грани трапеций - отрезки препятствий



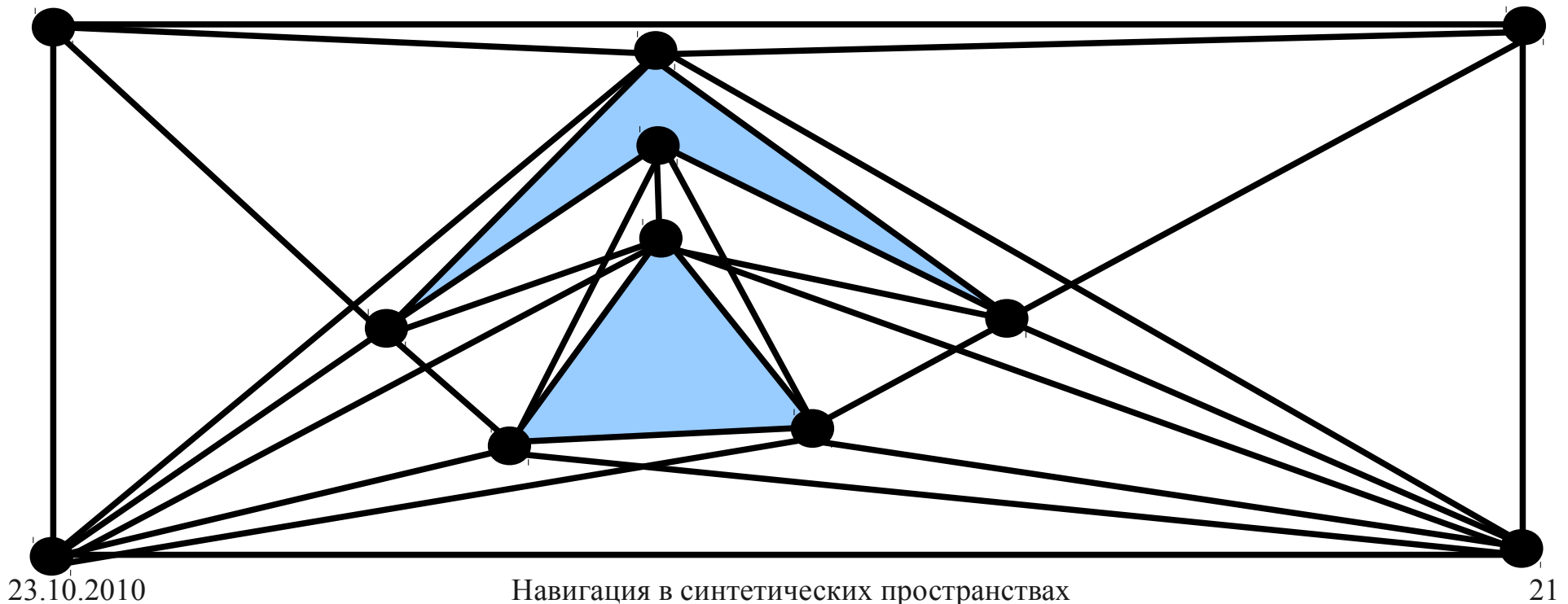
# Road Graph. Трапециодальная карта (2)

- Построим граф по трапециям
  - Вершины в центрах трапеций и на серединах оснований
  - Рёбра внутри трапеций (внутри одной трапеции поиск пути тривиален)



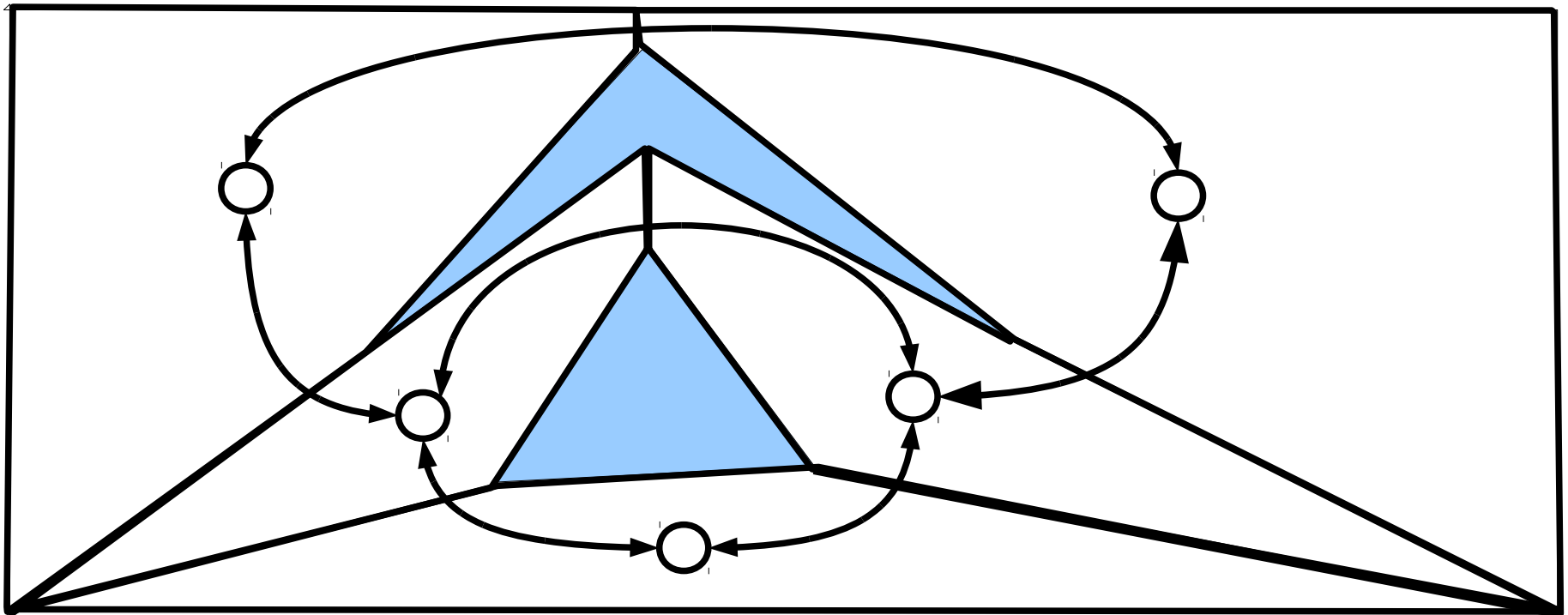
# Visibility Graph

- Построим граф:
  - вершины - вершины препятствий (плюс точки A и B)
  - соединим рёбрами видимые вершины
- Находит кратчайший путь



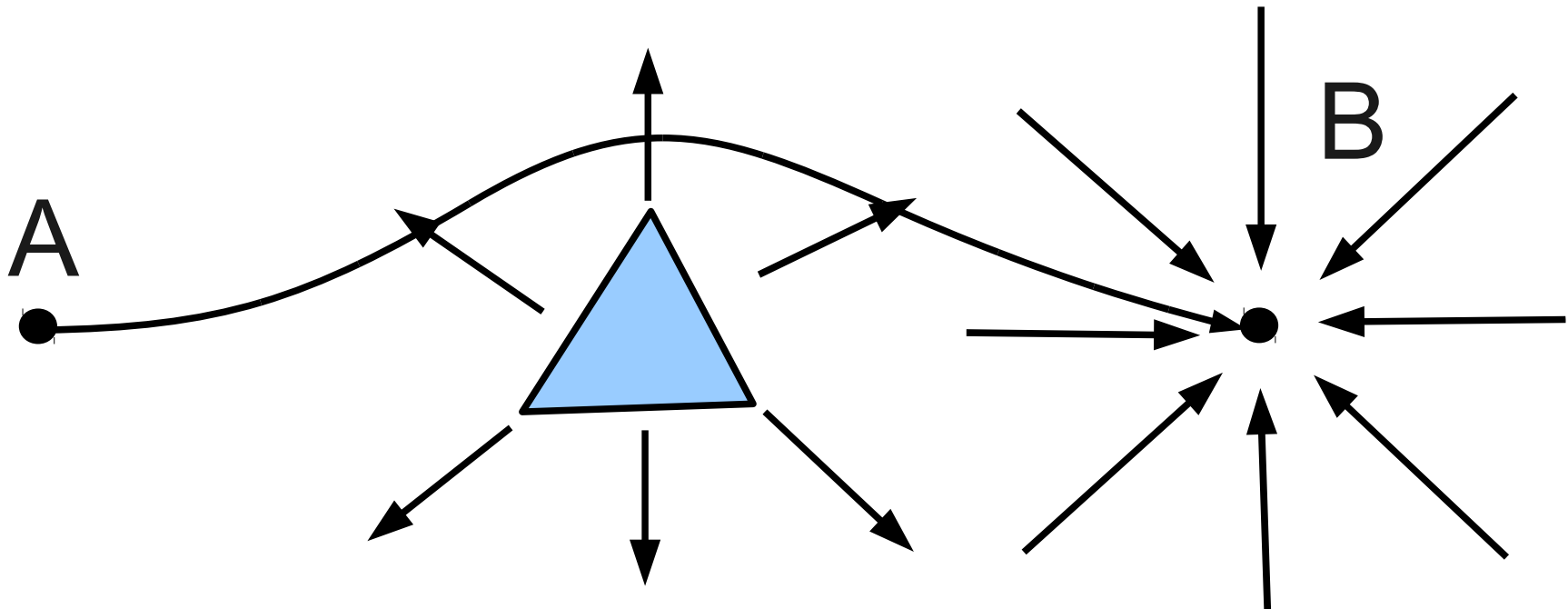
# Метод декомпозиции

- Разбиваем свободное пространство на ячейки такие, что поиск пути между точками внутри ячейки тривиален
  - Например, ячейки - выпуклые или звёздные полигоны
- Ячейка - вершина. Смежные ячейки соединены ребром



# Метод потенциалов

- Вводится потенциальное поле:  $P(v) = G(v) + O(v)$ 
  - $G(v)$  - убывает с приближением  $v$  к цели
  - $O(v)$  - возрастает при приближении к препятствию
- Поиск пути - градиентный спуск
- Не всегда работает: возможно возникновение локальных минимумов



# Выводы

- Рассмотрены две категории алгоритмов поиска пути между двумя точками в 2D агентом-материальной точкой:
  - ориентированные на агента,
  - ориентированные на пространство



# Источники

- С.Ю. Жуков. «Навигация интеллектуальных агентов в сложных синтетических пространствах» Диссертация на соискание ученой степени кандидата физико-математических наук, СПбГТУ, 2000.
- Материалы сайта <http://wikipedia.org/>

Спасибо за внимание!