

Planification de mouvement

Florent Lamiroux

CNRS-LAAS, Toulouse, France

Planification de mouvement

Contexte

robot industriel



drone



véhicule autonome



Systèmes mobiles autonomes

- ▶ se déplaçant dans un environnement encombré d'obstacles,
- ▶ soumis à des contraintes cinématiques ou dynamiques.

Planification de mouvement : calculer automatiquement un mouvement sans collisions entre deux configurations du système.

Contexte

robot industriel



drone



véhicule autonome



Systèmes mobiles autonomes

- ▶ se déplaçant dans un environnement encombré d'obstacles,
- ▶ soumis à des contraintes cinématiques ou dynamiques.

Planification de mouvement : calculer automatiquement un mouvement sans collisions entre deux configurations du système.

Contexte

robot industriel



drone



véhicule autonome



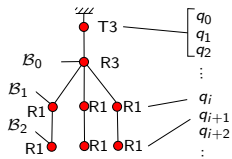
Systèmes mobiles autonomes

- ▶ se déplaçant dans un environnement encombré d'obstacles,
- ▶ soumis à des contraintes cinématiques ou dynamiques.

Planification de mouvement : calculer automatiquement un mouvement sans collisions entre deux configurations du système.

Robot

Un robot est un ensemble de corps solides $\mathcal{B}_0, \dots, \mathcal{B}_m$, reliés entre eux par des *articulations*.



Articulation : déplacement rigide paramétré entre deux corps et à valeur dans $SE(3)$.

Déplacement rigide

Définitions

- ▶ $SO(3)$: groupe des matrices de rotation de dimension 3.

$$R \in SO(3) \Leftrightarrow R^T R = I_3 \text{ et } \det(R) = 1$$

- ▶ $SE(3)$: groupe des déplacement rigides

$$T \in SE(3) \Leftrightarrow \begin{aligned} &\exists t \in \mathbb{R}^3, \exists R \in SO(3) \\ &\forall x \in \mathbb{R}^3 \quad T(x) = Rx + t \end{aligned}$$

On notera $T = T_{(R,t)}$.

Articulation

Une articulation est représentée par une application d'une sous-variété de \mathbb{R}^p dans $SE(3)$, où $p \geq 1$ est un entier.

Exemples :

- ▶ Translation T1 :

$$\begin{array}{ll} \mathbb{R} & \rightarrow SE(3) \\ t & \rightarrow T_{(I_3, (t \ 0 \ 0))} \end{array} \quad \text{translation selon } x$$

Articulation

Une articulation est représentée par une application d'une sous-variété de \mathbb{R}^p dans $SE(3)$, où $p \geq 1$ est un entier.

Exemples :

- ▶ Translation T3 :

$$\begin{array}{ll} \mathbb{R}^3 & \rightarrow SE(3) \\ t & \rightarrow T_{(I_3, t)} \end{array} \quad \text{translation}$$

Articulation

Une articulation est représentée par une application d'une sous-variété de \mathbb{R}^p dans $SE(3)$, où $p \geq 1$ est un entier.

Exemples :

- ▶ Rotation R1 :

$$\mathbb{R} \rightarrow SE(3)$$

$$t \rightarrow T_{(R,0)}$$

$$R = \begin{pmatrix} \cos t & -\sin t & 0 \\ \sin t & \cos t & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Articulation

Une articulation est représentée par une application d'une sous-variété de \mathbb{R}^p dans $SE(3)$, où $p \geq 1$ est un entier.

Exemples :

► Rotation R3 :

$$\begin{aligned}\mathbb{R}^4 &\rightarrow SE(3) \\ t &\rightarrow T_{(R,0)}\end{aligned}$$

$$\|t\| = 1$$

$$R = \begin{pmatrix} 1 - 2(t_2^2 + t_3^2) & 2t_2t_1 - 2t_3t_0 & 2t_3t_1 + 2t_2t_0 \\ 2t_2t_1 + 2t_3t_0 & 1 - 2(t_1^2 + t_3^2) & 2t_3t_2 - 2t_1t_0 \\ 2t_3t_1 - 2t_2t_0 & 2t_3t_2 + 2t_1t_0 & 1 - 2(t_1^2 + t_2^2) \end{pmatrix}$$

$t_0 + t_1i + t_2j + t_3k$ est un quaternion.

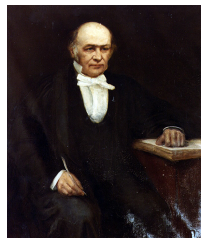
Quaternions

Corps non commutatif isomorphe à \mathbb{R}^4 , engendré par trois éléments i, j, k satisfaisant les relations suivantes :

$$i^2 = j^2 = k^2 = ijk = -1$$

dont on déduit immédiatement

$$ij = k, \quad jk = i, \quad ki = j$$



Hamilton (1843)

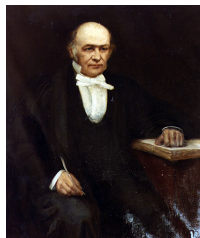
Quaternions

Corps non commutatif isomorphe à \mathbb{R}^4 , engendré par trois éléments i, j, k satisfaisant les relations suivantes :

$$i^2 = j^2 = k^2 = ijk = -1$$

dont on déduit immédiatement

$$ij = k, \quad jk = i, \quad ki = j$$



Hamilton (1843)

Quaternions unitaires et rotations

Soit $q = q_0 + q_1i + q_2j + q_3k$ un quaternion unitaire :

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$$

$\forall x = (x_0, x_1, x_2) \in \mathbb{R}^3$, soit $u = x_0i + x_1j + x_2k$

$$q \cdot u \cdot q^* = y_0i + y_1j + y_2k$$

où $q^* = q_0 - q_1i - q_2j - q_3k$ est le conjugué de q .

$y = (y_0, y_1, y_2)$ est l'image de x par la rotation de matrice

$$\begin{pmatrix} 1 - 2(q_2^2 + q_3^2) & 2q_2q_1 - 2q_3q_0 & 2q_3q_1 + 2q_2q_0 \\ 2q_2q_1 + 2q_3q_0 & 1 - 2(q_1^2 + q_3^2) & 2q_3q_2 - 2q_1q_0 \\ 2q_3q_1 - 2q_2q_0 & 2q_3q_2 + 2q_1q_0 & 1 - 2(q_1^2 + q_2^2) \end{pmatrix}$$

Quaternions unitaires et rotations

Soit $q = q_0 + q_1i + q_2j + q_3k$ un quaternion unitaire :

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$$

$\forall x = (x_0, x_1, x_2) \in \mathbb{R}^3$, soit $u = x_0i + x_1j + x_2k$

$$q \cdot u \cdot q^* = y_0i + y_1j + y_2k$$

où $q^* = q_0 - q_1i - q_2j - q_3k$ est le conjugué de q .

$y = (y_0, y_1, y_2)$ est l'image de x par la rotation de matrice

$$\begin{pmatrix} 1 - 2(q_2^2 + q_3^2) & 2q_2q_1 - 2q_3q_0 & 2q_3q_1 + 2q_2q_0 \\ 2q_2q_1 + 2q_3q_0 & 1 - 2(q_1^2 + q_3^2) & 2q_3q_2 - 2q_1q_0 \\ 2q_3q_1 - 2q_2q_0 & 2q_3q_2 + 2q_1q_0 & 1 - 2(q_1^2 + q_2^2) \end{pmatrix}$$

Quaternions unitaires et rotations

Soit $q = q_0 + q_1i + q_2j + q_3k$ un quaternion unitaire :

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$$

$\forall x = (x_0, x_1, x_2) \in \mathbb{R}^3$, soit $u = x_0i + x_1j + x_2k$

$$q \cdot u \cdot q^* = y_0i + y_1j + y_2k$$

où $q^* = q_0 - q_1i - q_2j - q_3k$ est le conjugué de q .

$y = (y_0, y_1, y_2)$ est l'image de x par la rotation de matrice

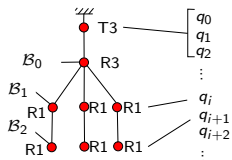
$$\begin{pmatrix} 1 - 2(q_2^2 + q_3^2) & 2q_2q_1 - 2q_3q_0 & 2q_3q_1 + 2q_2q_0 \\ 2q_2q_1 + 2q_3q_0 & 1 - 2(q_1^2 + q_3^2) & 2q_3q_2 - 2q_1q_0 \\ 2q_3q_1 - 2q_2q_0 & 2q_3q_2 + 2q_1q_0 & 1 - 2(q_1^2 + q_2^2) \end{pmatrix}$$

Quaternions unitaires et rotations

- ▶ A noter que q et $-q$ représentent la même rotation.
- ▶ $SO(3)$ est isomorphe à la demi-sphère de \mathbb{R}^4 .

Configuration d'un robot

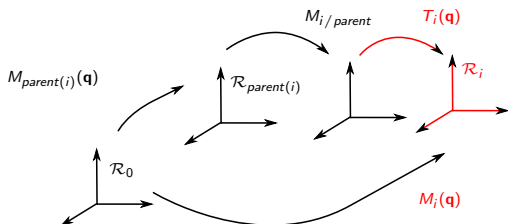
La configuration \mathbf{q} d'un robot est représentée par la concaténation des paramètres de chaque articulation.



Cinématique directe

Calcul de la position de chaque articulation dans le repère global.

$$M_i(\mathbf{q}) = M_{parent(i)}(\mathbf{q}) M_{i/parent} T_i(\mathbf{q})$$



Définitions

- ▶ Espace de travail : $\mathcal{W} = \mathbb{R}^2$ or \mathbb{R}^3 : espace dans lequel le robot se déplace.
- ▶ Obstacle dans l'espace de travail : partie compacte de \mathcal{W} , noté \mathcal{O} .
- ▶ Espace des configurations : \mathcal{C} .
- ▶ Position en configuration \mathbf{q} d'un point $M \in \mathcal{B}_i$: $\mathbf{x}_i(M, \mathbf{q})$.
- ▶ Obstacle dans l'espace des configurations :

$$\mathcal{C}_{obst} = \{ \mathbf{q} \in \mathcal{C}, \exists i \in \{1, \dots, m\}, \exists M \in \mathcal{B}_i, \mathbf{x}_i(M, \mathbf{q}) \in \mathcal{O} \text{ ou} \\ \exists i, j \in \{1, \dots, m\}, \exists M_i \in \mathcal{B}_i, \exists M_j \in \mathcal{B}_j, \\ \mathbf{x}_i(M_i, \mathbf{q}) = \mathbf{x}_j(M_j, \mathbf{q}) \}$$

- ▶ Espace des configurations libres : $\mathcal{C}_{free} = \mathcal{C} \setminus \mathcal{C}_{obst}$.

Définitions

- ▶ Espace de travail : $\mathcal{W} = \mathbb{R}^2$ or \mathbb{R}^3 : espace dans lequel le robot se déplace.
- ▶ Obstacle dans l'espace de travail : partie compacte de \mathcal{W} , noté \mathcal{O} .
- ▶ Espace des configurations : \mathcal{C} .
- ▶ Position en configuration \mathbf{q} d'un point $M \in \mathcal{B}_i$: $\mathbf{x}_i(M, \mathbf{q})$.
- ▶ Obstacle dans l'espace des configurations :

$$\mathcal{C}_{obst} = \{ \mathbf{q} \in \mathcal{C}, \exists i \in \{1, \dots, m\}, \exists M \in \mathcal{B}_i, \mathbf{x}_i(M, \mathbf{q}) \in \mathcal{O} \text{ ou} \\ \exists i, j \in \{1, \dots, m\}, \exists M_i \in \mathcal{B}_i, \exists M_j \in \mathcal{B}_j, \\ \mathbf{x}_i(M_i, \mathbf{q}) = \mathbf{x}_j(M_j, \mathbf{q}) \}$$

- ▶ Espace des configurations libres : $\mathcal{C}_{free} = \mathcal{C} \setminus \mathcal{C}_{obst}$.

Définitions

- ▶ Espace de travail : $\mathcal{W} = \mathbb{R}^2$ or \mathbb{R}^3 : espace dans lequel le robot se déplace.
- ▶ Obstacle dans l'espace de travail : partie compacte de \mathcal{W} , noté \mathcal{O} .
- ▶ Espace des configurations : \mathcal{C} .
- ▶ Position en configuration \mathbf{q} d'un point $M \in \mathcal{B}_i$: $\mathbf{x}_i(M, \mathbf{q})$.
- ▶ Obstacle dans l'espace des configurations :

$$\mathcal{C}_{obst} = \{ \mathbf{q} \in \mathcal{C}, \exists i \in \{1, \dots, m\}, \exists M \in \mathcal{B}_i, \mathbf{x}_i(M, \mathbf{q}) \in \mathcal{O} \text{ ou} \\ \exists i, j \in \{1, \dots, m\}, \exists M_i \in \mathcal{B}_i, \exists M_j \in \mathcal{B}_j, \\ \mathbf{x}_i(M_i, \mathbf{q}) = \mathbf{x}_j(M_j, \mathbf{q}) \}$$

- ▶ Espace des configurations libres : $\mathcal{C}_{free} = \mathcal{C} \setminus \mathcal{C}_{obst}$.

Définitions

- ▶ Espace de travail : $\mathcal{W} = \mathbb{R}^2$ or \mathbb{R}^3 : espace dans lequel le robot se déplace.
- ▶ Obstacle dans l'espace de travail : partie compacte de \mathcal{W} , noté \mathcal{O} .
- ▶ Espace des configurations : \mathcal{C} .
- ▶ Position en configuration \mathbf{q} d'un point $M \in \mathcal{B}_i$: $\mathbf{x}_i(M, \mathbf{q})$.
- ▶ Obstacle dans l'espace des configurations :

$$\mathcal{C}_{obst} = \{ \mathbf{q} \in \mathcal{C}, \exists i \in \{1, \dots, m\}, \exists M \in \mathcal{B}_i, \mathbf{x}_i(M, \mathbf{q}) \in \mathcal{O} \text{ ou} \\ \exists i, j \in \{1, \dots, m\}, \exists M_i \in \mathcal{B}_i, \exists M_j \in \mathcal{B}_j, \\ \mathbf{x}_i(M_i, \mathbf{q}) = \mathbf{x}_j(M_j, \mathbf{q}) \}$$

- ▶ Espace des configurations libres : $\mathcal{C}_{free} = \mathcal{C} \setminus \mathcal{C}_{obst}$.

Définitions

- ▶ Espace de travail : $\mathcal{W} = \mathbb{R}^2$ or \mathbb{R}^3 : espace dans lequel le robot se déplace.
- ▶ Obstacle dans l'espace de travail : partie compacte de \mathcal{W} , noté \mathcal{O} .
- ▶ Espace des configurations : \mathcal{C} .
- ▶ Position en configuration \mathbf{q} d'un point $M \in \mathcal{B}_i$: $\mathbf{x}_i(M, \mathbf{q})$.
- ▶ Obstacle dans l'espace des configurations :

$$\mathcal{C}_{obst} = \{ \mathbf{q} \in \mathcal{C}, \exists i \in \{1, \dots, m\}, \exists M \in \mathcal{B}_i, \mathbf{x}_i(M, \mathbf{q}) \in \mathcal{O} \text{ ou} \\ \exists i, j \in \{1, \dots, m\}, \exists M_i \in \mathcal{B}_i, \exists M_j \in \mathcal{B}_j, \\ \mathbf{x}_i(M_i, \mathbf{q}) = \mathbf{x}_j(M_j, \mathbf{q}) \}$$

- ▶ Espace des configurations libres : $\mathcal{C}_{free} = \mathcal{C} \setminus \mathcal{C}_{obst}$.

Définitions

- ▶ Espace de travail : $\mathcal{W} = \mathbb{R}^2$ or \mathbb{R}^3 : espace dans lequel le robot se déplace.
- ▶ Obstacle dans l'espace de travail : partie compacte de \mathcal{W} , noté \mathcal{O} .
- ▶ Espace des configurations : \mathcal{C} .
- ▶ Position en configuration \mathbf{q} d'un point $M \in \mathcal{B}_i$: $\mathbf{x}_i(M, \mathbf{q})$.
- ▶ Obstacle dans l'espace des configurations :

$$\mathcal{C}_{obst} = \{ \mathbf{q} \in \mathcal{C}, \exists i \in \{1, \dots, m\}, \exists M \in \mathcal{B}_i, \mathbf{x}_i(M, \mathbf{q}) \in \mathcal{O} \text{ ou} \\ \exists i, j \in \{1, \dots, m\}, \exists M_i \in \mathcal{B}_i, \exists M_j \in \mathcal{B}_j, \\ \mathbf{x}_i(M_i, \mathbf{q}) = \mathbf{x}_j(M_j, \mathbf{q}) \}$$

- ▶ Espace des configurations libres : $\mathcal{C}_{free} = \mathcal{C} \setminus \mathcal{C}_{obst}$.

Mouvement

- ▶ Espace des configurations :
 - ▶ variété différentielle
- ▶ Mouvement :
 - ▶ fonction continue de $[0, 1]$ dans \mathcal{C} .
- ▶ Mouvement sans collision :
 - ▶ fonction continue de $[0, 1]$ dans \mathcal{C}_{free} .

Mouvement

- ▶ Espace des configurations :
 - ▶ variété différentielle
- ▶ Mouvement :
 - ▶ fonction continue de $[0, 1]$ dans \mathcal{C} .
- ▶ Mouvement sans collision :
 - ▶ fonction continue de $[0, 1]$ dans \mathcal{C}_{free} .

Mouvement

- ▶ Espace des configurations :
 - ▶ variété différentielle
- ▶ Mouvement :
 - ▶ fonction continue de $[0, 1]$ dans \mathcal{C} .
- ▶ Mouvement sans collision :
 - ▶ fonction continue de $[0, 1]$ dans \mathcal{C}_{free} .

Méthodes aléatoires

- ▶ Au début des années 1990 sont apparues des méthodes aléatoires
- ▶ Principe
 - ▶ tirer aléatoirement des configurations,
 - ▶ tester si elles sont en collision,
 - ▶ construire un graphe dont les noeuds sont des configurations sans collisions
 - ▶ et dont les arêtes sont des interpolations linéaires sans collisions.

Méthodes aléatoires

- ▶ Au début des années 1990 sont apparues des méthodes aléatoires
- ▶ Principe
 - ▶ tirer aléatoirement des configurations,
 - ▶ tester si elles sont en collision,
 - ▶ construire un graphe dont les noeuds sont des configurations sans collisions
 - ▶ et dont les arêtes sont des interpolations linéaires sans collisions.

Méthodes aléatoires

- ▶ Au début des années 1990 sont apparues des méthodes aléatoires
- ▶ Principe
 - ▶ tirer aléatoirement des configurations,
 - ▶ tester si elles sont en collision,
 - ▶ construire un graphe dont les noeuds sont des configurations sans collisions
 - ▶ et dont les arêtes sont des interpolations linéaires sans collisions.

Méthodes aléatoires

- ▶ Au début des années 1990 sont apparues des méthodes aléatoires
- ▶ Principe
 - ▶ tirer aléatoirement des configurations,
 - ▶ tester si elles sont en collision,
 - ▶ construire un graphe dont les noeuds sont des configurations sans collisions
 - ▶ et dont les arêtes sont des interpolations linéaires sans collisions.

Méthodes aléatoires

- ▶ Au début des années 1990 sont apparues des méthodes aléatoires
- ▶ Principe
 - ▶ tirer aléatoirement des configurations,
 - ▶ tester si elles sont en collision,
 - ▶ construire un graphe dont les noeuds sont des configurations sans collisions
 - ▶ et dont les arêtes sont des interpolations linéaires sans collisions.

Probabilistic roadmap (PRM) 1994

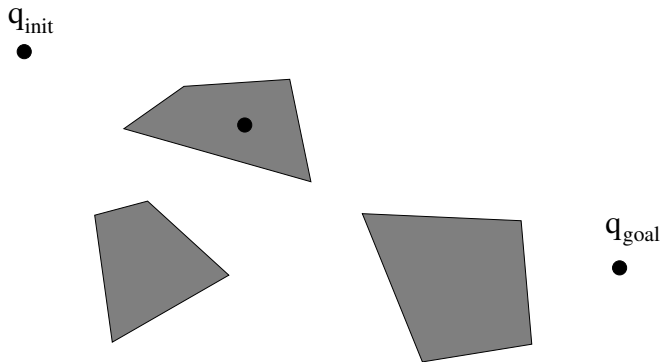
q_{init}



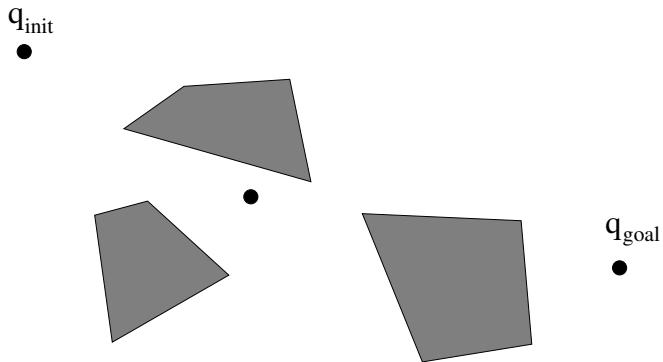
The diagram shows a 2D environment with three gray polygonal obstacles. The start point q_{init} is a black dot in the upper left, and the goal point q_{goal} is a black dot in the middle right. The obstacles are: a quadrilateral at the top center, a quadrilateral at the bottom left, and a quadrilateral at the bottom right.

q_{goal}

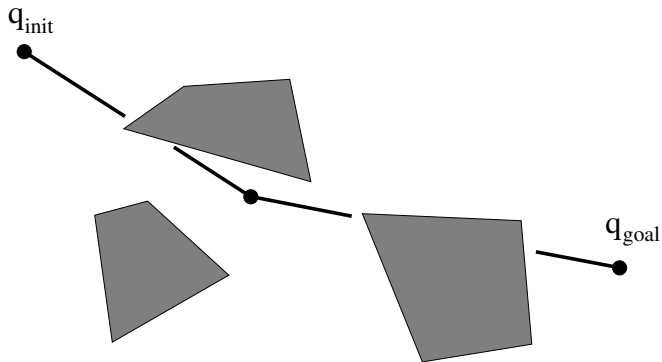
Probabilistic roadmap (PRM) 1994



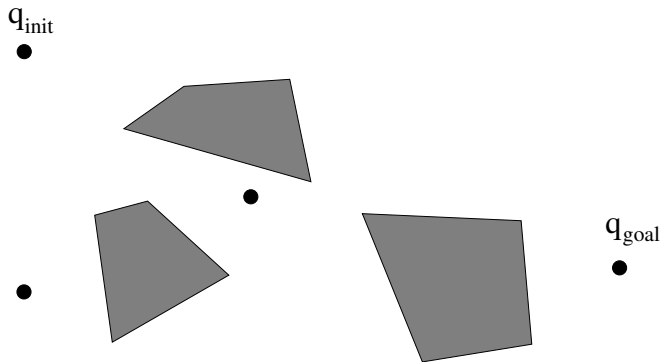
Probabilistic roadmap (PRM) 1994



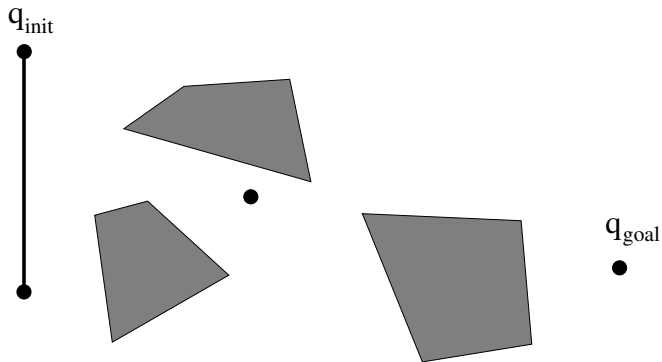
Probabilistic roadmap (PRM) 1994



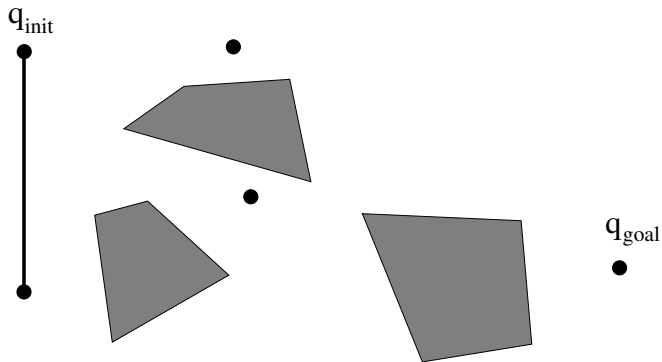
Probabilistic roadmap (PRM) 1994



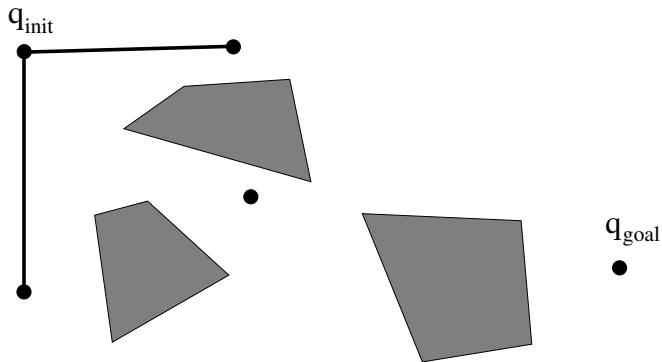
Probabilistic roadmap (PRM) 1994



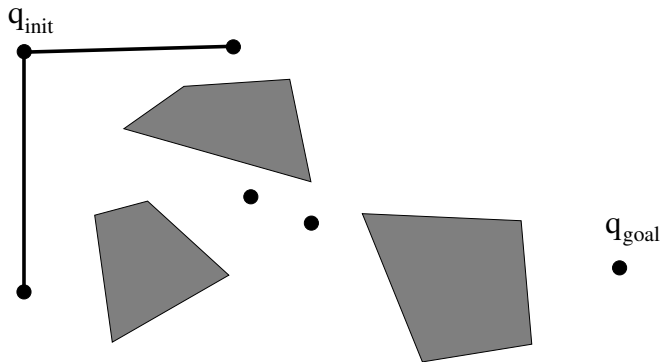
Probabilistic roadmap (PRM) 1994



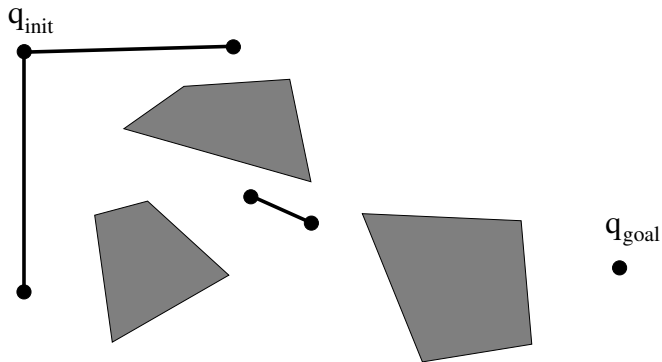
Probabilistic roadmap (PRM) 1994



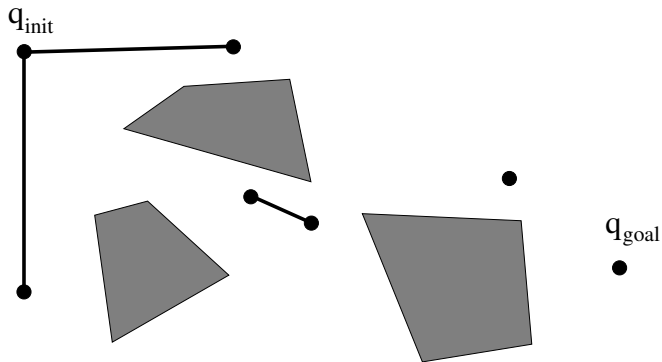
Probabilistic roadmap (PRM) 1994



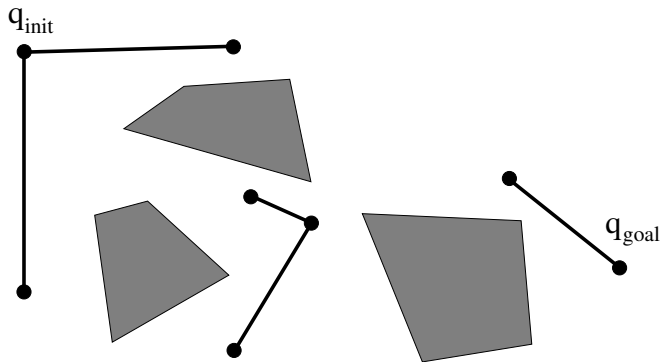
Probabilistic roadmap (PRM) 1994



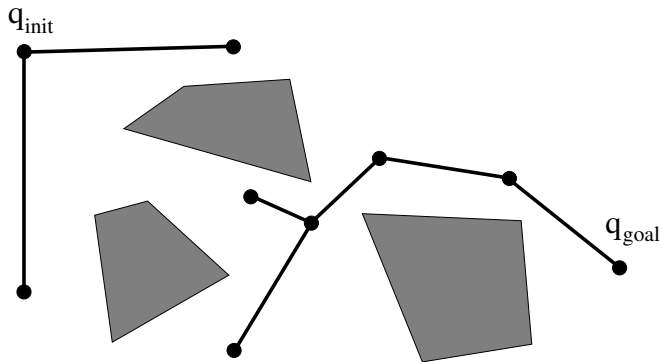
Probabilistic roadmap (PRM) 1994



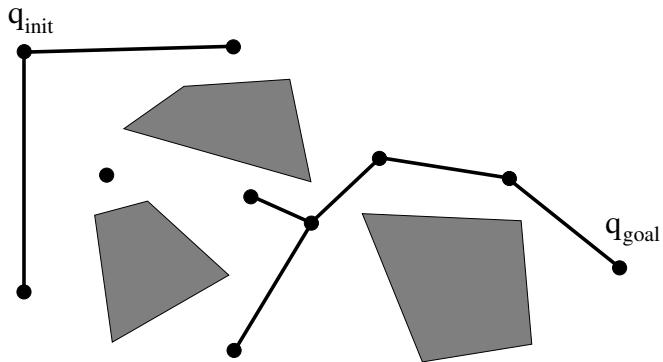
Probabilistic roadmap (PRM) 1994



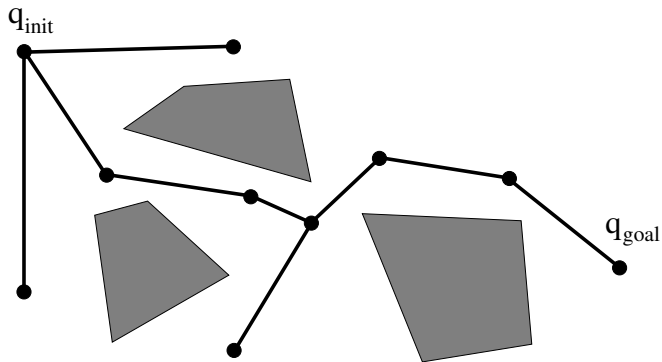
Probabilistic roadmap (PRM) 1994



Probabilistic roadmap (PRM) 1994



Probabilistic roadmap (PRM) 1994



Probabilistic roadmap (PRM)

- ▶ Beaucoup de noeuds inutiles sont créés.
 - ▶ Cela rend la connexion de nouveaux noeuds au graphe existant de plus en plus coûteux.
- ▶ Variante : visibility-based PRM
 - ▶ On ne garde que les noeuds *intéressants*.

Visibility-based probabilistic roadmap (Visi-PRM) 1999

q_{init}



The diagram shows a 2D environment with three gray polygonal obstacles. The start point q_{init} is a black dot in the upper left, and the goal point q_{goal} is a black dot in the middle right. The obstacles are: a top obstacle, a bottom-left obstacle, and a bottom-right obstacle. There is a clear path from the start to the goal through the gaps between these obstacles.

q_{goal}

Visibility-based probabilistic roadmap (Visi-PRM) 1999

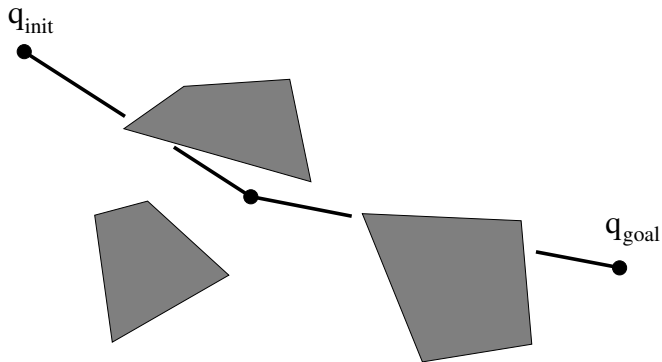
q_{init}



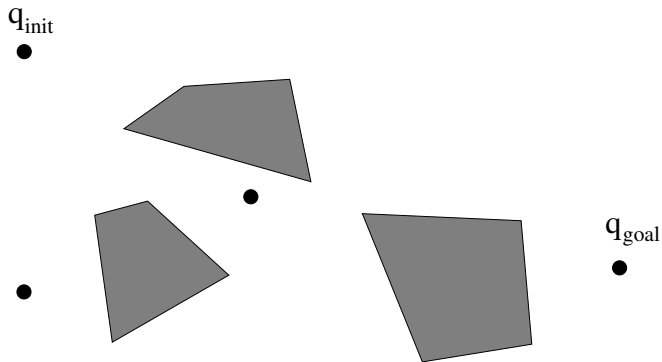
The diagram shows a 2D environment with three gray polygonal obstacles. The start point q_{init} is a black dot in the upper left. The goal point q_{goal} is a black dot on the right side. A third black dot is located in the central open area of the environment.

q_{goal}

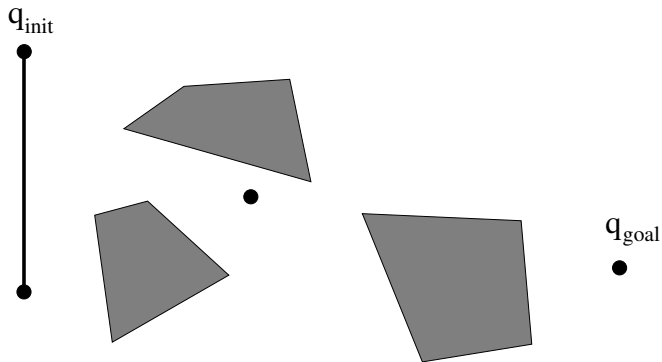
Visibility-based probabilistic roadmap (Visi-PRM) 1999



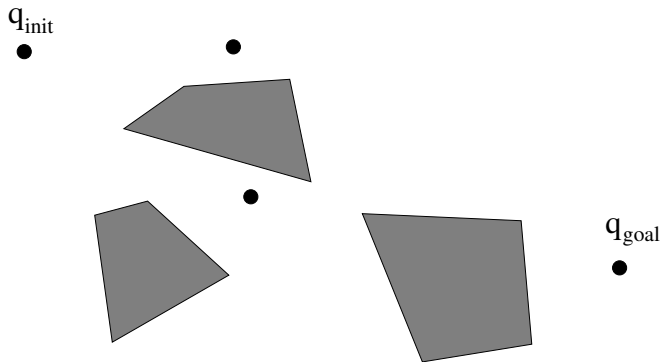
Visibility-based probabilistic roadmap (Visi-PRM) 1999



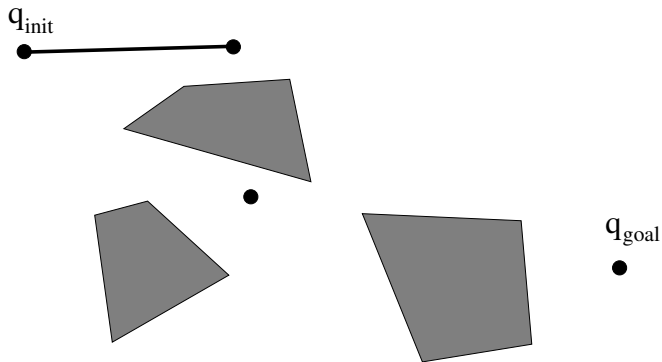
Visibility-based probabilistic roadmap (Visi-PRM) 1999



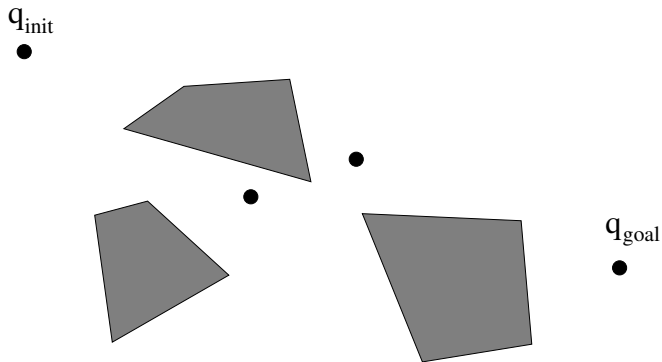
Visibility-based probabilistic roadmap (Visi-PRM) 1999



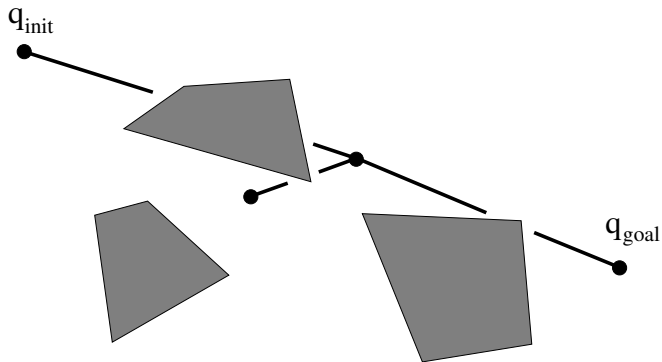
Visibility-based probabilistic roadmap (Visi-PRM) 1999



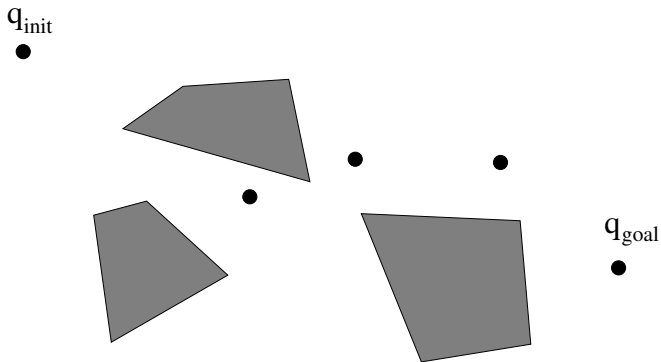
Visibility-based probabilistic roadmap (Visi-PRM) 1999



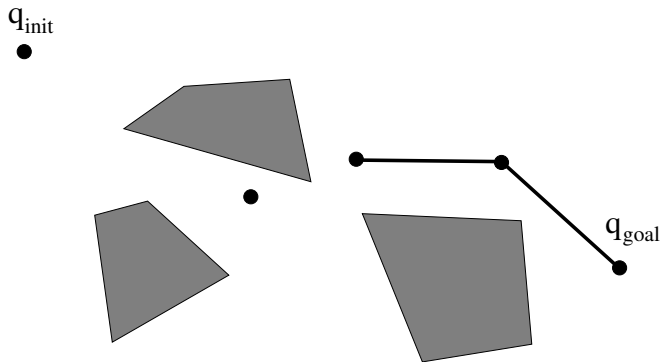
Visibility-based probabilistic roadmap (Visi-PRM) 1999



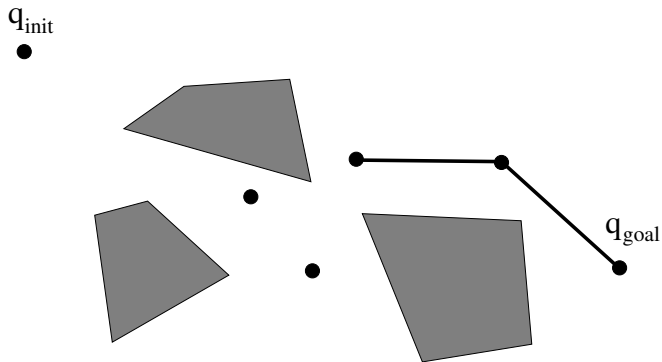
Visibility-based probabilistic roadmap (Visi-PRM) 1999



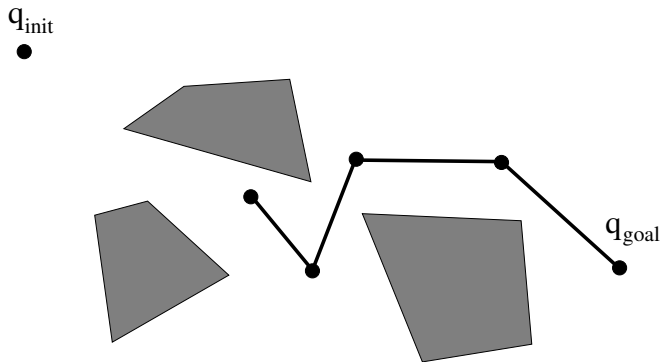
Visibility-based probabilistic roadmap (Visi-PRM) 1999



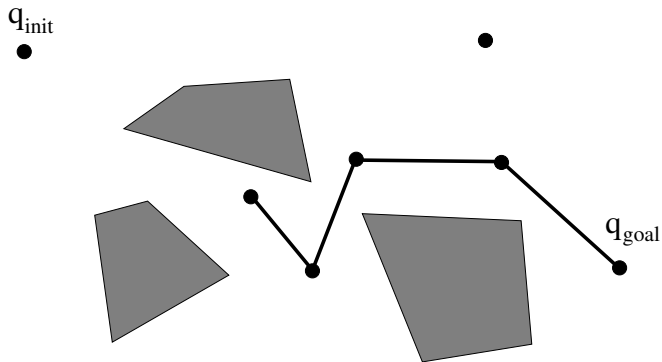
Visibility-based probabilistic roadmap (Visi-PRM) 1999



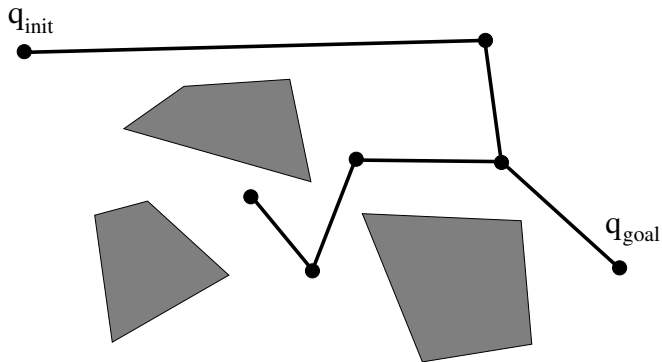
Visibility-based probabilistic roadmap (Visi-PRM) 1999



Visibility-based probabilistic roadmap (Visi-PRM) 1999



Visibility-based probabilistic roadmap (Visi-PRM) 1999



Rapidly exploring Random Tree (RRT) 2000

q_{init}



The diagram shows a 2D environment with three gray polygonal obstacles. The start point q_{init} is a black dot in the upper left, and the goal point q_{goal} is a black dot on the right. The obstacles are: a large irregular polygon at the top center, a smaller irregular polygon at the bottom left, and a large irregular polygon at the bottom right.

q_{goal}

Rapidly exploring Random Tree (RRT) 2000

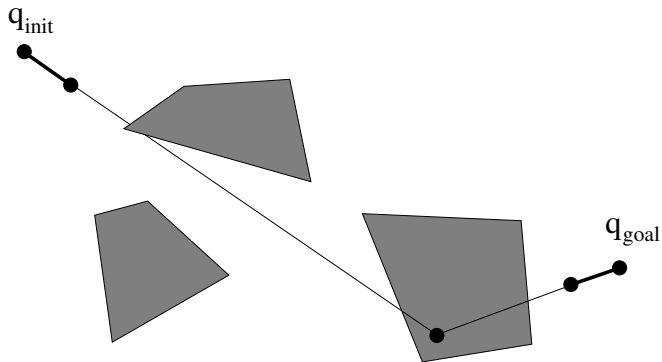
q_{init}



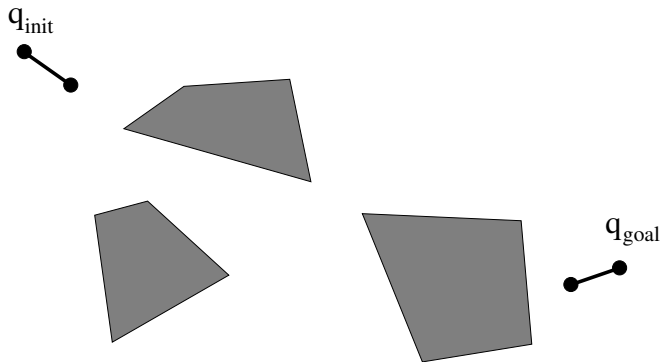
The diagram shows a 2D environment with three gray polygonal obstacles. The start point q_{init} is a black dot in the upper left. The goal point q_{goal} is a black dot in the middle right. A small black dot is also located inside the bottom-right obstacle.

q_{goal}

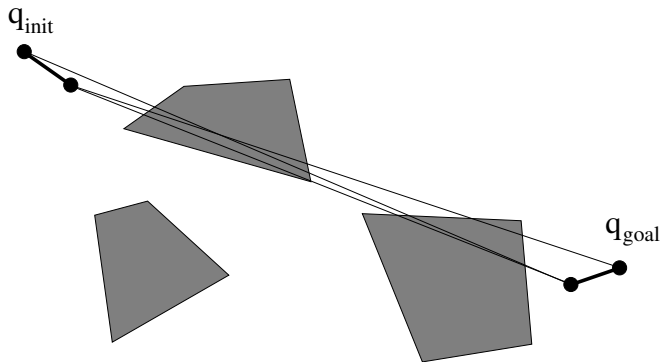
Rapidly exploring Random Tree (RRT) 2000



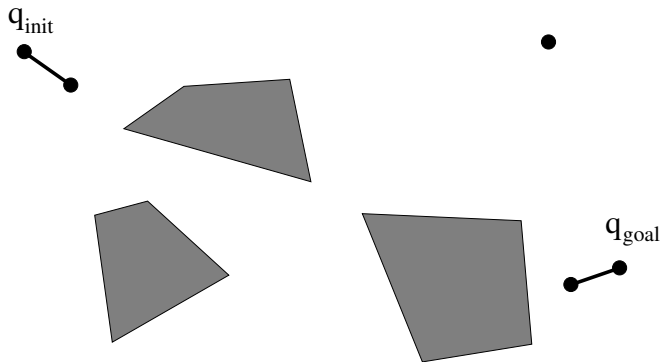
Rapidly exploring Random Tree (RRT) 2000



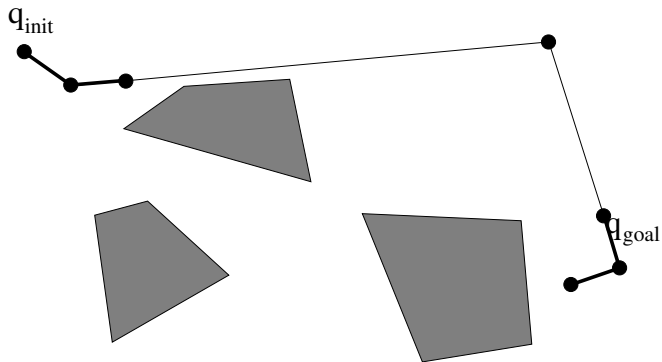
Rapidly exploring Random Tree (RRT) 2000



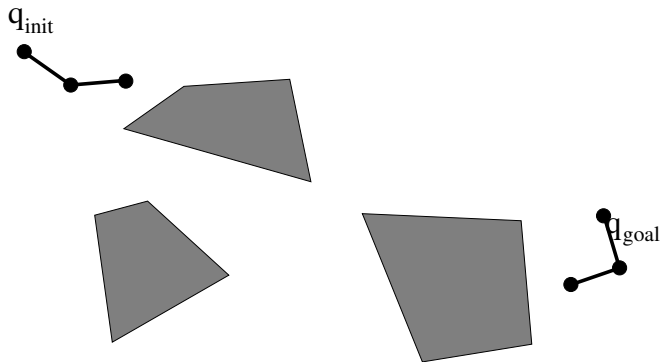
Rapidly exploring Random Tree (RRT) 2000



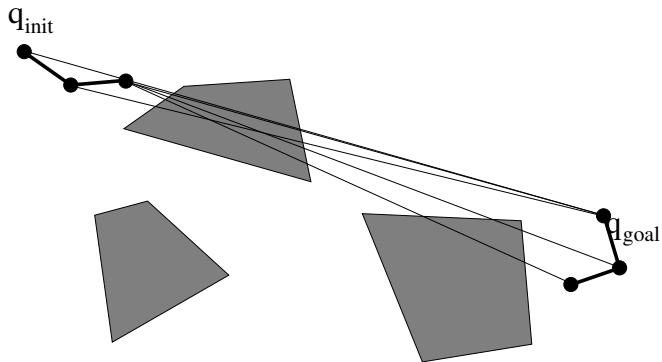
Rapidly exploring Random Tree (RRT) 2000



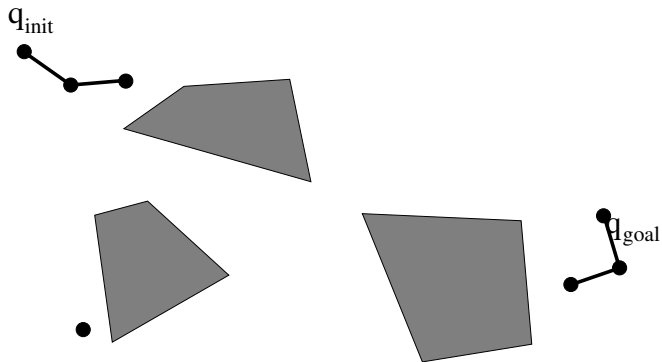
Rapidly exploring Random Tree (RRT) 2000



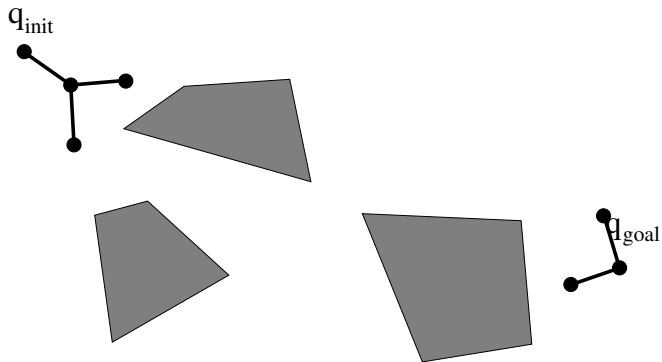
Rapidly exploring Random Tree (RRT) 2000



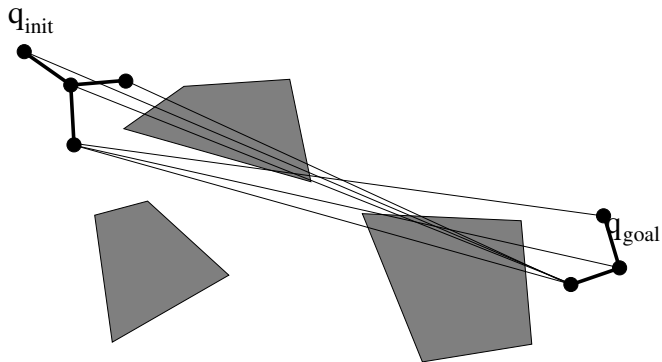
Rapidly exploring Random Tree (RRT) 2000



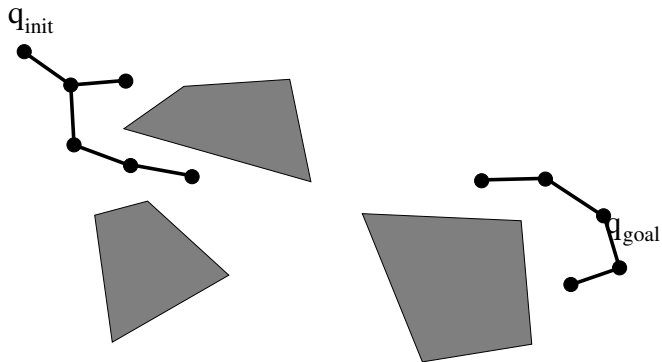
Rapidly exploring Random Tree (RRT) 2000



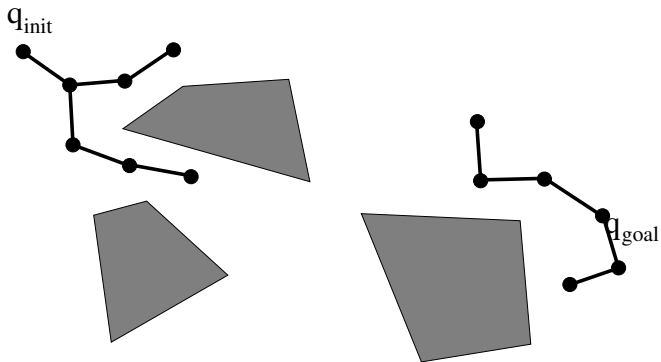
Rapidly exploring Random Tree (RRT) 2000



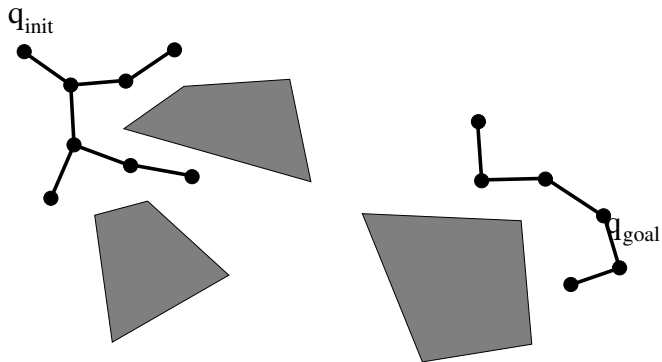
Rapidly exploring Random Tree (RRT) 2000



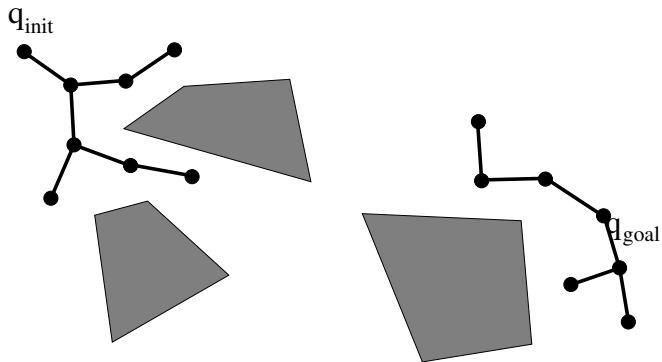
Rapidly exploring Random Tree (RRT) 2000



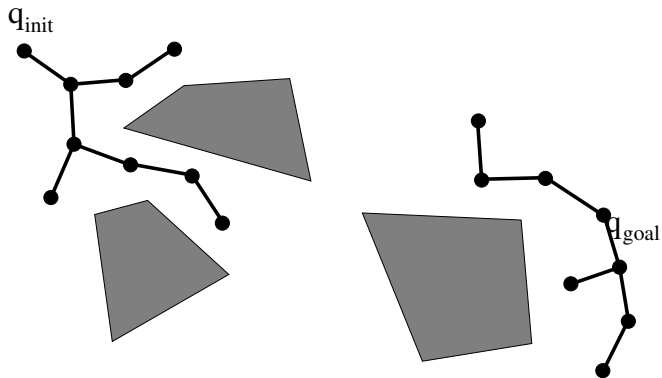
Rapidly exploring Random Tree (RRT) 2000



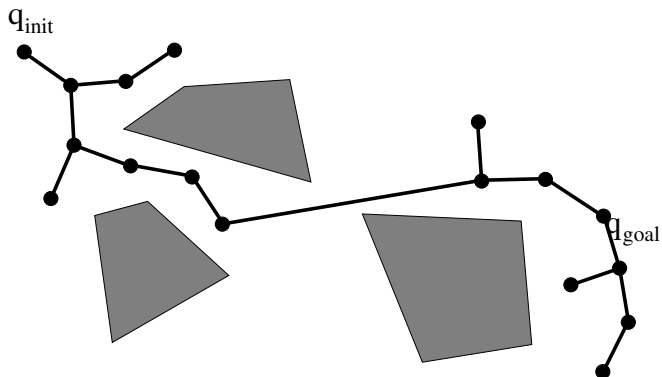
Rapidly exploring Random Tree (RRT) 2000



Rapidly exploring Random Tree (RRT) 2000



Rapidly exploring Random Tree (RRT) 2000



Méthodes aléatoires

- ▶ Avantages :
 - ▶ pas de calcul explicite de l'espace des configurations,
 - ▶ faciles à implémenter,
 - ▶ robustes.
- ▶ Inconvénients :
 - ▶ pas de propriété de complétude, complétude en probabilité seulement
 - ▶ difficile de trouver les passages étroits.
- ▶ Opérateurs requis :
 - ▶ Test de collision
 - ▶ pour les configurations,
 - ▶ pour les mouvements d'interpolation.

Méthodes aléatoires

- ▶ Avantages :
 - ▶ pas de calcul explicite de l'espace des configurations,
 - ▶ faciles à implémenter,
 - ▶ robustes.
- ▶ Inconvénients :
 - ▶ pas de propriété de complétude, complétude en probabilité seulement
 - ▶ difficile de trouver les passages étroits.
- ▶ Opérateurs requis :
 - ▶ Test de collision
 - ▶ pour les configurations,
 - ▶ pour les mouvements d'interpolation.

Méthodes aléatoires

- ▶ Avantages :
 - ▶ pas de calcul explicite de l'espace des configurations,
 - ▶ faciles à implémenter,
 - ▶ robustes.
- ▶ Inconvénients :
 - ▶ pas de propriété de complétude, complétude en probabilité seulement
 - ▶ difficile de trouver les passages étroits.
- ▶ Opérateurs requis :
 - ▶ Test de collision
 - ▶ pour les configurations,
 - ▶ pour les mouvements d'interpolation.

Tests de collision

- ▶ pour configurations
 - ▶ problème : étant donné
 - ▶ deux ensembles rigides de facettes triangulaires,
 - ▶ la position relative de l'un par rapport à l'autre,déterminer si l'intersection entre les ensembles est vide, ou bien calculer la distance entre les deux ensembles.

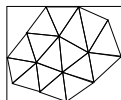
Hiérarchie de volumes englobants

- ▶ Arbres binaires de volumes englobants tels que
 - ▶ chaque noeud contient deux fils,
 - ▶ les feuilles sont les facettes triangulaires.



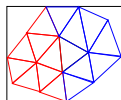
Hiérarchie de volumes englobants

- ▶ Arbres binaires de volumes englobants tels que
 - ▶ chaque noeud contient deux fils,
 - ▶ les feuilles sont les facettes triangulaires.



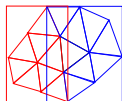
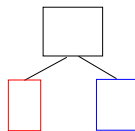
Hiérarchie de volumes englobants

- ▶ Arbres binaires de volumes englobants tels que
 - ▶ chaque noeud contient deux fils,
 - ▶ les feuilles sont les facettes triangulaires.



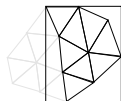
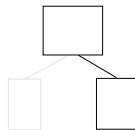
Hiérarchie de volumes englobants

- ▶ Arbres binaires de volumes englobants tels que
 - ▶ chaque noeud contient deux fils,
 - ▶ les feuilles sont les facettes triangulaires.



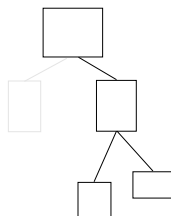
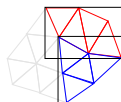
Hiérarchie de volumes englobants

- ▶ Arbres binaires de volumes englobants tels que
 - ▶ chaque noeud contient deux fils,
 - ▶ les feuilles sont les facettes triangulaires.



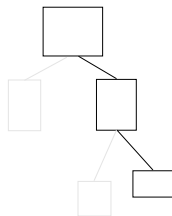
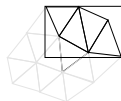
Hiérarchie de volumes englobants

- ▶ Arbres binaires de volumes englobants tels que
 - ▶ chaque noeud contient deux fils,
 - ▶ les feuilles sont les facettes triangulaires.



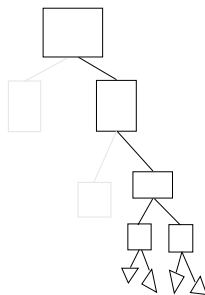
Hiérarchie de volumes englobants

- ▶ Arbres binaires de volumes englobants tels que
 - ▶ chaque noeud contient deux fils,
 - ▶ les feuilles sont les facettes triangulaires.



Hiérarchie de volumes englobants

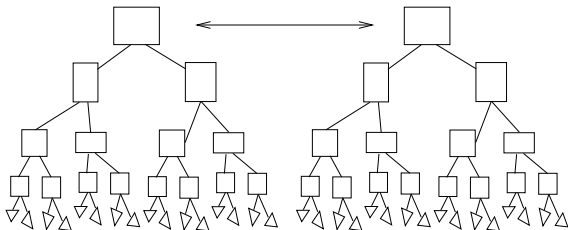
- ▶ Arbres binaires de volumes englobants tels que
 - ▶ chaque noeud contient deux fils,
 - ▶ les feuilles sont les facettes triangulaires.



Tests de collision pour configurations

► Algorithme

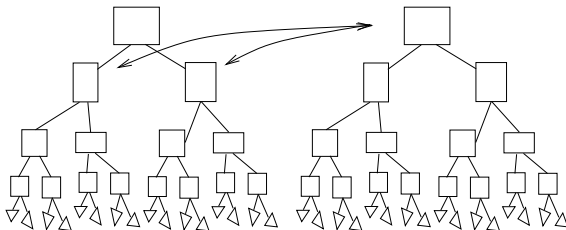
- tester les noeuds racine de chaque arbre,
- si deux volumes sont en collision, tester l'un avec les fils de l'autre.



Tests de collision pour configurations

► Algorithme

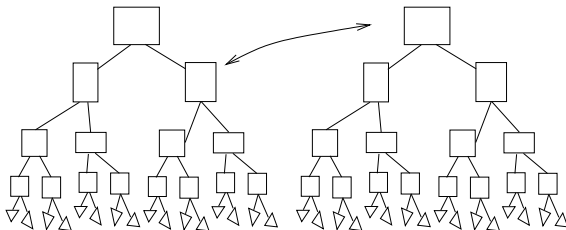
- tester les noeuds racine de chaque arbre,
- si deux volumes sont en collision, tester l'un avec les fils de l'autre.



Tests de collision pour configurations

► Algorithme

- tester les noeuds racine de chaque arbre,
- si deux volumes sont en collision, tester l'un avec les fils de l'autre.



Tests de collision pour configurations

► Algorithme

- tester les noeuds racines de chaque arbre,
- si deux volumes sont en collision, tester l'un avec les fils de l'autre.

