

Robotique: Outils d'imagerie

Positionnement

Jean-Philippe BRUNET

SLAM:

Simultaneous Localisation And Mapping

A la base: passer du point de vue localisé du robot (POV) a un point de vue global localisant le robot (MAP) dans un environnement fixe mais inconnu.

Les algorithmes SLAM sont l'une des grandes réussites de la robotique probabiliste.

Quand avons-nous besoin des algorithmes SLAM

- Quand un robot doit être vraiment autonome (pas d'intervention humaine).
- Quand le robot a réellement besoin de savoir où il se trouve
- Quand on ne sait rien à l'avance sur l'environnement hormis le fait qu'il est majoritairement fixe.
- Lorsque nous ne pouvons pas placer de balises ou même utiliser le GPS comme à l'intérieur ou sous l'eau.

Comment ca marche: les entrées

- La plupart des algorithmes SLAM créent des cartes des caractéristiques naturelles de la scène a partir de capteurs 2d ou 3d et facultativement d'une évaluation du déplacement.
- Caméras, lidar, sonar vont fournir des segments de ligne, des plans 3D, des coins, etc...
- Ces points fixes vont être utilisés comme amers pour se localiser, et comme élément de corrélation pour évaluer son déplacement propre.
- On recherche des caractéristiques de points saillants, de lignes, de surfaces texturées visibles depuis autant de points de vue que possible

Comment ca marche: le filtrage

- Parmi ces points caractéristiques l'algorithme teste des points supposés fixes
- Il faut ensuite tester l'hypothèse « point fixe » d'éliminer des mobiles qui pourraient être éventuellement présents dans la scène.
- La plupart des algorithmes sont piégés par des plans se comportant comme des « miroirs » pour le type de capteur considéré.
 - Il peut être intéressant de doubler les capteurs pour discriminer les miroirs
 - Par exemple doubler un lidar par un sonar

Privilégier l'approche multi senseur

- Donc un conseil: Privilégiez l'approche multi senseur!
- Caméras
- Lidar
- Sonar ou radar pour la sécurité (confirmation)
- Une caméra thermique permet de discriminer facilement piétons, animaux et véhicules

Une caméra et un lidar sont basés sur l'optique. Si un élément de l'environnement fait miroir, ils seront piégés de la même façon. Le Sonar ou le radar qui n'utilisent pas les mêmes bandes de fréquence ne seront pas piégés.

« Nous conduisons avec deux yeux »

- Elon Musk a exigé de ses ingénieurs qu'ils se limitent à la vision
- L'entraînement des IA est insuffisant comparé à l'entraînement humain et nous n'avons pas encore les moyens de vérifier sa pertinence.
- D'où des accidents qu'un humain aurait probablement évité

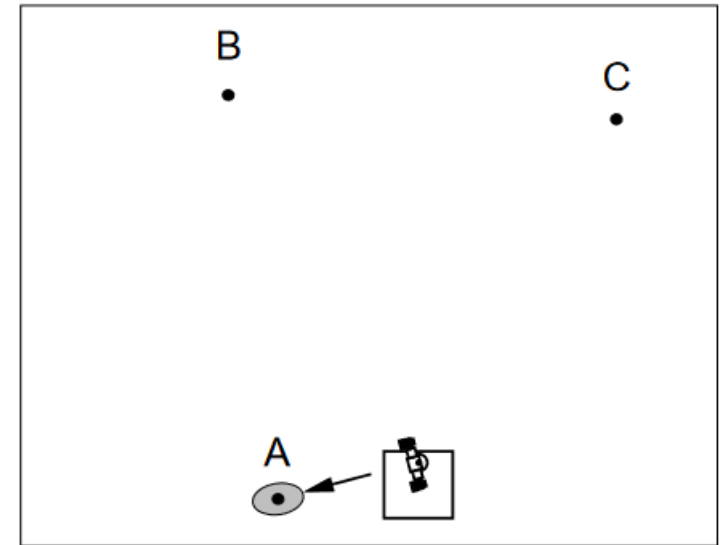
Le New York Times a publié un long article sur Tesla et le quotidien rapporte une situation inquiétante pour les initiatives en matière de conduite autonome du constructeur. Les deux journalistes ont interrogé 19 employés qui ont travaillé sur l'Autopilot et les conclusions publiées dans leur article accusent Elon Musk. Selon eux, le CEO aurait imposé une ligne directrice sans tenir compte de l'avis de ses équipes, ce qui expliquerait à la fois les retards apparents et surtout les limites de la conduite autonome chez Tesla.

Le dilemme du SLAM

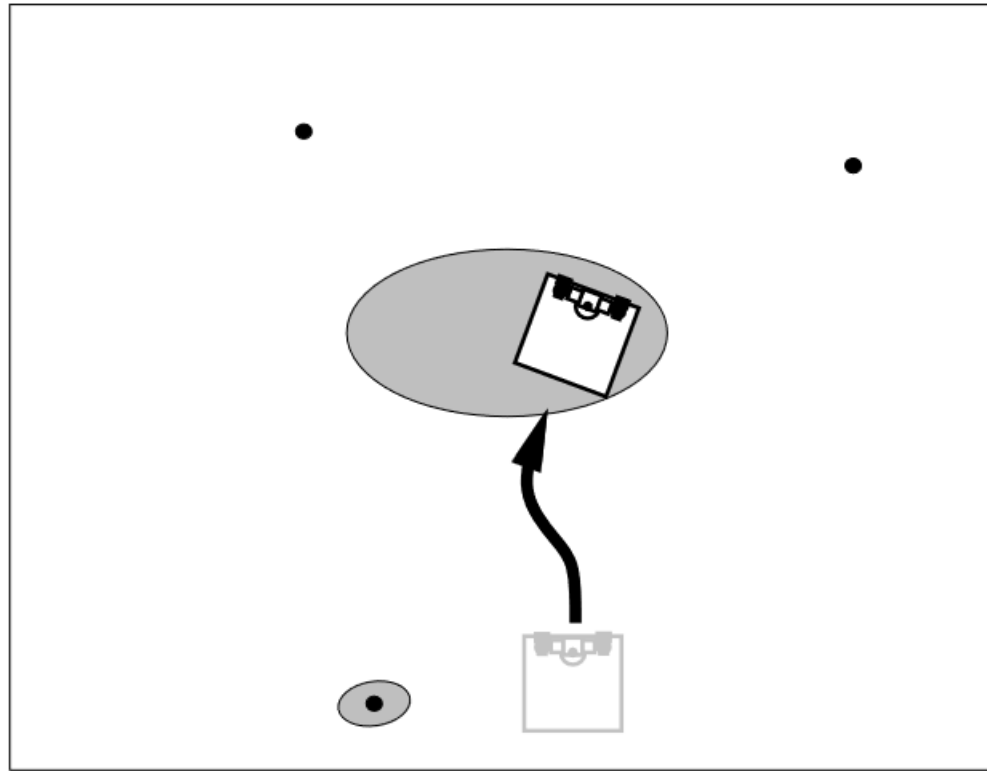
- Le SLAM semble être un problème de poule et d'œuf
- Pour sortir du dilemme nous supposons que le robot est la seule chose qui bouge.
- Hypothèse principale : le monde est statique.
- Nous éliminerons tous les points aberrants provenant d'objets mobiles autres que nous.
- Par la suite nous pourrions traiter le cas où il y a quelques objets mobiles dans un monde majoritairement statique.

Démarrage du robot

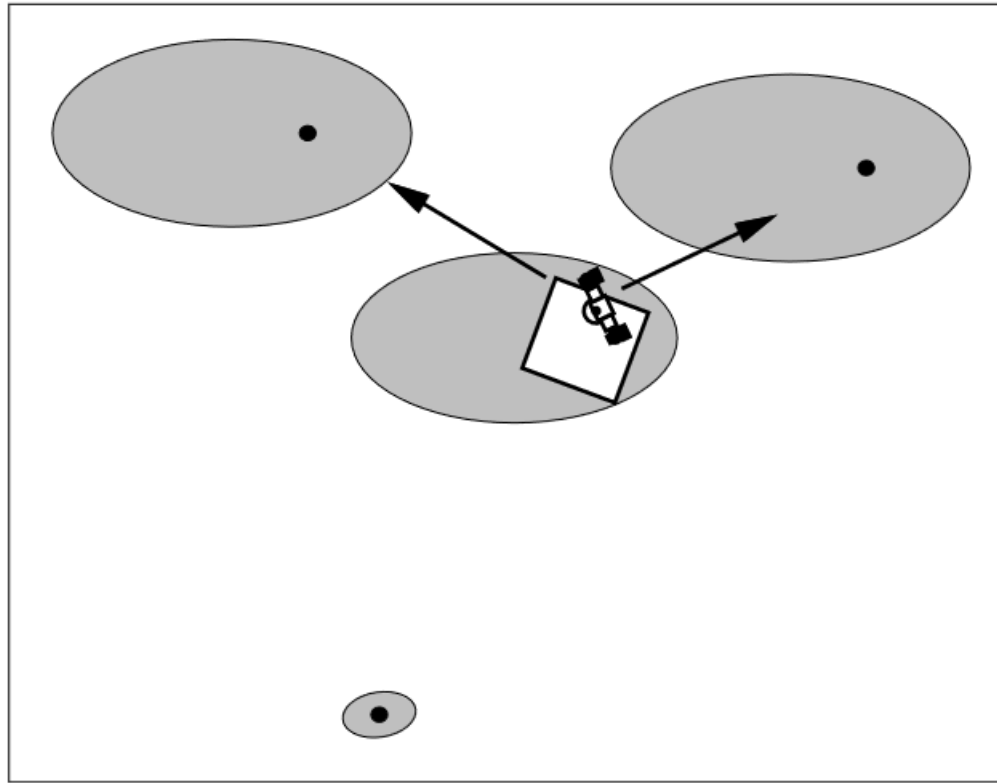
- Au départ (Le point de départ sert d'origine au repère) l'incertitude sur la position est nulle: on est au point de départ.



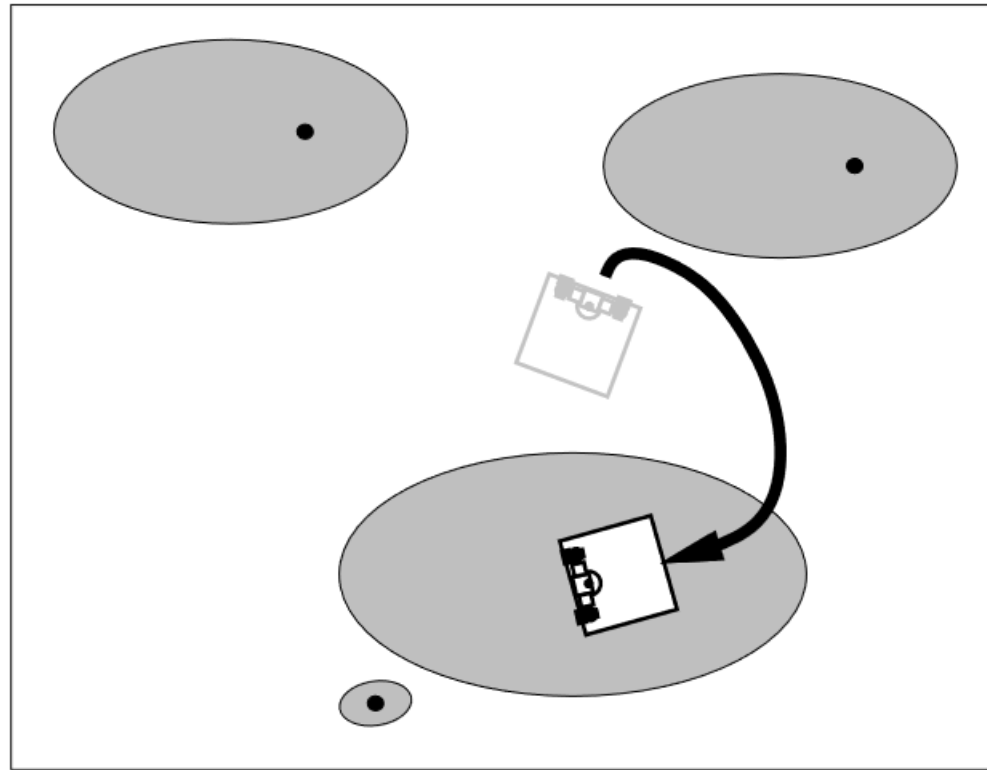
Le robot avance, l'incertitude s'accroît



Le robot voit maintenant B et C et fait sa première mesure

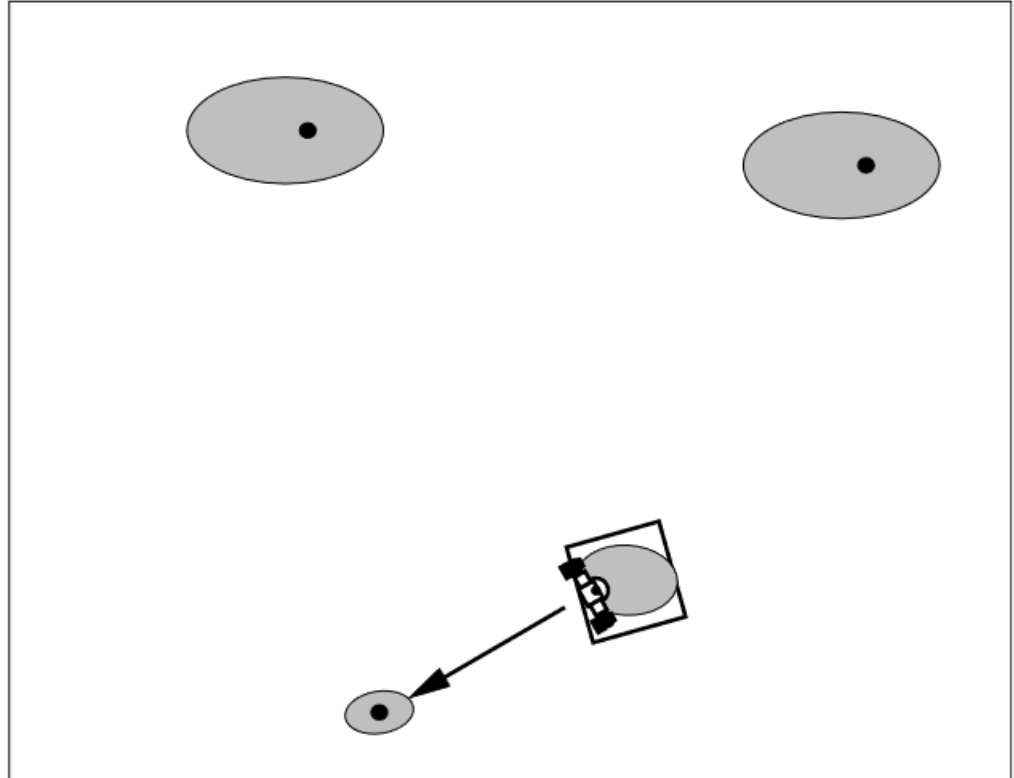


Le robot fait demi tour:
L'incertitude augmente



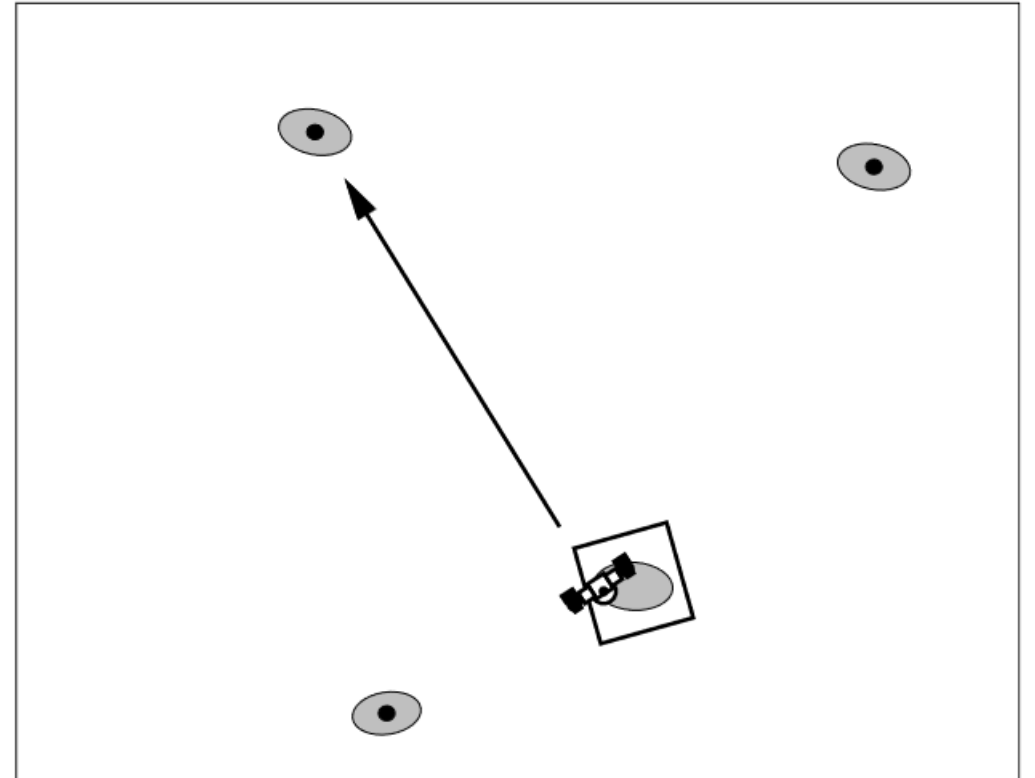
Le robot revoit le point A!

- Fermeture de la boucle!
- L'incertitude s'effondre!



Le robot re-mesure B

- L'incertitude se resserre, y compris pour le point C car les bases sont maintenant fixées



Propagation de l'incertitude au premier ordre

- x_v est l'état du robot, (x, y, θ) in 2D; y_i est le vecteur d'état du point de repère, (X, Y) en 2D
 - *A ma connaissance il n'existe pas d'algorithme SLAM prenant en compte l'orientation θ du point de repère.*

$$\hat{\mathbf{x}} = \begin{pmatrix} \hat{\mathbf{x}}_v \\ \hat{\mathbf{y}}_1 \\ \hat{\mathbf{y}}_2 \\ \vdots \end{pmatrix}, \quad \mathbf{P} = \begin{bmatrix} P_{xx} & P_{xy_1} & P_{xy_2} & \dots \\ P_{y_1x} & P_{y_1y_1} & P_{y_1y_2} & \dots \\ P_{y_2x} & P_{y_2y_1} & P_{y_2y_2} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

Modélisation de l'écart

- L'écart entre les paramètres du robot et de la carte est modélisé comme une seule quantité gaussienne à plusieurs variables et nous pouvons utiliser un filtre de Kalman étendu.
 - Vous aurez un cours sur les filtres de Kalmann

Le SLAM purement basé sur une métrique probabiliste est limité à de petits domaines à cause de:

- La mauvaise mise à l'échelle de calcul des filtres probabilistes.
- La croissance de l'incertitude à de grandes distances de l'origine de la carte qui rend la représentation de l'incertitude inexacte.
- L'association de données devient difficile en cas d'incertitude élevée, puis impossible au delà d'un certain seuil

Démonstrations

- Lidar environnement 2D simple
 - <https://www.youtube.com/watch?v=ZAESH7bu3IY>
- Lidar avec recalage sur boucle
 - <https://www.youtube.com/watch?v=Q1ipn42rMh8>
- Lidar multifaisceau
 - <https://www.youtube.com/watch?v=3faQktJNTWY>
- Lidar multifaisceau en environnement ouvert
 - <https://www.youtube.com/watch?v=otp63JOve2s>

Récréation

Dans cette série d'exercices, nous allons manipuler quelques calculs pratiques pour se familiariser avec les algorithmes de base

Exercice (on est hors cadre réglementaire)

- Une coupure de GPS a complètement désorienté un quadrucoptère qui se retrouve à survoler Brest.
 - Heureusement un système de secours basé sur une caméra panoramique est a bord du quadrucoptère.
 - L'enseigne du centre commercial casino « Le phare de l'iroise » est vu au gisement (pas de correction magnétique) 90
 - Le pilier EST du pont de recouvrance est vu à l'azimut 180
- Question 1: Ou est le drone?
- Question 2: Sous quel écart angulaire sont vues les piles du pont de recouvrance?

Le pont de recouvrance



Exercice 2 (même drone)

- On veut se repérer par rapport aux piles nord du pont de recouvrance
 - Ecart de pile à pile: 137 m
 - Hauteur d'une pile: 70m / mer
 - Pile Ouest: 48.384284, -4.497217
 - Pile Est: 48.384433, -4.495360
- On suppose qu'on va droit vers le pont donc cap au 180, perpendiculairement à l'axe du pont
- 1) Quelle est l'équation donnant la distance du pont en fonction de l'écart angulaire de vue des deux piles
- 2) Si on dispose d'une boussole et d'une correction nous donnant le cap avec précision. Comment pourrait-on utiliser l'image du pont vue par la caméra du drone pour se repérer dans Brest

Exercice 3 (piqueur de rappel)

- A partir des données de l'exercice 2, on veut vérifier la longueur du pont c'est-à-dire la cohérence entre les coordonnées géographiques et la longueur annoncée. On se méfie parce qu'on est en zone militaire et que les données sont brouillées.
- Vous devez:
 - Chercher sur internet le rayon terrestre à la latitude de Brest
 - Calculer la précision de calcul à partir du rayon terrestre et de données
 - Calculer le cosinus de la latitude pour corriger la longitude
 - Calculer la distance à partir des coordonnées obtenues

Exercice 4 (Même drone)

- Notre caméra de navigation (navcam) a un angle de vision de 90° et une résolution UHD de $3\,840 \times 2\,160$ pixels.
- Elle est parfaitement alignée avec l'axe du drone
- La centrale à inertie indique une inclinaison nulle du drone et un cap au 180.
- On se repère par rapport aux piles nord du pont de recouvrance
- La pile EST est aux coordonnées (1022,587)
- La pile OUEST est aux coordonnées (3127,586)

Exercice 4 (suite)

- Est-on plus haut ou plus bas que le sommet du pont (70 m)
- A quelle distance du pont est-on?
 - Si vous êtes en avance=> Ou est-on?
- A quelle altitude est-on?

Cartographie a grande échelle

Des solutions modernes pratiques à la cartographie à grande échelle suivent une approche métrique et topologique

A partir de quels éléments:

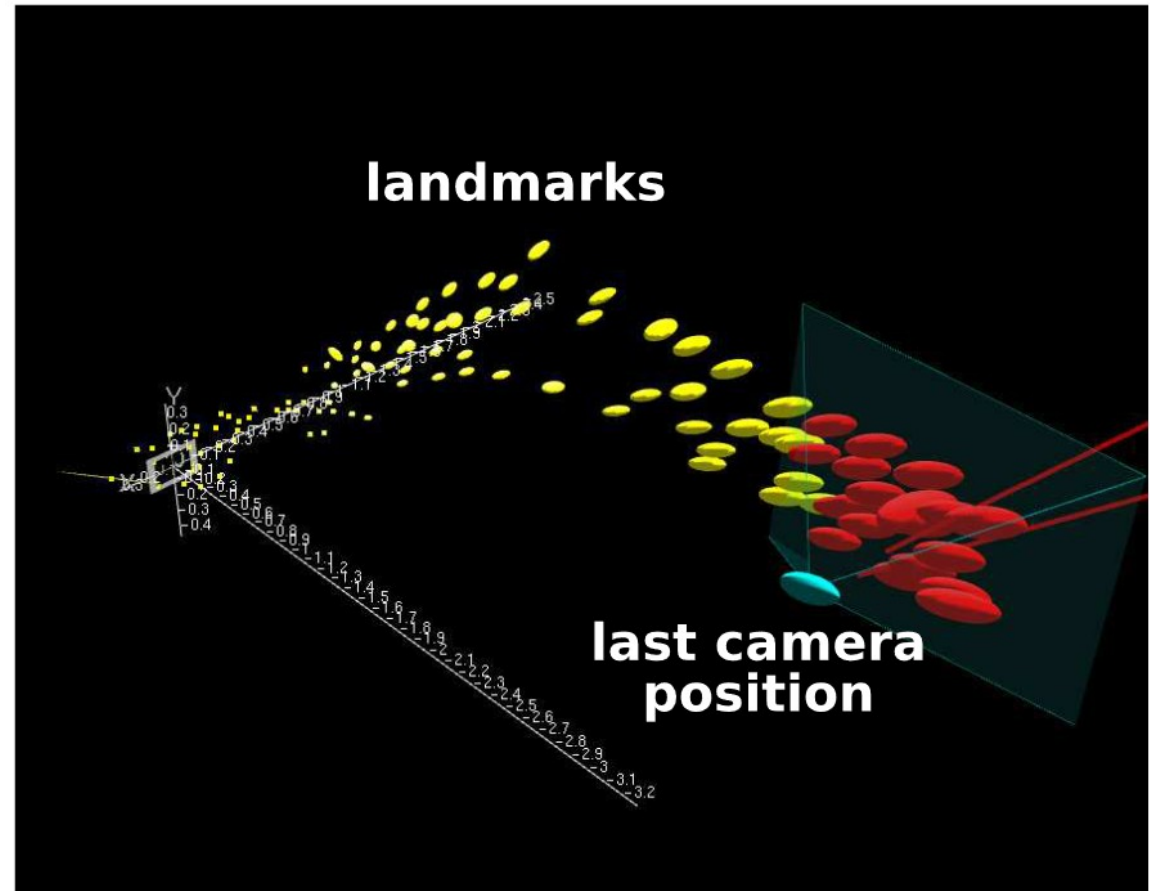
- Cartographie métrique locale pour estimer la trajectoire et éventuellement établir une carte locale.
 - *Ca on sait faire, c'est ce que l'on vient de voir.*
- Reconnaissance de lieu, pour effectuer une « fermeture de boucle » ou relocaliser le robot lorsqu'il est perdu.
 - Au lieu de « subir » la trajectoire du robot, nous allons la programmer pour donner une priorité à la cartographie.
- Optimisation/relaxation de la carte pour optimiser une carte lorsque des boucles sont fermées.
 - Algorithmes « élastic matching », « recuit simulé »

En résumé

- On va partir de ce qu'on sait bien faire: de petites cartes locales à partir d'une métrique locale
- On va assembler le puzzle pour obtenir une carte globale
- On va jouer sur la trajectoire du robot pour créer des « boucles » qui vont permettre de fermer les cartes.

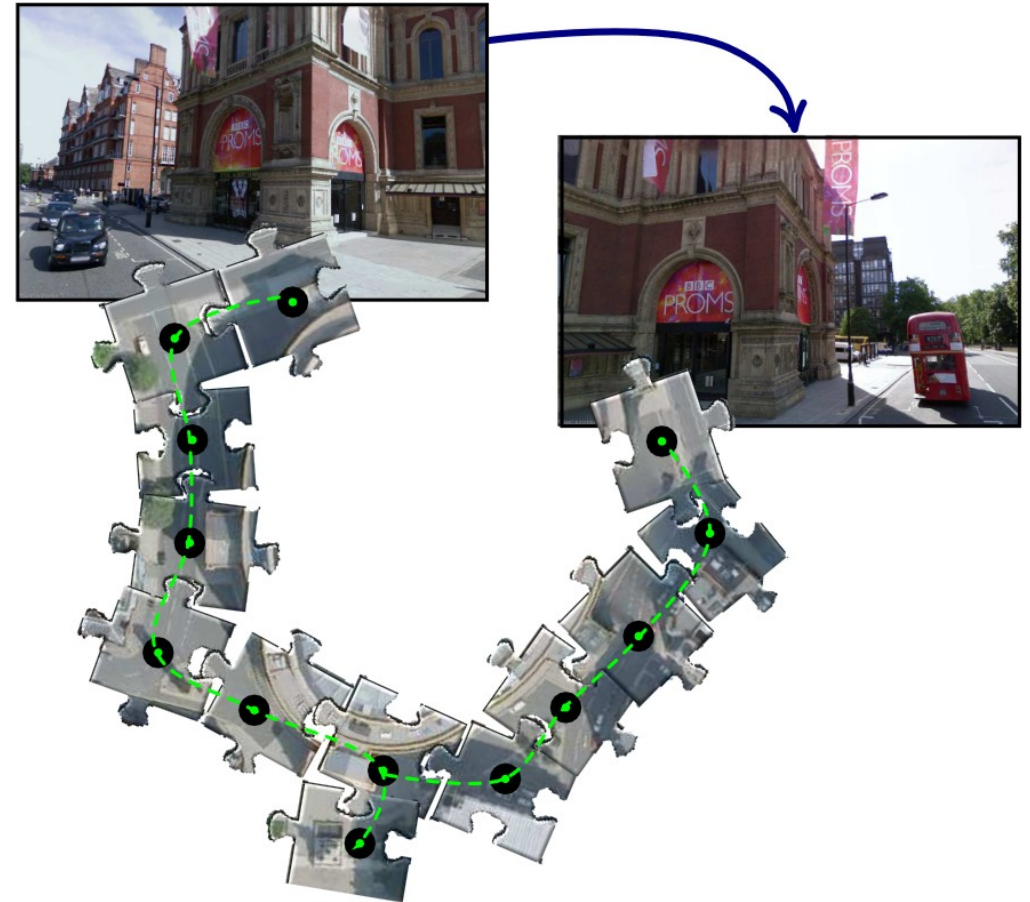
On va partir de ce qu'on sait bien faire: de petites cartes locales à partir d'une métrique locale

Métrique Locale



On va assembler le puzzle pour obtenir une carte globale

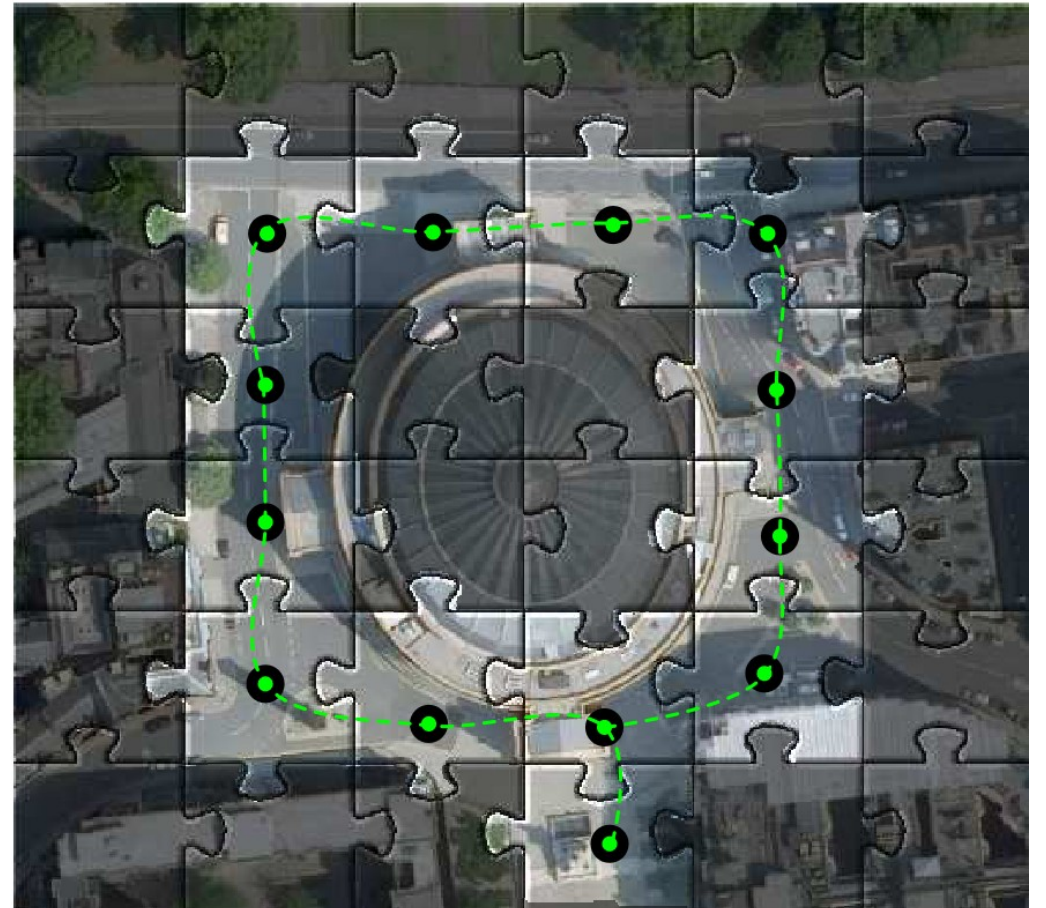
Reconnaissance des emplacements



On va jouer sur la trajectoire du robot pour créer des « boucles » de fermeture

Optimisation globale

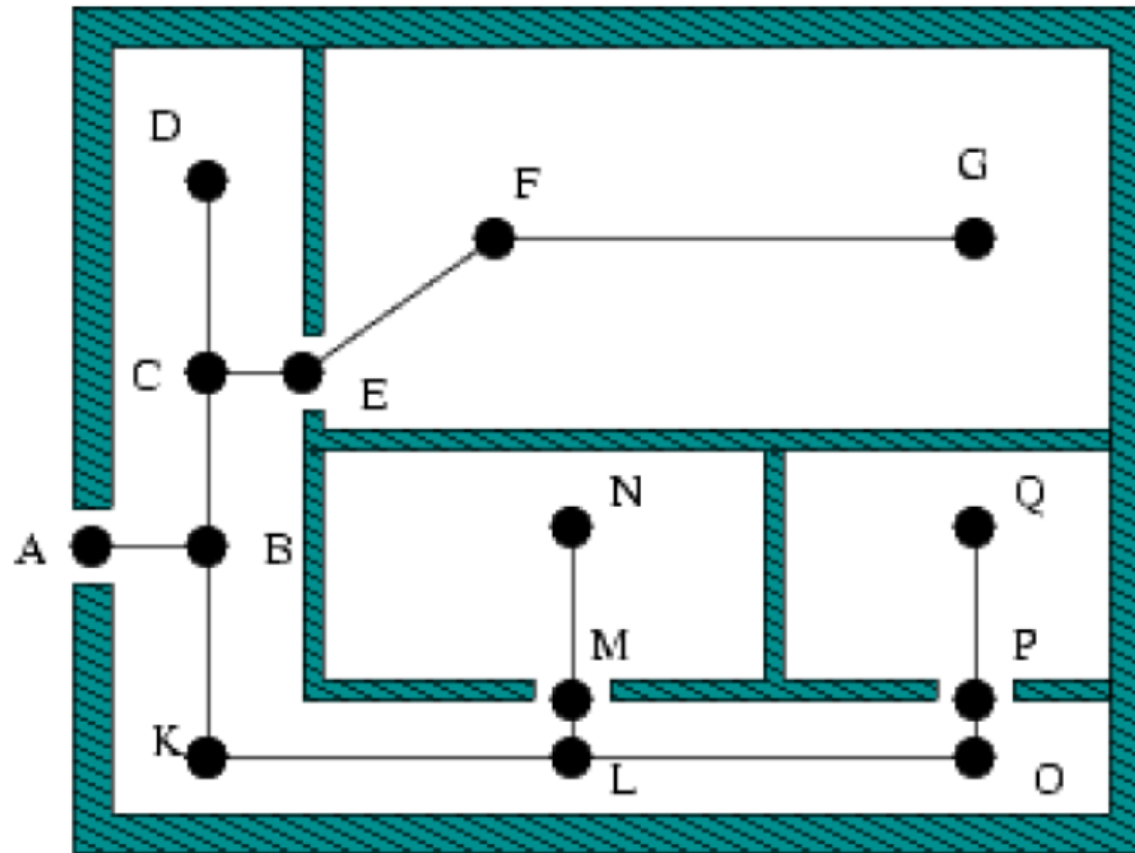
Quand on observe une carte en cours de construction, cette action de l'algorithme est très reconnaissable. On « redresse » et on assemble les pièces qui semblent « sauter » d'un coup



SLAM topologique

- C'est une représentation basée sur un graphe
- Segmentation de l'environnement entre des places pré-définies
- Parfaitement adaptée pour une préparation de mission symbolique et pour la navigation
- Nous avons étudié en recherche (non encore publié) des algorithmes de « routage topologique » adaptés au routage maritime
- Dans le domaine routier, nos GPS utilisent ce type de routage mais sur des cartes figées. Certaines plates formes en ligne comme « Waze » commencent à ajouter des éléments dynamiques

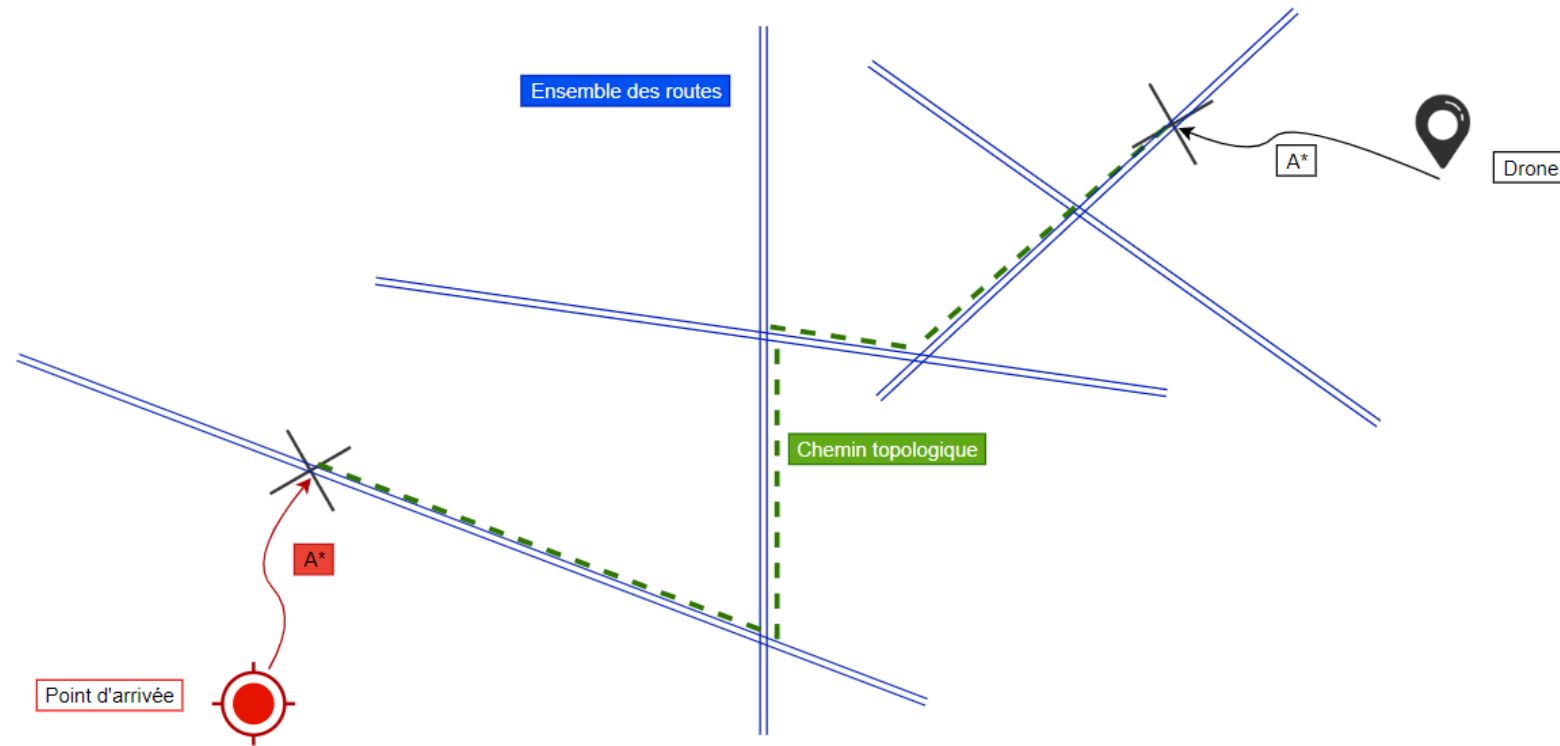
Exemple de représentation



Modèle d'environnement

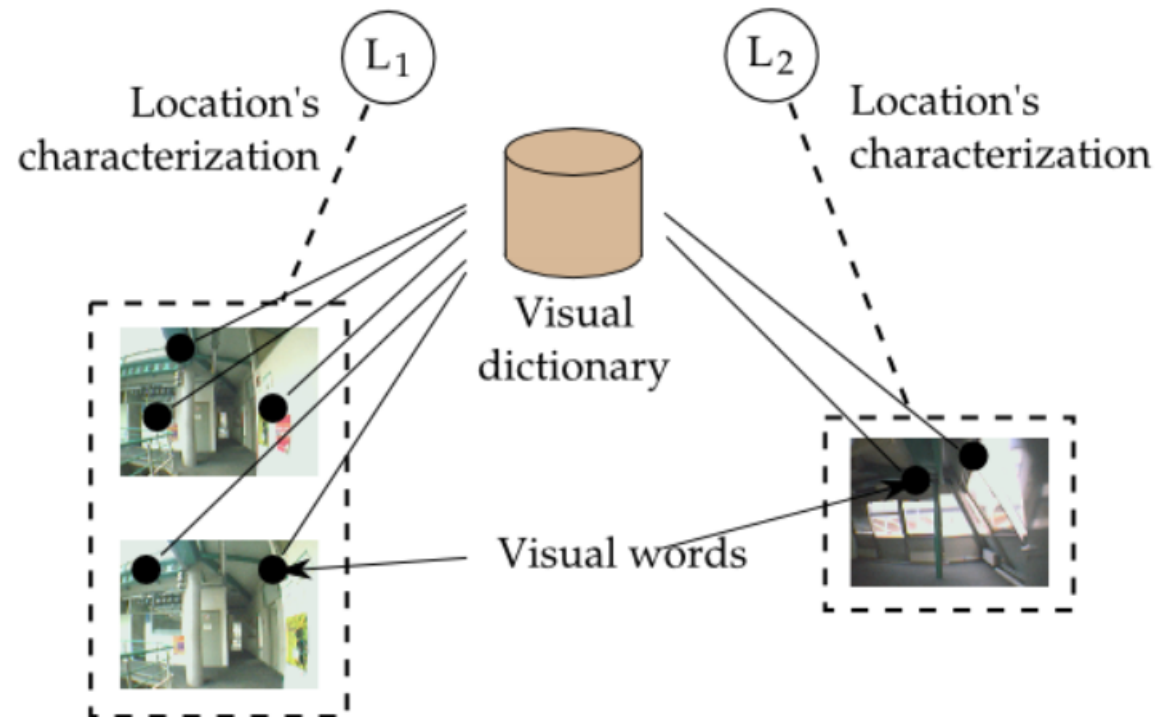
- La carte est définie comme un graphe des emplacements connectés par des arêtes
- Les arêtes modélisent les relations entre les emplacements (par exemple, la traversabilité, la similarité, des règles à suivre (marée, limitation de vitesse, interdiction a certaines catégories de véhicules...))
- Dans un travail de recherche réalisée à l'ISEN nous avons étudié un graphe constitué uniquement « d'arêtes » en relation

Graphe d'arêtes en relation

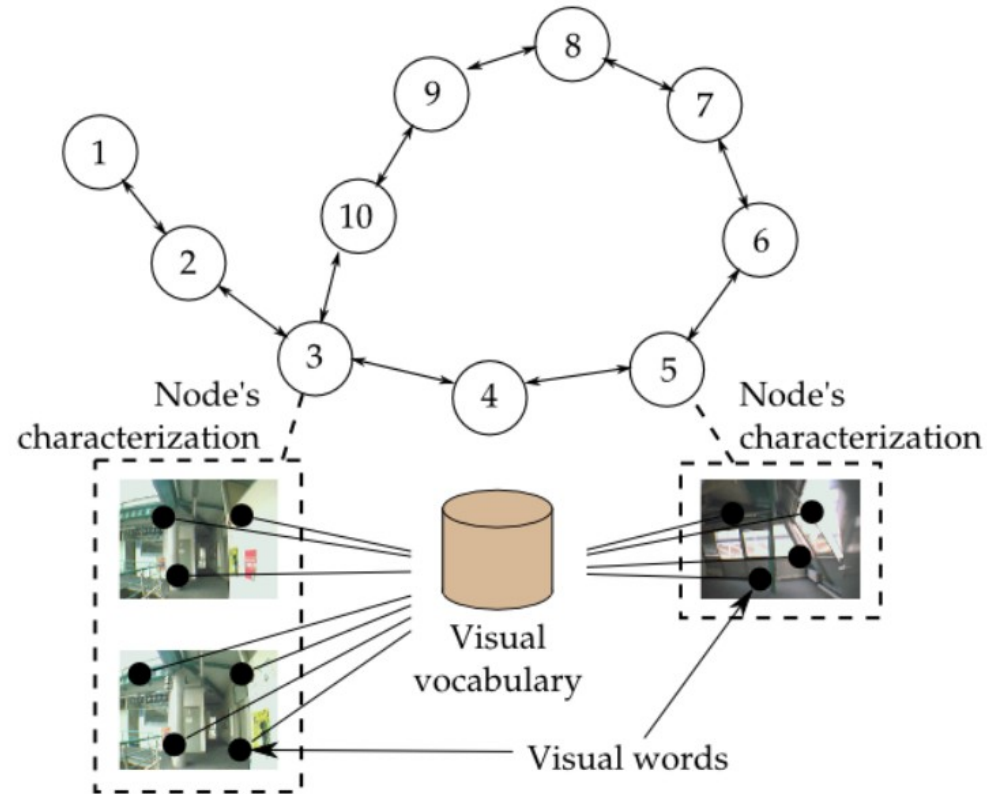


Julien REY
Jean-Philippe BRUNET

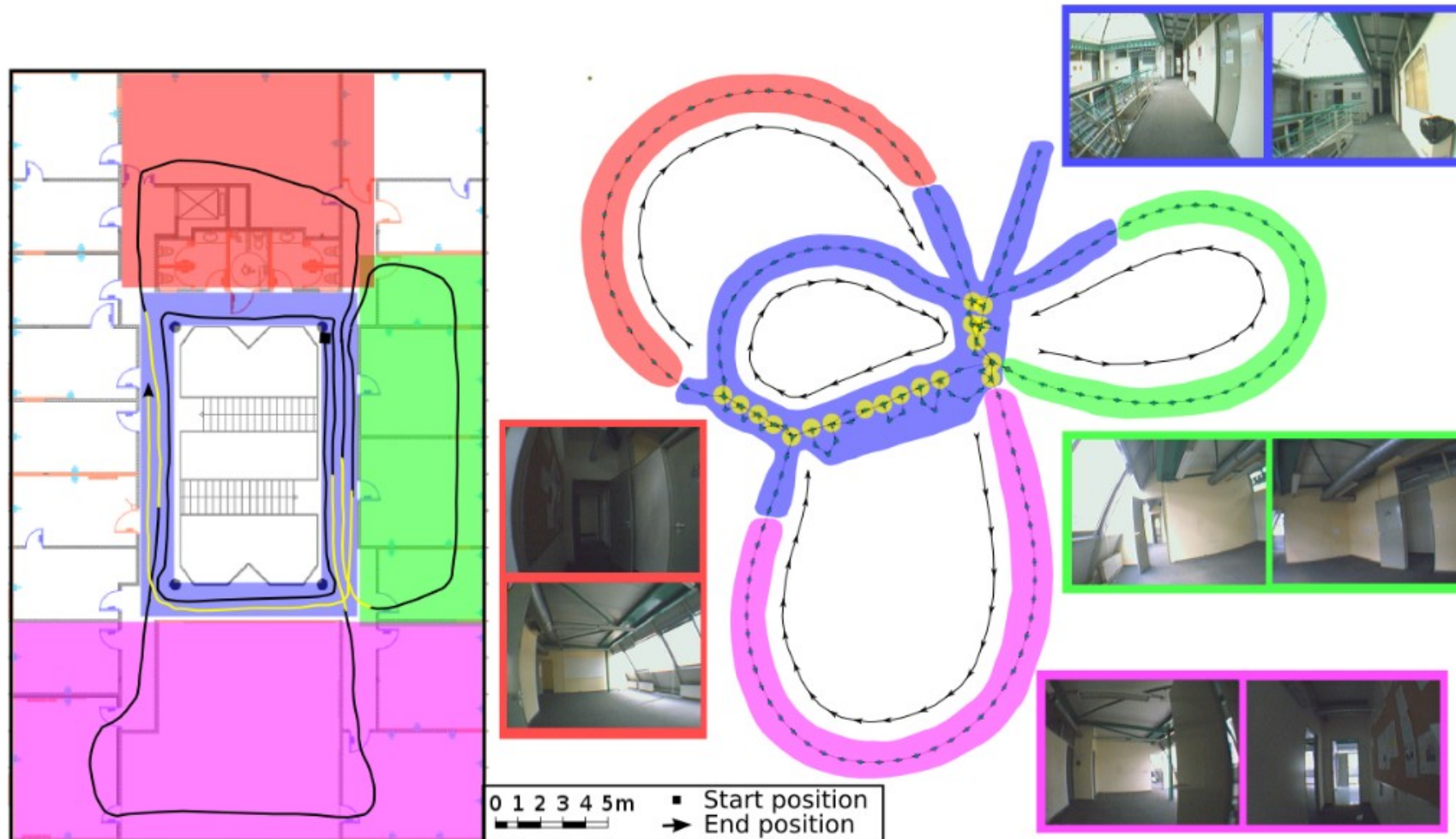
Caractérisation des emplacements



Caractérisation des nœuds



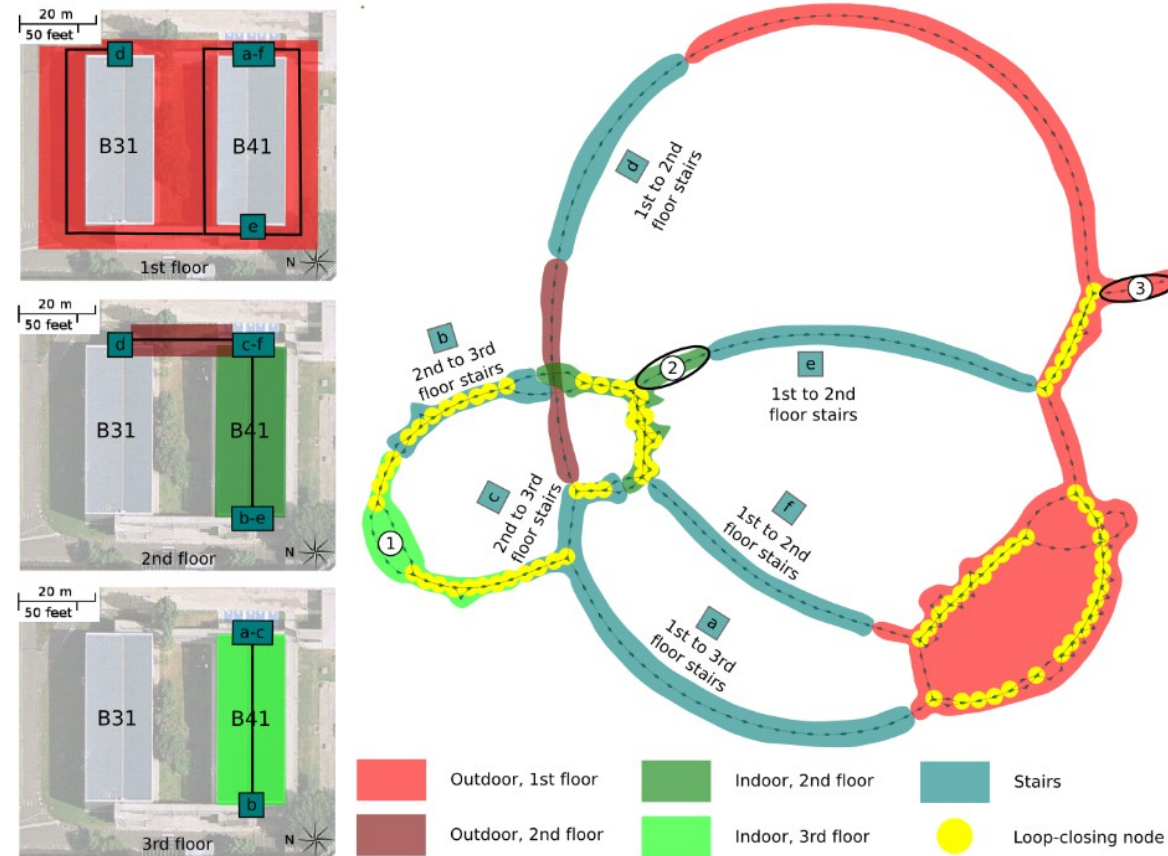
Carte topologique d'un bâtiment



Computing imperial
college. London

Carte intérieur/extérieur avec étages

Computing imperial
college. London

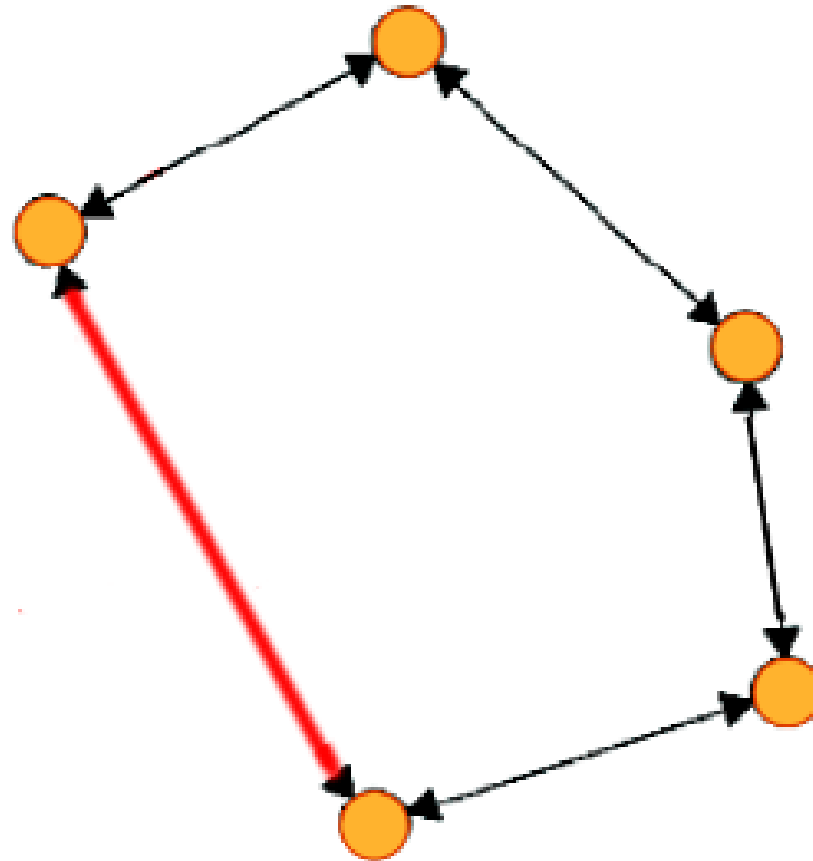


Ajouter une information métrique aux arêtes

- On profite des mesures d'odométrie d'un robot à roues pour ajouter des informations de déplacement relatif entre les nœuds.
- On applique un algorithme simple de relaxation de graphe pour calculer précisément les positions absolues pour les nœuds
- Remarque: en 3d spatiale (drones aériens) on a deux problèmes:
 - L'odométrie n'est pas précise (ce qu'on peut résoudre avec un GPS et une radiosonde)
 - La fermeture de boucle en 3 D est beaucoup plus complexe.

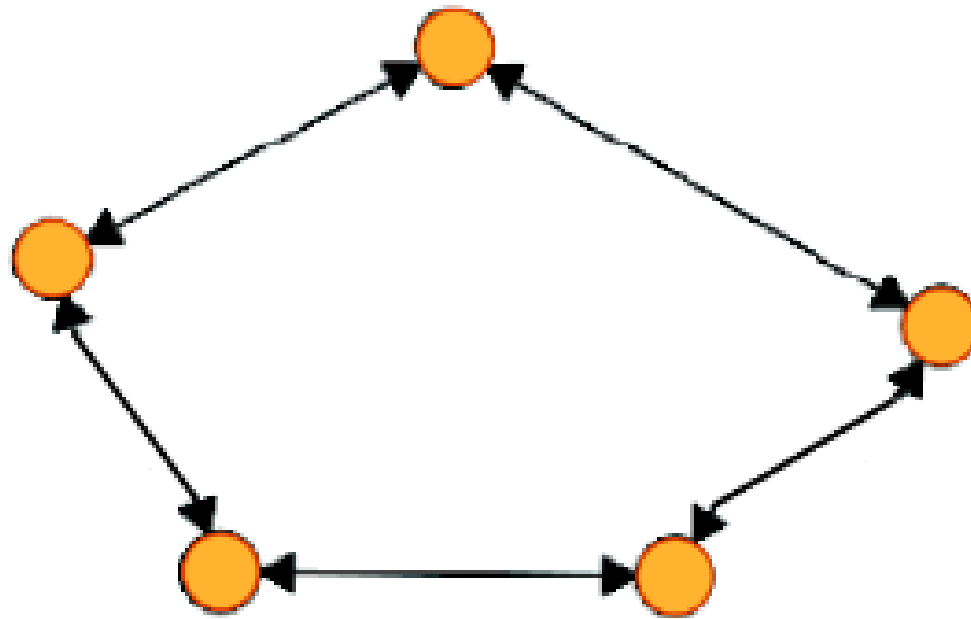
Détection d'une fermeture de boucle

- On ajoute la nouvelle contrainte au graphe



On applique un algorithme de « relaxation » (relâchements des contraintes)

- Ajouter cette nouvelle contrainte produit un graphe beaucoup plus précis



Exemple d'algorithme de relaxation

(Dukett, 2000)

- Estimer la position et la variance du nœud i par rapport a chacun de ses nœuds voisins j

$$(x'_i)_j = x_j + d_{ji} \cos(\theta_{ji} + \theta_j) \quad (y'_i)_j = y_j + d_{ji} \sin(\theta_{ji} + \theta_j) \quad (\theta'_i)_j = \theta_j + \varphi_{ji} \quad (1)$$

$$(v'_i)_j = v_j + v_{ji} \quad (2)$$

Relaxation (2)

- Estimer la variance du nœud i en utilisant la moyenne harmonique des estimations de ses voisins
- n_i est le nombre de voisins du noeud i

$$v_i = \frac{n_i}{\sum_j \frac{1}{(v'_i)_j}} \quad (3)$$

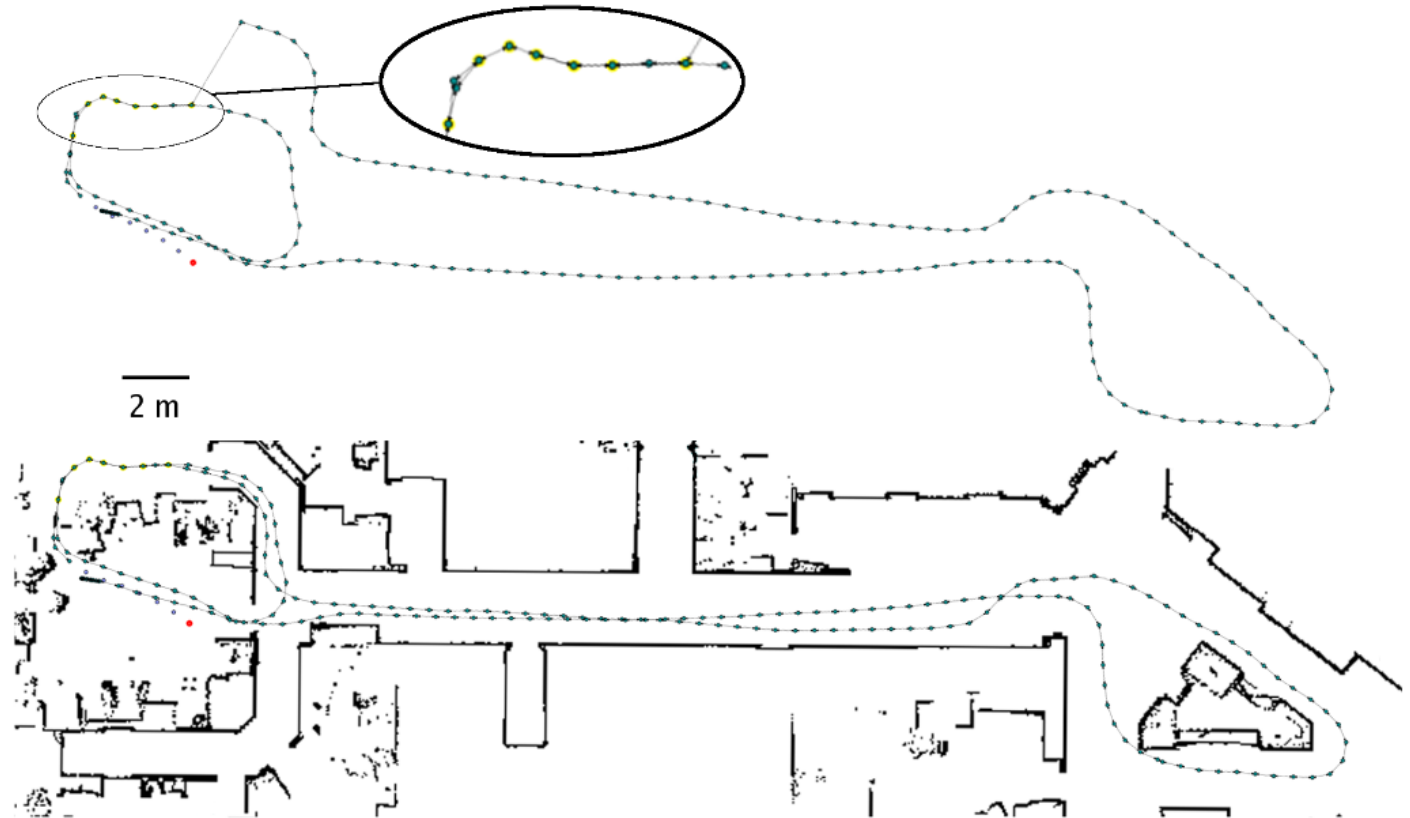
Relaxation (3)

- Estimer la position du lieu i en tant que moyenne des estimations de ses voisins:

$$x_i = \frac{1}{n_i} \sum_j \frac{(x'_i)_j v_i}{(v'_i)_j} \quad y_i = \frac{1}{n_i} \sum_j \frac{(y'_i)_j v_i}{(v'_i)_j} \quad \theta_i = \arctan \left(\frac{\sum_j \frac{\sin((\theta'_i)_j)}{(v'_i)_j}}{\sum_j \frac{\cos((\theta'_i)_j)}{(v'_i)_j}} \right) \quad (4)$$

Et voilà le travail:

- Relaxation
- Bonne Odométrie
- Fermeture de boucle



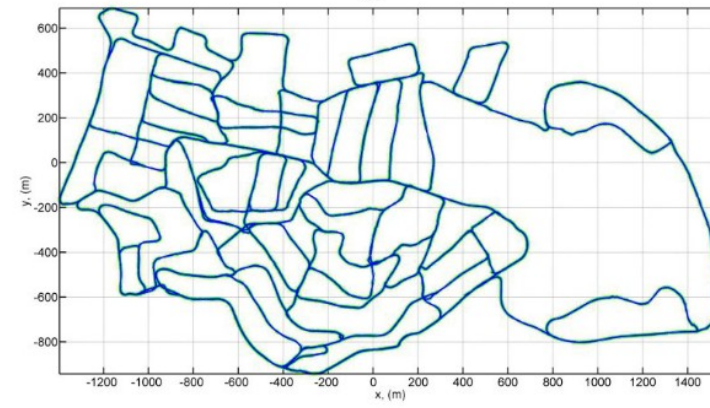
Algorithme simple a grande échelle: RATSLAM

- Milford and Wyeth, 2007.
<http://www.youtube.com/watch?v=-0XSUi69Yvs>
- Une « odométrie visuelle » très simple donne une trajectoire approximative.
- La reconnaissance visuelle simple des lieux permet de nombreuses fermetures de boucles.
- Relaxation/optimisation de la carte pour construire une carte globale.

RATSLAM



(a)



(b)