Contributions to Localization, Mapping and Navigation in Mobile Robotics

PhD dissertation

Jose Luis Blanco Claraco

Advisors: J. González, J.A. Fernández-Madrigal

UNIVERSIDAD DE MÁLAGA

University of Málaga

November 13th, 2009

Thesis motivation

Robotics

Robotics is a success in the industry (millions of robotic arms installed in the world)

Thesis motivation

Mobile Robotics

Why hasn't mobile robotics enjoyed such a success yet?

→ A mobile robot should be autonomous (quite complex!)

Images: http://singularityhub.com/, http://lemonodor.com/, http://www.robots-dreams.com/

Thesis motivation

Requirements for an autonomous robot

Among others:

- Move without colliding.
- Keep track of its position in the world.
- Be able to model its environment by itself.
- Solve complex plans.
- Reason in an uncertain world.
- ...

Thesis motivation

Requirements for an autonomous robot

Among others:

- Move without colliding.
- Keep track of its position in the world.
- Be able to model its environment by itself.
- Solve complex plans.
- Reason in an uncertain world.
- ...

In this thesis

- Localization.
- Mapping.
- Navigation.

Thesis outline

This talk is organized into four parts, each having contributions about a different ability of mobile robots:

1. Localization

An omnipresent problem: probabilistic filtering

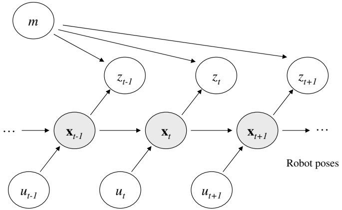
Given a map of an environment, how can a robot localize itself robustly using its sensors?

An omnipresent problem: probabilistic filtering

Given a map of an environment, how can a robot localize itself robustly using its sensors?
Bayesian probabilistic filtering: an extremely versatile framework for estimating variables from noisy observations.

An omnipresent problem: probabilistic filtering

Given a map of an environment, how can a robot localize itself robustly using its sensors?
Bayesian probabilistic filtering: an extremely versatile framework for estimating variables from noisy observations.



- m: The map.
- x_t: The robot poses.
- z_t: The observations.
- u_t: The robot actions.

The problem: probabilistic filtering

$$\underbrace{p(x_t | z_{1:t}, u_{1:t}, m)}_{\text{Posterior}}$$

The problem: probabilistic filtering

$$\underbrace{p(x_t|z_{1:t}, u_{1:t}, m)}_{\text{Posterior}} \propto \underbrace{p(z_t|x_t, m)}_{\text{Observation likelihood}} \times$$

$$\int \underbrace{p(x_t|x_{t-1}, u_t)}_{\text{Transition model}} \underbrace{p(x_{t-1}|z_{1:t-1}, u_{1:t-1}, m)}_{\text{Last posterior}} dx_{t-1}$$

The problem: probabilistic filtering

$$\underbrace{p(x_t|z_{1:t},u_{1:t},m)}_{\text{Posterior}} \propto \underbrace{p(z_t|x_t,m)}_{\text{Observation likelihood}} \times$$

$$\int \underbrace{p(x_t|x_{t-1},u_t)}_{\text{Transition model}} \underbrace{p(x_{t-1}|z_{1:t-1},u_{1:t-1},m)}_{\text{Last posterior}} dx_{t-1}$$

Part I of this thesis is completely built around this equation!

- **As a whole:** A new optimal filter.

The problem: probabilistic filtering

$$\underbrace{p(x_t|z_{1:t}, u_{1:t}, m)}_{\text{Posterior}} \propto \underbrace{p(z_t|x_t, m)}_{\text{Observation likelihood}} \times$$

$$\int \underbrace{p(x_t|x_{t-1}, u_t)}_{\text{Transition model}} \underbrace{p(x_{t-1}|z_{1:t-1}, u_{1:t-1}, m)}_{\text{Last posterior}} dx_{t-1}$$

Part I of this thesis is completely built around this equation!

- **As a whole:** A new optimal filter.
- **Observation model:** Consensus-based likelihood.

The problem: probabilistic filtering

$$\underbrace{p(x_t|z_{1:t}, u_{1:t}, m)}_{\text{Posterior}} \propto \underbrace{p(z_t|x_t, m)}_{\text{Observation likelihood}} \times$$

$$\int \underbrace{p(x_t|x_{t-1}, u_t)}_{\text{Transition model}} \underbrace{p(x_{t-1}|z_{1:t-1}, u_{1:t-1}, m)}_{\text{Last posterior}} dx_{t-1}$$

Part I of this thesis is completely built around this equation!

- **As a whole:** A new optimal filter.
- **Observation model:** Consensus-based likelihood.
- **Transition model:** IMU + odometry fusion.

Thesis outline

This talk is organized into four parts:

1. Localization

Thesis outline

This talk is organized into four parts:

1. Localization
2. SLAM
3. Large-scale SLAM
4. Navigation

Part I

Mobile Robot Localization

1. Optimal particle filtering: applications to robot localization
2. A Consensus-based observation likelihood
3. Fusion of odometry and an IMU

Outline of Part I

A classification of Bayesian filters

Proposal distribution	**System models**	**Algorithms**
–	Linear Gaussian	
–	Non-Linear Gaussian	
Standard	Non-Linear Non-Gaussian	
Optimal	Non-Linear Gaussian	
Optimal	Non-Linear Non-Gaussian	

A classification of Bayesian filters

Proposal distribution	**System models**	**Algorithms**
–	Linear Gaussian	Kalman Filter
–	Non-Linear Gaussian	EKF, UKF
Standard	Non-Linear Non-Gaussian	
Optimal	Non-Linear Gaussian	
Optimal	Non-Linear Non-Gaussian	

A classification of Bayesian filters

Proposal distribution	**System models**	**Algorithms**
–	Linear Gaussian	Kalman Filter
–	Non-Linear Gaussian	EKF, UKF
Standard	Non-Linear Non-Gaussian	SIR, Aux. PF, FastSLAM
Optimal	Non-Linear Gaussian	
Optimal	Non-Linear Non-Gaussian	

A classification of Bayesian filters

Proposal distribution	**System models**	**Algorithms**
–	Linear Gaussian	Kalman Filter
–	Non-Linear Gaussian	EKF, UKF
Standard	Non-Linear Non-Gaussian	SIR, Aux. PF, FastSLAM
Optimal	Non-Linear Gaussian	FastSLAM 2.0, Grisetti *et al.*
Optimal	Non-Linear Non-Gaussian	

A classification of Bayesian filters

Proposal distribution	**System models**	**Algorithms**
–	Linear Gaussian	Kalman Filter
–	Non-Linear Gaussian	EKF, UKF
Standard	Non-Linear Non-Gaussian	SIR, Aux. PF, FastSLAM
Optimal	Non-Linear Gaussian	FastSLAM 2.0, Grisetti *et al.*
Optimal	Non-Linear Non-Gaussian	**Optimal PF** **(In this thesis)**

The proposed filter

Characteristics of the proposed method:

- A particle filter based on the optimal proposal [Doucet 2000].
- Can deal with non-parameterized observation models, using rejection sampling to approximate the actual densities.
- Integrated KLD-sampling [Fox 2003] for a dynamic sample size (Optional).
- The weights of all the samples are always equal.

The proposed filter

The key for an efficient particle filter is using the optimal proposal distribution [Doucet2000]:

$$x_t^{[i]} \sim q(x_t | x^{t-1,[i]}, z^t, u^t) = p(x_t | x^{t-1,[i]}, z^t, u^t)$$

The proposed filter

The key for an efficient particle filter is using the optimal proposal distribution [Doucet2000]:

$$x_t^{[i]} \sim q(x_t | x^{t-1,[i]}, z^t, u^t) = p(x_t | x^{t-1,[i]}, z^t, u^t)$$

In localization with grid maps, there is **no closed-form expression** for this term.

The proposed filter

The key for an efficient particle filter is using the optimal proposal distribution [Doucet2000]:

$$x_t^{[i]} \sim q(x_t | x^{t-1,[i]}, z^t, u^t) = p(x_t | x^{t-1,[i]}, z^t, u^t)$$

This term can be expanded using the Bayes rule as:

$$q(x_t | x^{t-1,[i]}, z^t, u^t) = \frac{p(z_t | x_t, x^{t-1,[i]}, z^{t-1}, u^t) p(x_t | x^{t-1,[i]}, z^{t-1}, u^t)}{p(z_t | x^{t-1,[i]}, z^{t-1}, u^t)}$$

The proposed filter

The key for an efficient particle filter is using the optimal proposal distribution [Doucet2000]:

$$x_t^{[i]} \sim q(x_t | x^{t-1,[i]}, z^t, u^t) = p(x_t | x^{t-1,[i]}, z^t, u^t)$$

This term can be expanded using the Bayes rule as:

$$q(x_t | x^{t-1,[i]}, z^t, u^t) = \frac{p(z_t | x_t, x^{t-1,[i]}, z^{t-1}, u^t) p(x_t | x^{t-1,[i]}, z^{t-1}, u^t)}{p(z_t | x^{t-1,[i]}, z^{t-1}, u^t)}$$

The denominator (new weight) does **not** depend on x_t!!

Comparison to Sequential Importance Resampling (SIR)

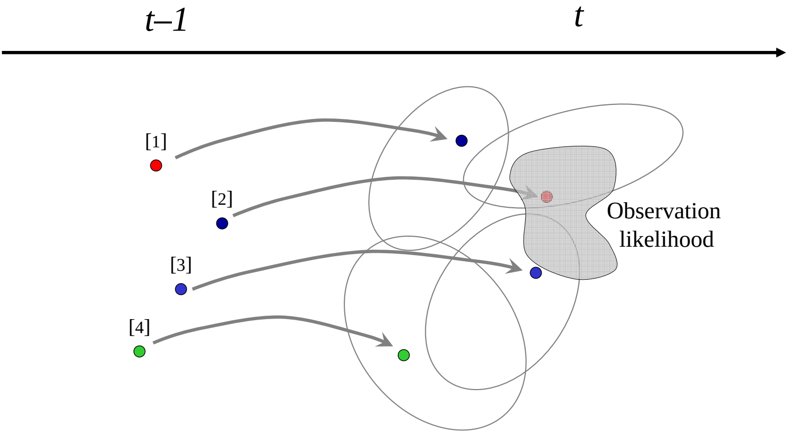
Method: SIR

1 particle $\longrightarrow$ 1 particle (With the standard proposal).

Comparison to Sequential Importance Resampling (SIR)

Method: SIR

1 particle $\rightarrow$ 1 particle (With the standard proposal).



Comparison to the Auxiliary PF

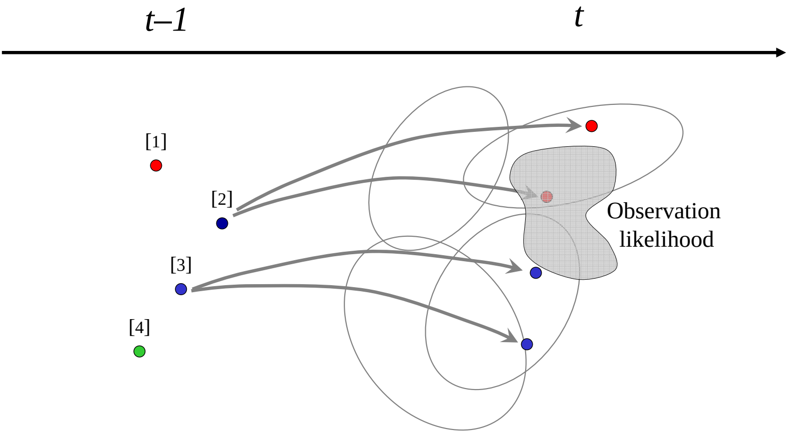
Method: APF

1 particle $\rightarrow$ N particles (With the standard proposal).

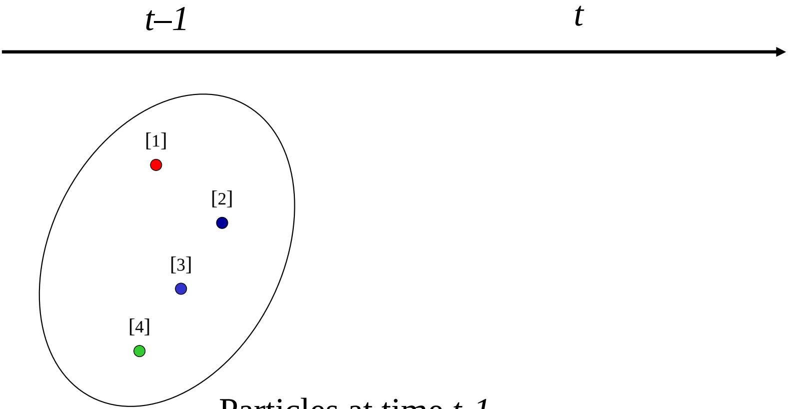
Comparison to the Auxiliary PF

Method: APF

1 particle $\longrightarrow$ N particles (With the standard proposal).

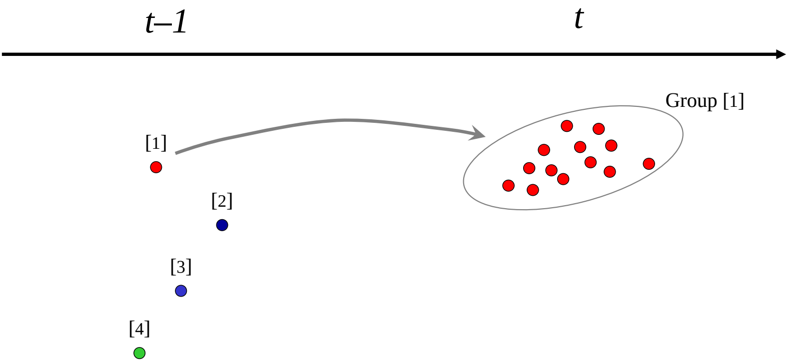


How does our method work?



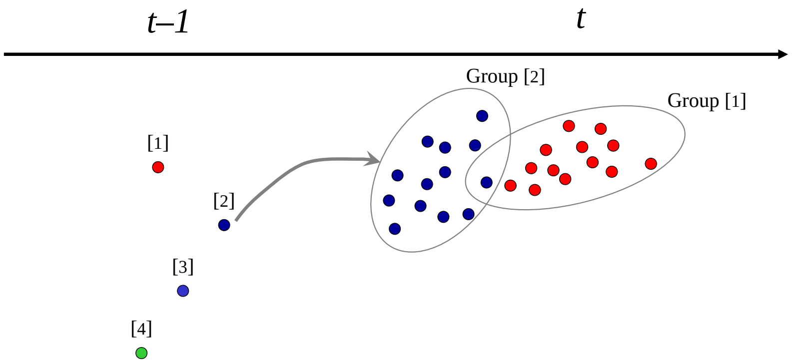
Particles at time *t-1*

How does our method work?



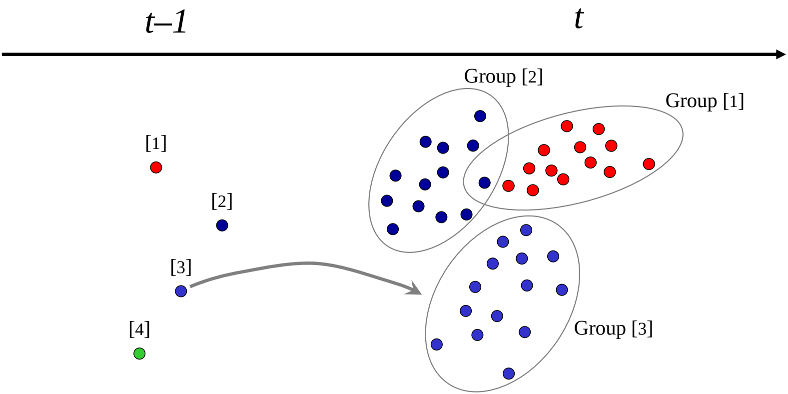
Each particle propagates in time probabilistically: this is the reason of the *duplication*.

How does our method work?



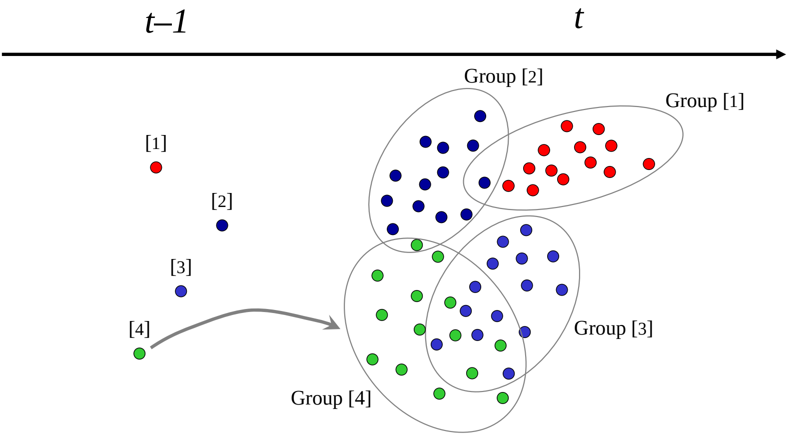
Each particle propagates in time probabilistically: this is the reason of the *duplication*.

How does our method work?



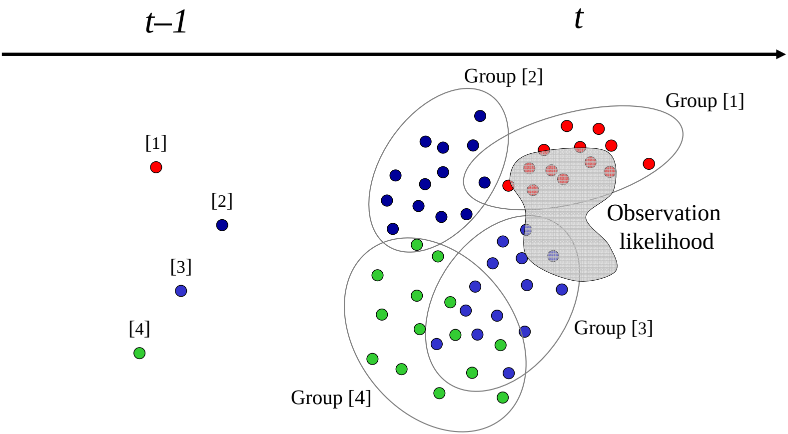
Each particle propagates in time probabilistically:
this is the reason of the *duplication*.

How does our method work?



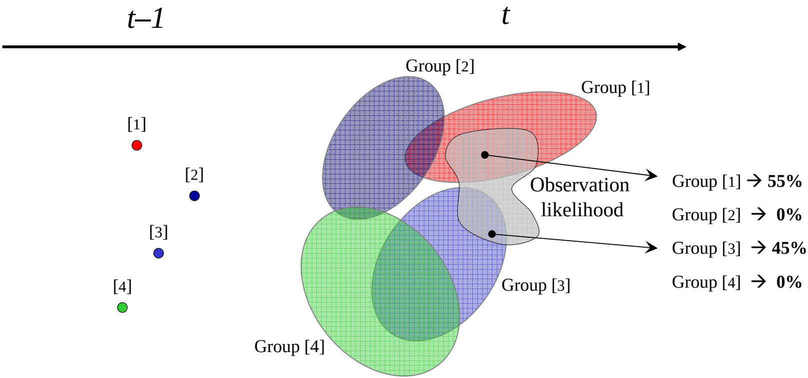
Each particle propagates in time probabilistically: this is the reason of the *duplication*.

How does our method work?



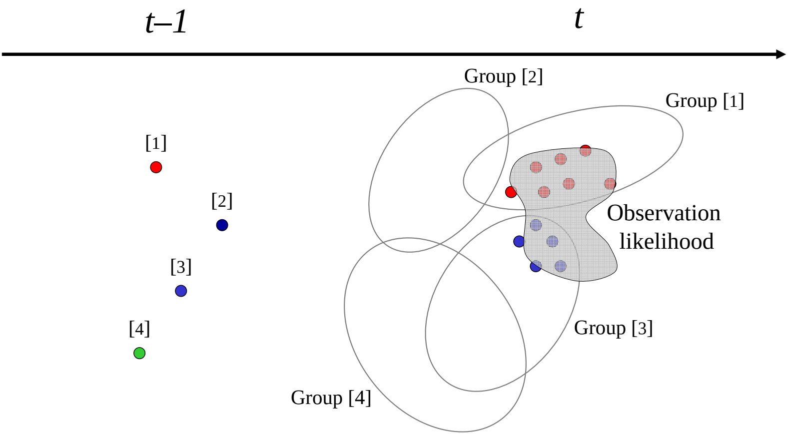
The observation likelihood states which particles are really important.

How does our method work?



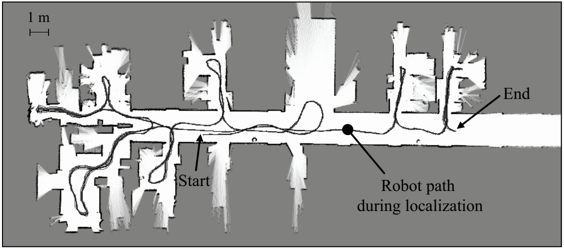
We can predict which groups will be more important, before really generating the new samples!

How does our method work?

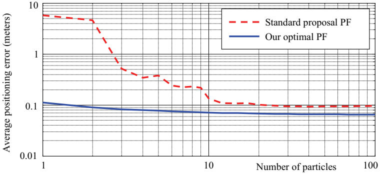


Particles are drawn according to the optimal proposal, only for those groups that really contribute to the posterior.

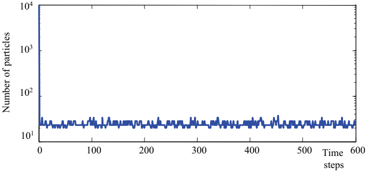
Experiment with a real robot



(a)



(b)



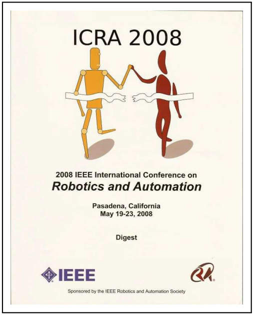
(c)

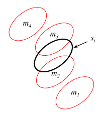
Experiment with a real robot

(optimal_sampling_localization.avi)

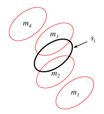
Papers

- ICRA 2008 (Pasadena, USA): *An Optimal Filtering Algorithm for Non-Parametric Observation Models in Robot Localization*.
- IJRR: Under review.





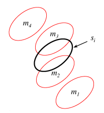
Outline of Part I



Introduction

A central component of Bayesian filtering is the sensor observation model:

$$\underbrace{p\left(x|z\right)}_{\text{Posterior}} \propto \underbrace{p\left(x\right)}_{\text{Prior}} \; \underbrace{p\left(z|x\right)}_{\substack{\text{Observation}\\ \text{likelihood}}}$$



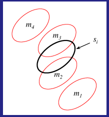
Introduction

A central component of Bayesian filtering is the sensor observation model:

$$\underbrace{p\left(x|z\right)}_{\text{Posterior}} \propto \underbrace{p\left(x\right)}_{\text{Prior}} \underbrace{p\left(z|x\right)}_{\substack{\text{Observation}\\ \text{likelihood}}}$$

For a laser range scanner, the observations at instant t are:

$$z_t = \left\{z_t^i\right\}_{i=1..L}$$



Introduction

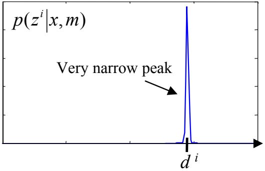
A central component of Bayesian filtering is the sensor observation model:

$$\underbrace{p\left(x|z\right)}_{\text{Posterior}} \propto \underbrace{p\left(x\right)}_{\text{Prior}} \; \underbrace{p\left(z|x\right)}_{\text{Observation likelihood}}$$

For a laser range scanner, the observations at instant t are:

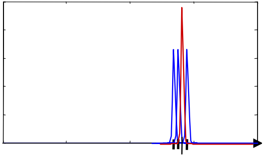
$$z_t = \left\{ z_t^i \right\}_{i=1..L}$$

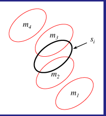
Given a pose x and a perfectly known map m:



Assuming cond. independence

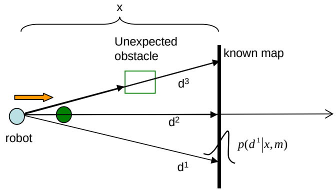
$$p\left(z \mid x,m\right) = \prod_i p\left(z^i \mid x,m\right)$$

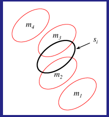




What does this *product* fusion imply for robot localization?

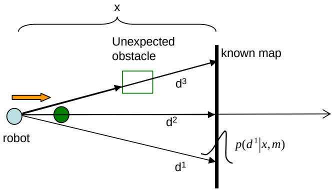
An example: a range scan with 3 measurements:



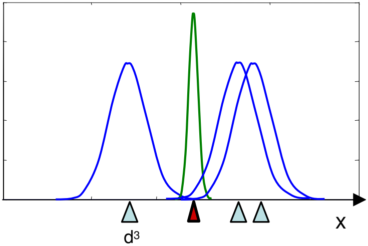


What does this *product* fusion imply for robot localization?

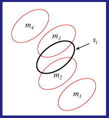
An example: a range scan with 3 measurements:



But in the presence of unexpected changes...

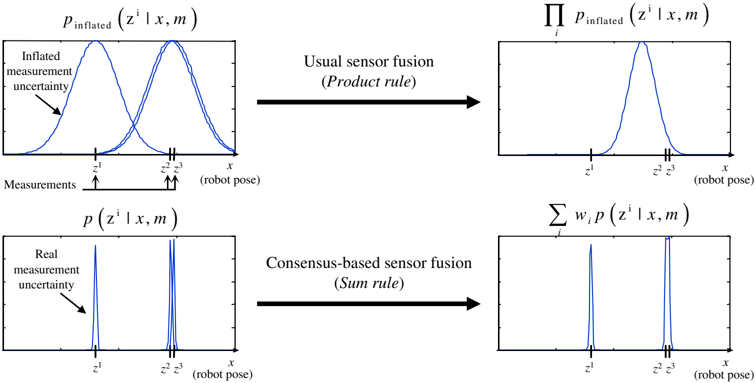


The peak likelihood is not compatible with any of the ranges!!



Our proposal: The *sum* fusion rule

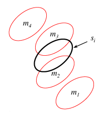
Our solution: to consider individual likelihood values as opinions about the final fused likelihood which is calculated by methods from Consensus Theory.



Our proposal: The Range Scan Likelihood Consensus (RSLC)

Our proposal: Linear Opinion Pool (LOP) of individual likelihoods:

$$p\left(z_t \middle| \mathbf{x}_t, m\right) \propto \sum_{i=1}^{L} \underbrace{p\left(z_t^i \middle| \mathbf{x}_t, m\right)}_{\text{Individual likelihood values}}$$



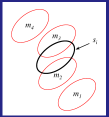
Our proposal: The Range Scan Likelihood Consensus (RSLC)

Our proposal: Linear Opinion Pool (LOP) of individual likelihoods:

$$p\left(z_t \middle| \mathbf{x}_t, m\right) \propto \sum_{i=1}^{L} \underbrace{p\left(z_t^i \middle| \mathbf{x}_t, m\right)}_{\substack{\text{Individual} \\ \text{likelihood values}}}$$

Each likelihood is computed from the "matching likelihood" for the M points in the map:

$$p\left(z_t^i \middle| \mathbf{x}_t, m\right) \propto \sum_{j=1}^{M} \mathrm{P}\left(c_{ij} \middle| \mathbf{x}_t, m, z_t^i\right)$$



Our proposal: The Range Scan Likelihood Consensus (RSLC)

Our proposal: Linear Opinion Pool (LOP) of individual likelihoods:

$$p\left(z_t \middle| \mathbf{x}_t, m\right) \propto \sum_{i=1}^{L} \underbrace{p\left(z_t^i \middle| \mathbf{x}_t, m\right)}_{\substack{\text{Individual} \\ \text{likelihood values}}}$$

Each likelihood is computed from the "matching likelihood" for the M points in the map:

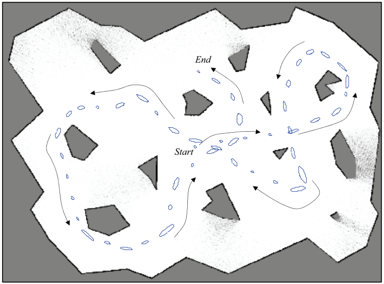
$$p\left(z_t^i \middle| \mathbf{x}_t, m\right) \propto \sum_{j=1}^{M} \mathrm{P}\left(c_{ij} \middle| \mathbf{x}_t, m, z_t^i\right)$$

Including the option of the point not corresponding with any map point ($\emptyset$):

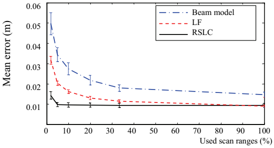
$$\mathrm{P}\left(c_{i\emptyset} \middle| \mathbf{x}_t, m, z_t^i\right) = 1 - \sum_{j=1}^{M} \mathrm{P}\left(c_{ij} \middle| \mathbf{x}_t, m, z_t^i\right)$$

Simulations: Static world

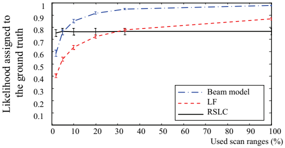
Simulations: Static world



(a)



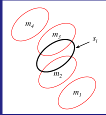
(b)



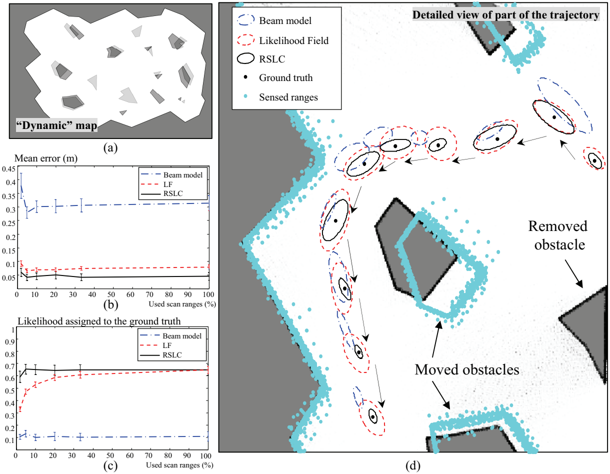
(c)

In the case of a perfectly known map, the RSLC behaves similarly to BM or LF.

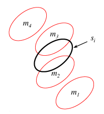
Simulations: Dynamic world



Simulations: Dynamic world

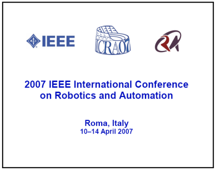


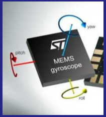
For a dynamic map, the RSLC reduces the error w.r.t. the BM and the LF.



Papers

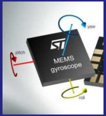
- ICRA 2007 (Rome, Italy): *A Consensus-based Approach for Estimating the Observation Likelihood of Accurate Range Sensors.*





Outline of Part I

1. Optimal particle filtering: applications to robot localization
2. A Consensus-based observation likelihood
3. Fusion of odometry and an IMU
   - Introduction
   - Filtering
   - Implementation
   - Results
   - Papers



Introduction

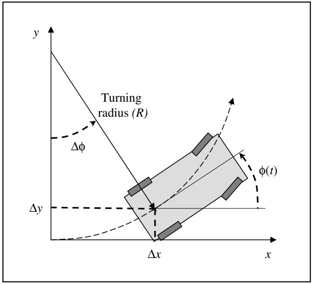
Another important part of the Bayesian filtering equation is the *transition model*:

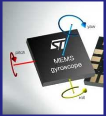
$$p(x_t|z_{1:t}, u_{1:t}, m) \quad \propto \quad p(z_t|x_t, m) \times \int \underbrace{p(x_t|x_{t-1}, u_t)}_{\text{Transition model}} p(x_{t-1}|z_{1:t-1}, u_{1:t-1}, m)dx_{t-1}$$

What is the new pose x_t *given the last one* x_{t-1} *and the action* u_t*?*

In mobile robotics this term is the probabilistic motion model: robot actions are not deterministic in the real world.

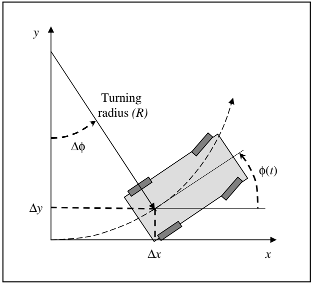
- The most usual approach: use the robot odometry as robot actions.

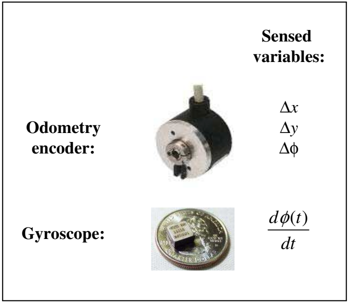


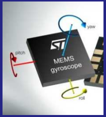


Introduction

- The most usual approach: use the robot odometry as robot actions.
- Our contribution here: probabilistic fusion of proprioceptive sensors.

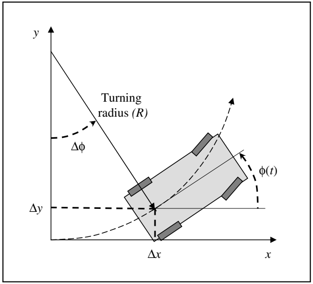


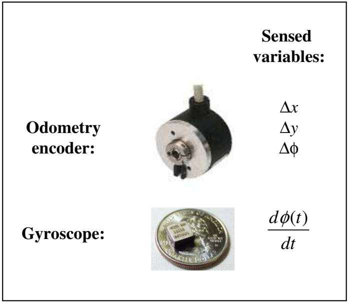


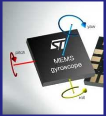


Introduction

- Each sensor contributes different information: they complement each other.





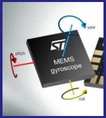


Filtering

Our solution: Apply an Extended Kalman Filter (EKF) to perform the fusion of odometry and a gyroscope.

- The state vector: the robot pose $(x_k \ y_k \ \phi_k)$ + the last orientation ϕ_{k-1}.

$$\mathbf{x}_k = \begin{pmatrix} x_k \\ y_k \\ \phi_k \\ \phi_{k-1} \end{pmatrix}$$



Filtering

1) Prediction:

- The mean: use the robot kinematic model.

$$\hat{\mathbf{x}}_k^- = f\left(\hat{\mathbf{x}}_{k-1}, u_k\right)$$

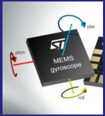
$$f\left(\hat{\mathbf{x}}_{k-1}, u_k\right) = \begin{pmatrix} x_k \\ y_k \\ \phi_k \\ \phi_{k-1} \end{pmatrix}$$

$$= \begin{pmatrix} x_{k-1} + \Delta x_k \cos\phi_{k-1} - \Delta y_k \sin\phi_{k-1} \\ y_{k-1} + \Delta x_k \sin\phi_{k-1} + \Delta y_k \cos\phi_{k-1} \\ \phi_{k-1} + \Delta\phi_k \\ \phi_{k-1} \end{pmatrix}$$

Filtering

- The covariance: independent contributions from x_{k-1} and u_k.

$$
\begin{aligned}
\mathbf{P}_k^- &= \begin{pmatrix} \nabla_{\mathbf{x}_k} f & \nabla_{u_k} f \end{pmatrix} \begin{pmatrix} \mathbf{P}_{k-1} & 0 \\ 0 & \mathbf{C}_{u_k} \end{pmatrix} \begin{pmatrix} \nabla_{\mathbf{x}_k} f^\top \\ \nabla_{u_k} f^\top \end{pmatrix} \\
&= \nabla_{\mathbf{x}_k} f \mathbf{P}_{k-1} \nabla_{\mathbf{x}_k} f^\top + \nabla_{u_k} f \mathbf{C}_{u_k} \nabla_{u_k} f^\top
\end{aligned}
$$



Filtering

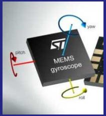
2) Update:

Kalman gain $\mathbf{K}_k$ computed from innovation $\mathbf{S}_k$:

$$\tilde{y}_k = z_k - h\left(\hat{\mathbf{x}}_k^-\right)$$

$$\mathbf{S}_k = \begin{pmatrix} \nabla_{\mathbf{x}_k} h & \nabla_{S_A} h & \nabla_n h \end{pmatrix} \begin{pmatrix} \mathbf{P}_k^- & 0 & 0 \\ 0 & \sigma_{S_A}^2 & 0 \\ 0 & 0 & \sigma_n^2 \end{pmatrix} \begin{pmatrix} \nabla_{\mathbf{x}_k} h^\top \\ \nabla_{S_A} h^\top \\ \nabla_n h^\top \end{pmatrix}$$

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^\top \mathbf{S}_k^{-1}$$



Filtering

2) Update:

Kalman gain $\mathbf{K}_k$ computed from innovation $\mathbf{S}_k$:

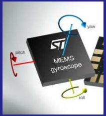
$$\tilde{y}_k = z_k - h\left(\hat{\mathbf{x}}_k^-\right)$$

$$\mathbf{S}_k = \begin{pmatrix} \nabla_{\mathbf{x}_k} h & \nabla_{S_A} h & \nabla_n h \end{pmatrix} \begin{pmatrix} \mathbf{P}_k^- & 0 & 0 \\ 0 & \sigma_{S_A}^2 & 0 \\ 0 & 0 & \sigma_n^2 \end{pmatrix} \begin{pmatrix} \nabla_{\mathbf{x}_k} h^\top \\ \nabla_{S_A} h^\top \\ \nabla_n h^\top \end{pmatrix}$$

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^\top \mathbf{S}_k^{-1}$$

Noise parameters:

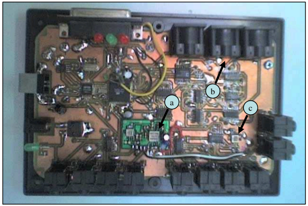
- $\sigma_{S_A}^2$: uncertainty about the actual sensor sensitivity.
- σ_n^2: gyroscope electrical noise (additive white Gaussian noise, AWGN).

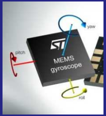


Implementation

Implementation on a real robot

- Hardware platform: Custom design based on a 8-bit AVR microcontroller.
- Sensor: Analog Devices gyroscope (ADXRS401).
- Real-time operation at 100Hz.



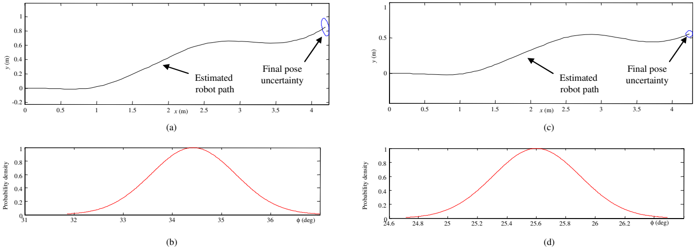


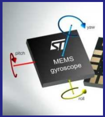
Results

Experiment I:

Smooth path of 4 meters.

Odometry only | Odometry + Gyroscope



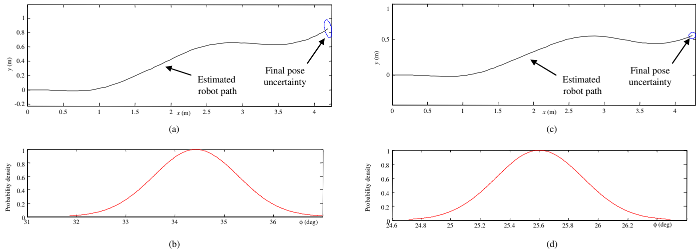


Results

Experiment I:

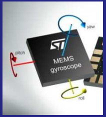
Smooth path of 4 meters.

Odometry only | Odometry + Gyroscope



Conclusion

Uncertainty in (x, y, ϕ) reduced by a factor of ~ 10.

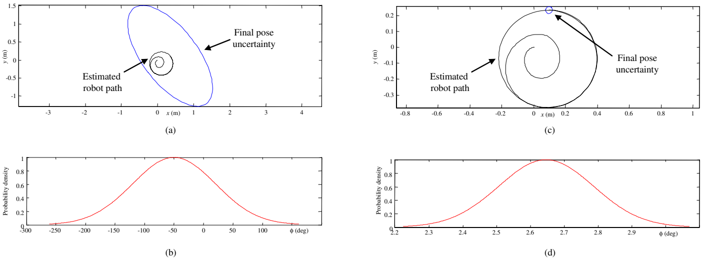


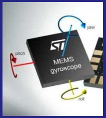
Results

Experiment II:

The robot turns around itself.

Odometry only | Odometry + Gyroscope





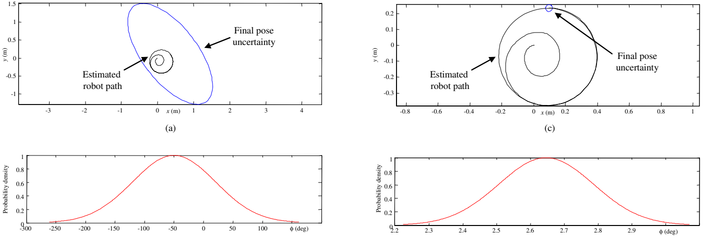
Results

Experiment II:

The robot turns around itself.

Odometry only

Odometry + Gyroscope

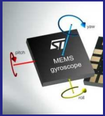


(a) (c)

(b) (d)

Conclusion

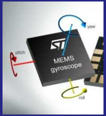
Uncertainty in (x, y, ϕ) reduced by a factor of $\sim 10^6$.



Results

Comparison to ground truth:

	Experiment I: Forward			Experiment II: Spinning		
	x	y	ϕ	x	y	ϕ
Odometry	4.194m	0.849m	34.42º	0.350m	0.114m	-49.55º
Sensor fusion	4.187m	0.934m	25.32º	0.096m	0.233m	2.65º
Ground truth	**4.169m**	**1.031m**	**25.80º**	**0.072m**	**0.282m**	**2.50º**



Papers

- ISSPA 2007 (Sharjah, United Arab Emirates): *Mobile robot ego-motion estimation by proprioceptive sensor fusion.*

Part II

SLAM

Outline of Part II

Differences between localization and SLAM

The optimal PF

- A generic optimal filtering algorithm has been introduced in this thesis.
- It has been applied to robot localization.

Would it be applicable to SLAM as well?

Differences between localization and SLAM

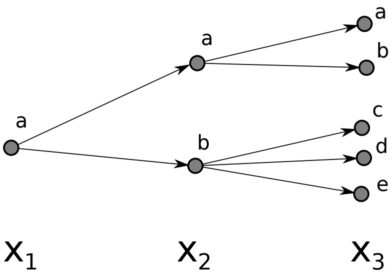
The optimal PF

- A generic optimal filtering algorithm has been introduced in this thesis.
- It has been applied to robot localization.

Would it be applicable to SLAM as well?

$\rightarrow$ Yes, with modifications.

RBPF-SLAM: estimating the robot *path*



Dimensionality $\propto$ time: exponential number of particles.

$$t = 1 \rightarrow \{x_1^a\}$$

$$t = 2 \rightarrow \{x_1^a, x_2^a\}, \{x_1^a, x_2^b\}$$

$$t = 3 \rightarrow \{x_1^a, x_2^a\ x_3^a\}, \{x_1^a, x_2^a, x_3^b\}, \{x_1^a, x_2^a, x_3^c\}, \ldots$$

The problems with RBPF-SLAM

Problems of our optimal filter with a RBPF

- It resamples at each step $\longrightarrow$ Quick loss of diversity.
- Dynamic number of samples $\longrightarrow$ Quick exponential growth.

The problems with RBPF-SLAM

Problems of our optimal filter with a RBPF

- It resamples at each step $\longrightarrow$ Quick loss of diversity.
- Dynamic number of samples $\longrightarrow$ Quick exponential growth.

The solutions

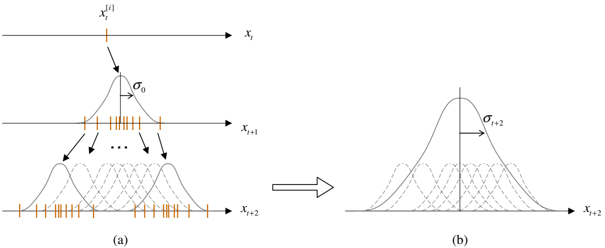
- Selective resampling.
- An approximation for KLD-sampling.

Selective resampling

Instead of selecting particles by their “survival probabilities”:

- Only run resampling if the ESS if below a threshold.
- Weights must be maintained now.

Approximation to KLD-sampling



Histogram over the whole path

$$M_t = O\left(k^{dt}\right)$$

Exponential

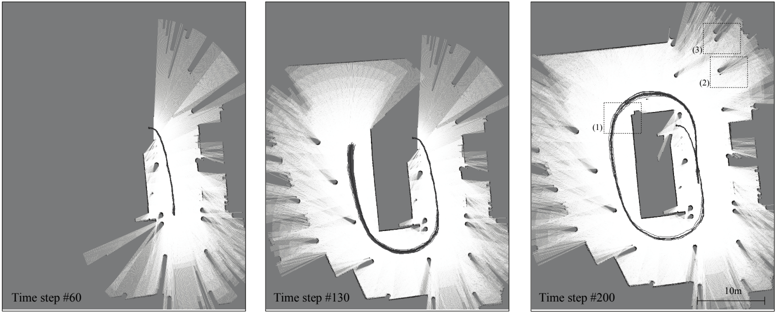
Histogram over the last pose

$$M_t = O\left(t^{\frac{d}{2}}\right)$$

Polynomial

Results

Results



PhD dissertation

Jose Luis Blanco Claraco

Optimal filtering in RBPF-SLAM

Introduction

The proposed changes

Experiments

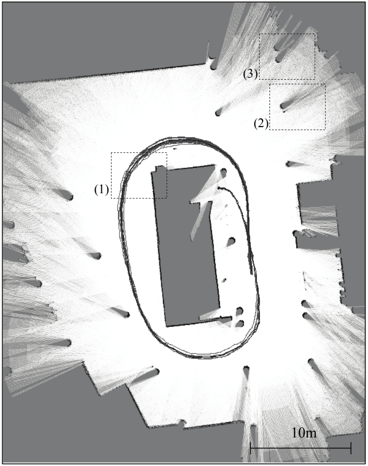
Papers

Range-Only SLAM

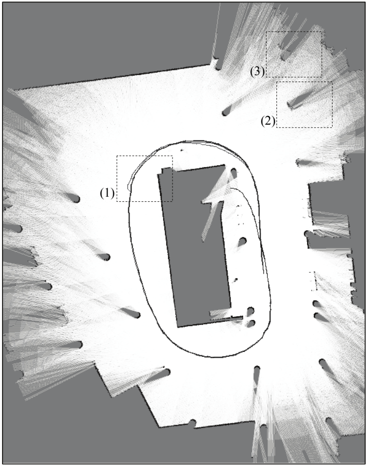
Uncertainty measures for SLAM and exploration

Results

Optimal Sampling

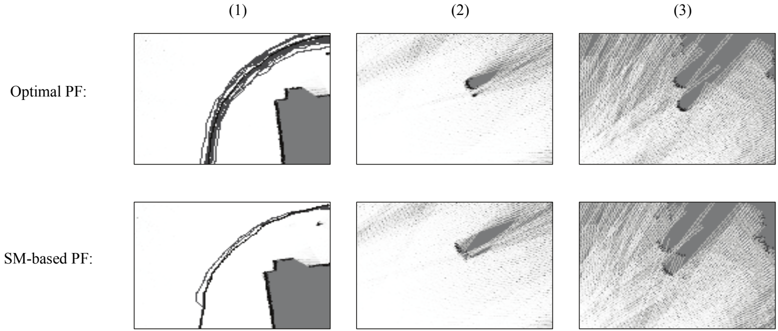


SM-based method

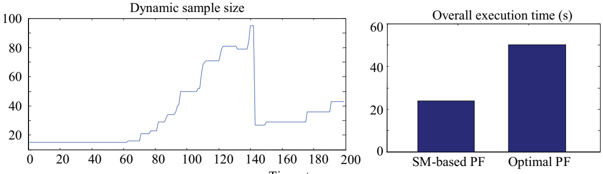


Time step #200

Results



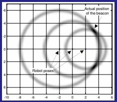
Results


Dynamic sample size
100
80
60
40
20
0
20
40
60
80
100
120
140
160
180
200
Time steps
Overall execution time (s)
60
40
20
0
SM-based PF
Optimal PF

(a) (b)

Papers

- IJRR: "Optimal Filtering for Non-Parametric Observation Models: Applications to Localization and SLAM" (Under review).

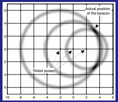


Outline of Part II

4. Optimal filtering in RBPF-SLAM

5. Range-Only SLAM
   - Introduction
   - Our Sum of Gaussians (SOG) approach
   - Results
   - Papers

6. Uncertainty measures for SLAM and exploration



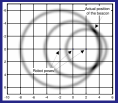
Range-only SLAM

Advantages of RO-SLAM

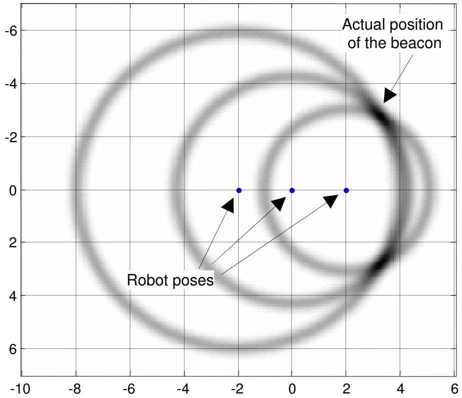
- Data association is usually trivial.
- Some technologies work across walls, etc.

One big drawback

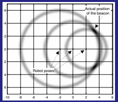
- Ambiguity...



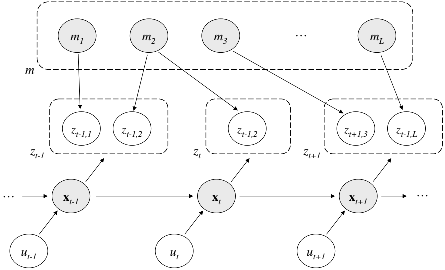
Range-only SLAM: Ambiguity



This complicates a standard EKF-SLAM approach.

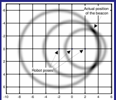


RBPF

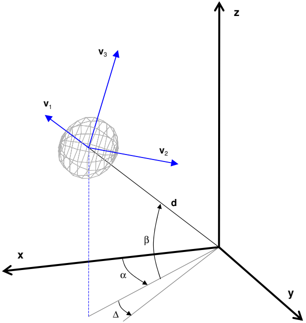


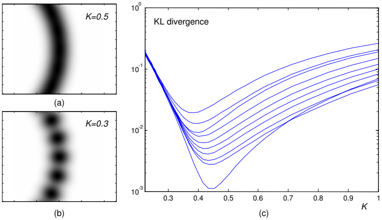
The proposed solution

- To employ a RBPF and keep independent distributions for each beacon.
- Use a Sum of Gaussians (SOG) for each beacon.



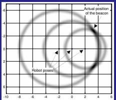
Inverse sensor model





Inverse sensor model implementation

- The first time a beacon is observed: create the SOG.
- Subsequent observations: update both the SOG parameters and its weight (each Gaussian mode is an EKF).

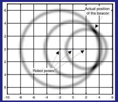


The symmetry in 3D RO-SLAM

For robot moving in planar scenarios, there is an ambiguity that will ever remain...

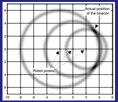
The symmetry in 3D RO-SLAM

(ROSLAM_symmetry1.avi)



The symmetry in 3D RO-SLAM

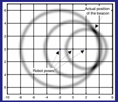
...if there is a priori knowledge → populate only one half of the 3D space.



Simulations

The method has been validated with several simulations:

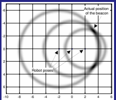
(roslam_simul.avi)



UWB real sensors

Experiments with real UWB sensors have been performed as well.

(roslam_real.avi)



Papers

The presented method:

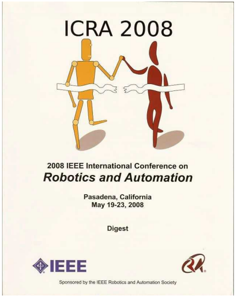
- IROS 2008 (Nice, France) *Efficient Probabilistic Range-Only SLAM*.

Very related to this method:

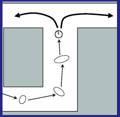
- ICRA 2008 (Pasadena, USA) *A Pure Probabilistic Approach to Range-Only SLAM*.

Parallel works (with other authors: A. Ortiz-de-Galisteo, F.A. Moreno, J. Martínez):

- RAS *Mobile Robot Localization based on Ultra-Wide-Band Ranging: A Particle Filter Approach*.
- ISSPA 2007 (Sharjah, United Arab Emirates) *Combination of UWB and GPS for indoor-outdoor vehicle localization*.



Outline of Part II



Information in SLAM

The problem:

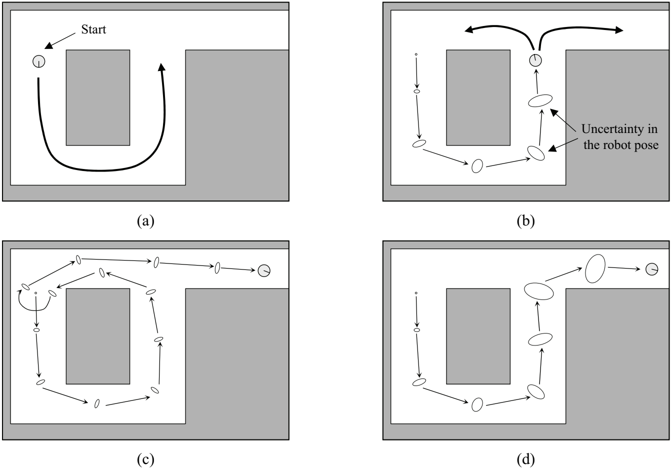
Measuring the information or *uncertainty* in a RBPF-based implementation of SLAM at a given instant of time.

Why measuring the information?

- In active exploration, to decide the next robot movement.
- In active loop-closing behavior: to detect the end of a loop closure.

Information-guided exploration

Why is it an advantage to be prone to loop closures?



(a) (b) (c) (d)

Information-guided exploration

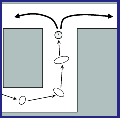
Previously employed uncertainty measures:

- Effective Sample Size (ESS).
- Area covered by particles.
- Entropy of the robot path.
- Joint entropy of the path + the map (the most grounded method) [Stachniss et al. 2005].

Instead we propose:

- To firstly build an auxiliary map, the expected map (EM).
- Then, measure its Information (EMI) or Mean Information (EMMI).

It is shown why our alternative behaves better that the others.



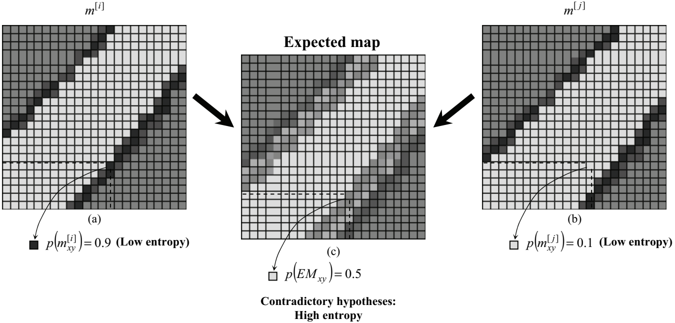
The expected map (EM)

The EM is defined as the average of the map m over all possible paths x^t. In particular, for RBPF-SLAM:

$$
\begin{aligned}
p(EM|z^t, u^t) &\doteq \mathrm{E}_{x^t}\left[p(m|x^t, z^t, u^t)\right] \\
&= \int \cdots \int_{-\infty}^{\infty} p(m|x^t, z^t, u^t) p(x^t|z^t, u^t) dx^t \\
&\approx \sum_{i=1}^{M} \omega_k^{[i]} p(m_{xy}|x^{[i],t}, z^t, u^t)
\end{aligned}
$$

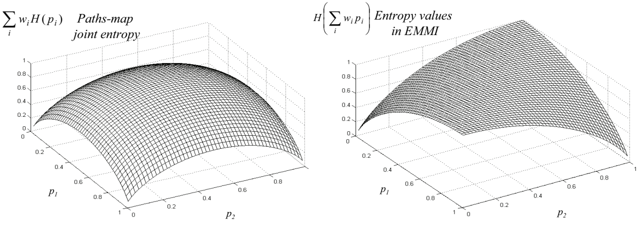
The expected map (EM)

For grid maps, the EM has the property of revealing discrepancies between individual maps:



The expected map (EM)

It is important to remark it one more time: the way the EM detects inconsistencies is really new in comparison to the joint entropy.

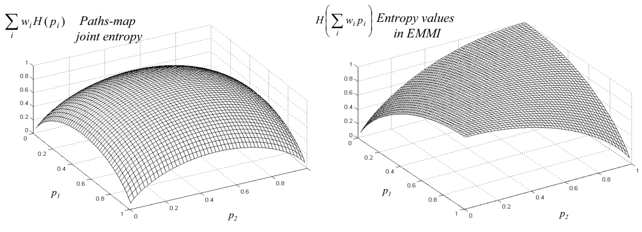


(a) (b)

Example with two particles, each holding one grid map cell: p_1 and p_2.

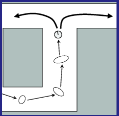
The expected map (EM)

It is important to remark it one more time: the way the EM detects inconsistencies is really new in comparison to the joint entropy.



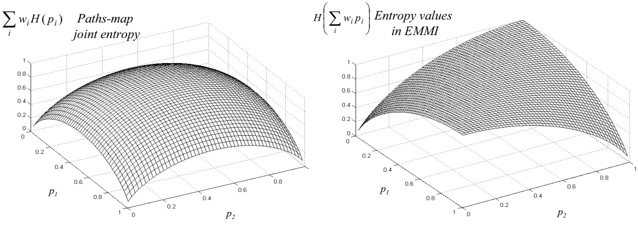
(a) (b)

If both cells are unobserved, $p_1 = p_2 = 0.5$, our method and the joint entropy agree.



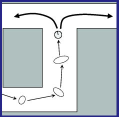
The expected map (EM)

It is important to remark it one more time: the way the EM detects inconsistencies is really new in comparison to the joint entropy.



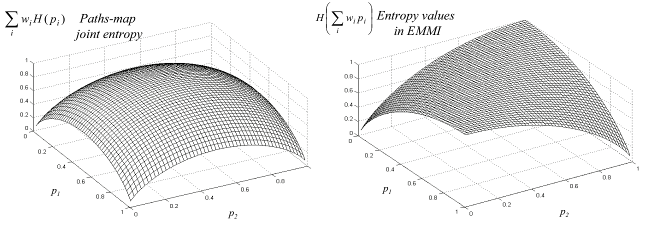
(a) (b)

If both cells are observed and equal, $p_1 = p_2 = 1$ or $p_1 = p_2 = 0$, our method and the joint entropy agree (low entropy $\rightarrow$ desirable).



The expected map (EM)

It is important to remark it one more time: the way the EM detects inconsistencies is really new in comparison to the joint entropy.



(a) (b)

If both cells are observed and inconsistent, e.g. $p_1 = 0$, $p_2 = 1$, our method gives a high entropy, while the joint entropy is low.

Information vs. Entropy

Apart from defining the EM, it is also proposed to replace "entropy" by "information":

Entropy:	$H(m_{xy}) = -p(m_{xy}) \log p(m_{xy}) - \bar{p}(m_{xy}) \log \bar{p}(m_{xy})$
Map information:	$I(m_{xy}) = 1 - H(m_{xy})$
Cell information:	$I(m) = \sum_{\forall x,y} I(m_{xy}) \qquad \text{(bits)}$

Information and entropy are almost synonymous, but $I(\cdot)$ is defined for convenience:

- Non-observed cells ($p = 0.5$) do not count.
- Independence of the size and resolution of grid maps.

The EMI and EMMI

Expected Map Information (EMI)

The overall information of the entire EM of a RBPF.

Useful for: Exploration.

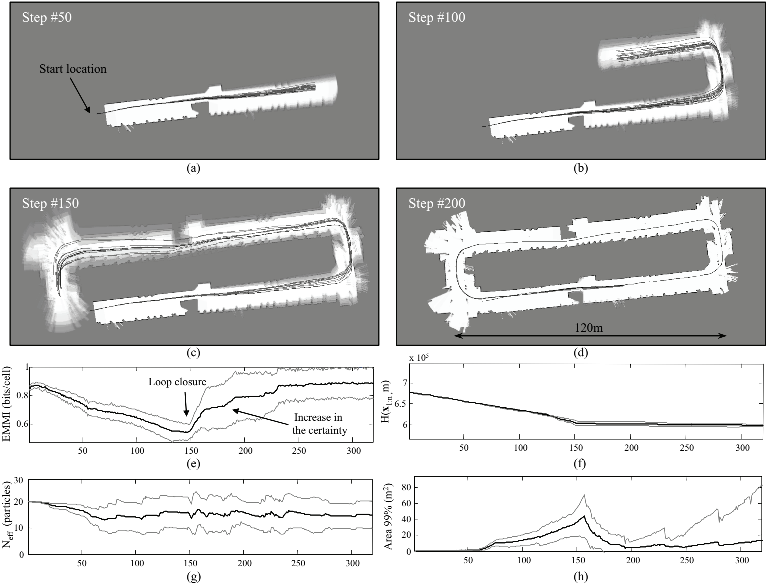
Expected Map Mean Information (EMMI)

The mean information per grid cell of the entire EM of a RBPF.

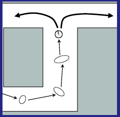
Useful for: Detection of loop closures.

EMMI for loop closure detection

EMMI for loop closure detection

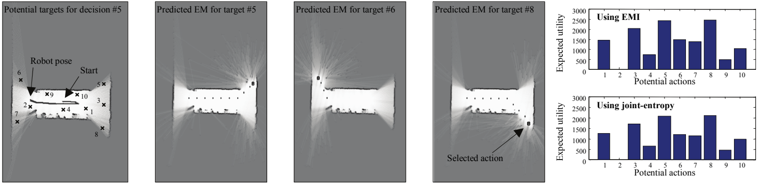


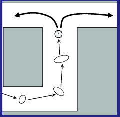
EMI for exploration



EMI for exploration

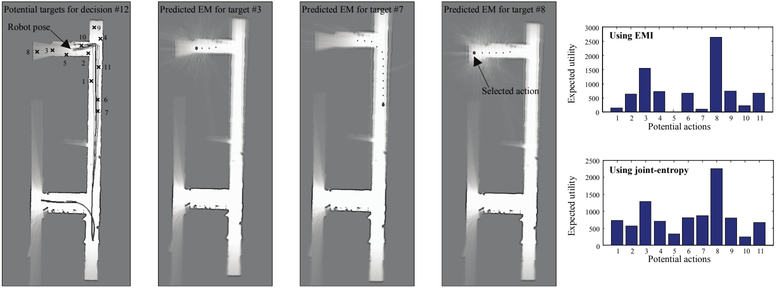
Decision #5:





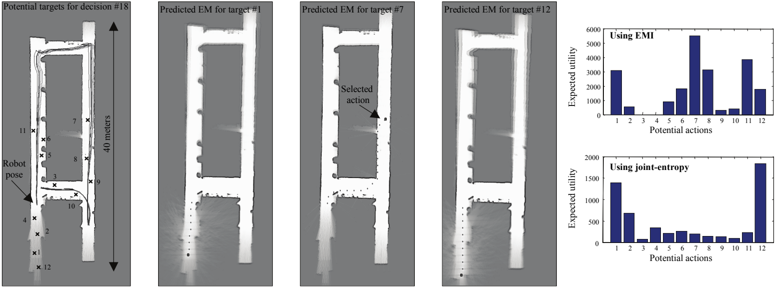
EMI for exploration

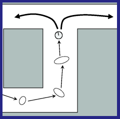
Decision #12:



EMI for exploration

Decision #18:





Papers

PhD dissertation

Jose Luis Blanco Claraco

- IJRR: *A Novel Measure of Uncertainty for Mobile Robot SLAM with Rao-Blackwellized Particle Filters*.

J. L. Blanco
J. A. Fernández-Madrigal
J. Gonzalez
Department of System Engineering and Automation,
University of Málaga, 29071 Málaga, Spain
{jlblanco,jafma,jgonzalez}@ctima.uma.es

A Novel Measure of Uncertainty for Mobile Robot SLAM with Rao–Blackwellized Particle Filters

Abstract

Rao–Blackwellized particle filters (RBPFs) are an implementation of sequential Bayesian filtering that has been successfully applied to mobile robot simultaneous localization and mapping (SLAM) and exploration. Measuring the uncertainty of the distribution estimated by a RBPF is required for tasks such as information gain-guided exploration or detecting loop closures in nested loop environments. In this paper we propose a new measure that takes the uncertainty in both the robot path and the map into account. Our approach relies on the entropy of the expected map (EM) of the RBPF, a new variable built by integrating the map hypotheses from all of the particles. Unlike previous works that use the joint entropy of the RBPF for active exploration, our proposal is better suited to detect opportunities to close loops, a key aspect to reduce the robot path uncertainty and consequently to improve the quality of the maps being built. We provide a theoretical discussion and experimental results with real data that support our claims.

KEY WORDS—localization, mapping

1. Introduction

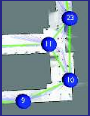
The automated mapping of unknown environments is one of the fundamental problems that need to be solved to achieve truly autonomous mobile robots. The difficulty of this task follows from the fact that a precise map can only be obtained from a well-localized robot, but in turn the quality of the robot pose estimation depends on the map accuracy: this is the simultaneous localization and mapping (SLAM) problem. In recent years, methods based on estimation theory have dominated the research in this field. In these approaches the robot pose and the map are represented in terms of probability densities which are tracked over time. Tracking is usually performed through a Bayesian filtering of either Gaussian (Dissanayake et al. 2001) or Gaussian mixture (Porta and Kröse 2006) distributions with Kalman filters, or sequential Monte Carlo (SMC) sampling schemes with particle filters (Doucet et al. 2000b). More concretely, the ability of SMC methods to solve the problems of global localization (Dellaert et al. 1999) and SLAM (Doucet et al. 2000a) efficiently has been demonstrated, because they can deal with non-linear models and any shape, multi-modal distributions. From this family of methods, Rao–Blackwellized particle filters (RBPFs) are widely employed to estimate both the robot path and the map simultaneously (Grisetti et al. 2007b; Montemerlo et al. 2002). In this scheme, probability densities are maintained by a set of weighted particles, which are hypotheses for the robot path. The Rao–Blackwellization consists of deriving maps analytically from these paths, which reduces the dimensional complexity of the SLAM problem (Doucet et al. 2000a). The most likely path (and, therefore, map) is usually considered to be that associated with the particle with the highest weight, as the example in Figure 1(a) shows.

In general, SLAM methods passively process incoming sensor data and update the map and path estimates iteratively. However, the advantages of allowing the robot to actively control its movements while building a map, that is, active exploration, are well known and have been reported in the literature (Stachniss et al. 2004; Sim and Roy 2005). Exploration methods aim at controlling a robot through an unknown scenario in such a way that the whole environment is mapped while minimizing some cost function, such as the total distance traveled (Yamauchi 1998; Burgard et al. 2000) or the uncertainty in localization (Stachniss et al. 2005b). To illustrate the potential impact of movement selection in the accuracy of the resulting map, consider the example in Figure 2(a) where a robot ex-

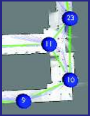
The International Journal of Robotics Research
Vol. 27, No. 1, January 2008, pp. 73–89
DOI: 10.1177/0278364907082610
©SAGE Publications 2008 Los Angeles, London, New Delhi and Singapore
Figures 3, 5, 8–10 appear in color online: http://ijr.sagepub.com

Part III

Large-scale SLAM



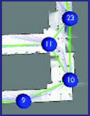
Outline of Part III



Overview

A fundamental choice in SLAM is the kind of **map representation**:

- Metric maps: Detailed, quantitative information (landmarks, occupancy grids).
- Topological maps: Graphs. Sparse information.
- Hybrid representations: Graphs where nodes are local metric maps.



Overview

A fundamental choice in SLAM is the kind of **map representation**:

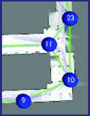
- Metric maps: Detailed, quantitative information (landmarks, occupancy grids).
- Topological maps: Graphs. Sparse information.
- Hybrid representations: Graphs where nodes are local metric maps.

Hybrid maps

Promising solution, adopted in many previous works.

- Kuipers, Byun, "A Robot Exploration and Mapping Strategy...", 2001
- Estrada, Neira, Tardós, "Hierarchical SLAM:...", 2005.

Typically, the problem is addressed from the point of view of a hierarchical **metric** map.



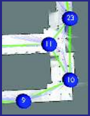
Overview

We introduce a new approach to SLAM, ideally suitable for large-scale environments and long-term, robust operation.

Contributions:

- Introduction of the concept of **hybrid metric-topological** (HMT) path.
- Consistent formulation of HMT-SLAM as a **unified** Bayesian estimation problem.

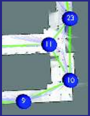
The estimation of the robot HMT path is an advance comparable to RBPF or FastSLAM in metric SLAM.



Overview

Some relevant advantages of HMT-SLAM

- Better **accuracy** in the estimation of loop closures.
- Efficient representation of multiple hypotheses of the **topological structure** of the environment.
- A potential application not addressed by previous works: **global localization** within a partially mapped environment.



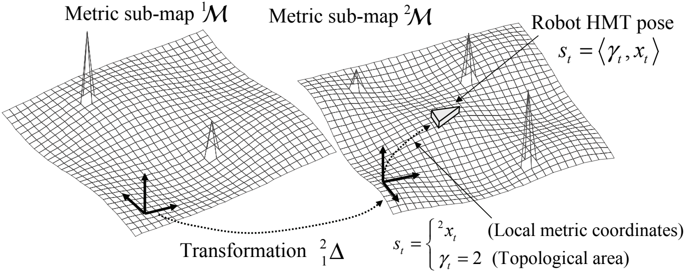
Basics of HMT-SLAM

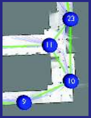
HMT path

We introduce the concept of **hybrid path** of a robot.

Each pose is composed of a metric x_t and a topological γ_t component:

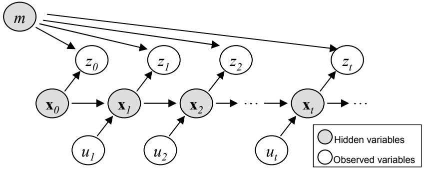
$$s_t = \{x_t, \gamma_t\}$$



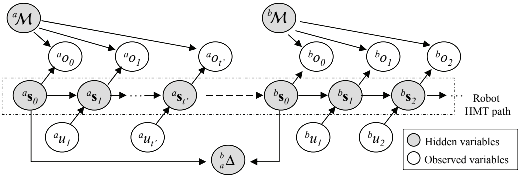


Probabilistic model

Graphical model of standard SLAM



Graphical model of HMT-SLAM



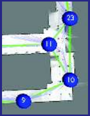
Derivation

$p(s^t, m | u^t, o^t)$

Derivation

$$p(s^t, m | u^t, o^t) = p(s^t | u^t, o^t) p(m | s^t, u^t, o^t)$$

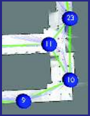
Definition of conditional probability:
one part for the HMT path, one for the map.



Derivation

$$p(s^t, m|u^t, o^t) = p(s^t|u^t, o^t)p(m|s^t, u^t, o^t)$$

$$p(m|s^t, u^t, o^t) = p\left(\left\{{}^kM\right\}, \left\{{}^{ab}\Delta\right\} \middle| s^t, u^t, o^t\right)$$

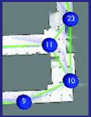


Derivation

$$p(s^t, m|u^t, o^t) = p(s^t|u^t, o^t)\color{red}{p(m|s^t, u^t, o^t)}$$

$$p(m|s^t, u^t, o^t) = p\left(\left\{{}^kM\right\}, \left\{{}^{ab}\Delta\right\}\middle| s^t, u^t, o^t\right)$$

$$= \underbrace{p\left(\left\{{}^kM\right\}\middle| s^t, u^t, o^t\right)}_{\text{Content of the metric sub - maps}} \underbrace{p\left(\left\{{}^{ab}\Delta\right\}\middle| \left\{{}^kM\right\}, s^t, u^t, o^t\right)}_{\text{Arcs in the topological part of HMT - maps}}$$



Derivation

$$p(s^t, m | u^t, o^t) = p(s^t | u^t, o^t) p(m | s^t, u^t, o^t)$$

$$p(m | s^t, u^t, o^t) = p\left(\left\{ {}^k M \right\}, \left\{ {}^{ab}\Delta \right\} \middle| s^t, u^t, o^t \right)$$

$$= p\left(\left\{ {}^k M \right\} \middle| s^t, u^t, o^t \right) p\left(\left\{ {}^{ab}\Delta \right\} \middle| \left\{ {}^k M \right\}, s^t, u^t, o^t \right)$$

$$p\left(\left\{ {}^k M \right\} \middle| s^t, u^t, o^t \right) = \prod_k p\left({}^k M \middle| {}^k s^t, u^t, o^t \right)$$

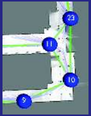
Conditional independence of sub-maps

Derivation

$$p(s^t, m|u^t, o^t) = p(s^t|u^t, o^t)p(m|s^t, u^t, o^t)$$

$$p(m|s^t, u^t, o^t) = p\left(\left\{{}^kM\right\}, \left\{{}^{ab}\Delta\right\}\middle| s^t, u^t, o^t\right)$$
$$= p\left(\left\{{}^kM\right\}\middle| s^t, u^t, o^t\right) p\left(\left\{{}^{ab}\Delta\right\}\middle| \left\{{}^kM\right\}, s^t, u^t, o^t\right)$$

$$p\left(\left\{{}^kM\right\}\middle| s^t, u^t, o^t\right) = \prod_k p\left({}^kM\middle| {}^ks^t, u^t, o^t\right)$$
$$= \prod_k p\left({}^kM\middle| {}^kx^t, {}^k\gamma^t, u^t, z^t, \psi^t\right)$$



Derivation

$$p(s^t, m | u^t, o^t) = p(s^t | u^t, o^t) p(m | s^t, u^t, o^t)$$

$$p(m|s^t,u^t,o^t) = p\left(\left\{{}^kM\right\}, \left\{{}^{ab}\Delta\right\} \middle| s^t, u^t, o^t\right)$$

$$= p\left(\left\{{}^kM\right\} \middle| s^t, u^t, o^t\right) p\left(\left\{{}^{ab}\Delta\right\} \middle| \left\{{}^kM\right\}, s^t, u^t, o^t\right)$$

$$\begin{aligned} p\left(\left\{{}^kM\right\} \middle| s^t, u^t, o^t\right) &= \prod_k p\left({}^kM \middle| {}^ks^t, u^t, o^t\right) \\ &= \prod_k p\left({}^kM \middle| {}^kx^t, {}^k\gamma^t, u^t, z^t, \psi^t\right) \\ &= \prod_k p\left({}^kM \middle| {}^kx^t, z^t\right) \end{aligned}$$

These have **closed-form** solutions for grid maps and landmark maps.

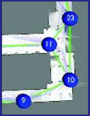
Derivation

$$p(s^t, m|u^t, o^t) = p(s^t|u^t, o^t) p(m|s^t, u^t, o^t)$$

$$p(m|s^t, u^t, o^t) = p\left(\left\{{}^kM\right\}, \left\{{}^{ab}\Delta\right\}\middle| s^t, u^t, o^t\right)$$

$$= p\left(\left\{{}^kM\right\}\middle| s^t, u^t, o^t\right) p\left(\left\{{}^{ab}\Delta\right\}\middle| \left\{{}^kM\right\}, s^t, u^t, o^t\right)$$

$$p\left(\left\{{}^{ab}\Delta\right\}\middle| \left\{{}^kM\right\}, s^t, u^t, o^t\right)$$



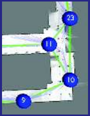
Derivation

$$p(s^t, m|u^t, o^t) = p(s^t|u^t, o^t)p(m|s^t, u^t, o^t)$$

$$p(m|s^t, u^t, o^t) = p\left(\left\{{}^kM\right\}, \left\{{}^{ab}\Delta\right\}\middle| s^t, u^t, o^t\right)$$

$$= p\left(\left\{{}^kM\right\}\middle| s^t, u^t, o^t\right) p\left(\left\{{}^{ab}\Delta\right\}\middle| \left\{{}^kM\right\}, s^t, u^t, o^t\right)$$

$$p\left(\left\{{}^{ab}\Delta\right\}\middle| \left\{{}^kM\right\}, s^t, u^t, o^t\right) = \prod_{(a,b)} p\left({}^{ab}\Delta\middle| \left\{{}^kM\right\}, s^t, u^t, o^t\right)$$



Derivation

$$p(s^t, m|u^t, o^t) = p(s^t|u^t, o^t)p(m|s^t, u^t, o^t)$$

$$p(m|s^t, u^t, o^t) = p\left(\left\{{}^kM\right\}, \left\{{}^{ab}\Delta\right\}\middle| s^t, u^t, o^t\right)$$

$$= p\left(\left\{{}^kM\right\}\middle| s^t, u^t, o^t\right) p\left(\left\{{}^{ab}\Delta\right\}\middle| \left\{{}^kM\right\}, s^t, u^t, o^t\right)$$

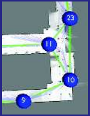
$$p\left(\left\{{}^{ab}\Delta\right\}\middle| \left\{{}^kM\right\}, s^t, u^t, o^t\right) = \prod_{(a,b)} p\left({}^{ab}\Delta\middle| \left\{{}^kM\right\}, s^t, u^t, o^t\right)$$

$$= \prod_{(a,b)} \prod_i N\left({}^{ab}\bar{\Delta}^i, {}^{ab}\Sigma^i\right)$$

Individual estimations obtained each time a topological loop closure occurs.

Derivation

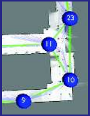
$$p(s^t, m|u^t, o^t) = p(s^t|u^t, o^t)p(m|s^t, u^t, o^t)$$



Derivation

$$p(s^t, m | u^t, o^t) = p(s^t | u^t, o^t) p(m | s^t, u^t, o^t)$$

$$p(s^t | u^t, o^t) \propto p(o_t | s^t, u^t, o^{t-1}) p(s^t | u^t, o^{t-1})$$



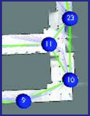
Derivation

$$p(s^t, m|u^t, o^t) = p(s^t|u^t, o^t)p(m|s^t, u^t, o^t)$$

$$p(s^t|u^t, o^t) \propto p(o_t|s^t, u^t, o^{t-1})p(s^t|u^t, o^{t-1})$$

This part can be estimated using a particle filter:

$$\left\{s^{t,[i]}\right\} \sim p(s^t|s^{t-1,[i]}, u^{t-1}, o^{t-1})$$



Derivation

$$p(s^t, m|u^t, o^t) = \color{red}{p(s^t|u^t, o^t)}\color{black}{p(m|s^t, u^t, o^t)}$$

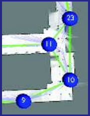
$$p(s^t|u^t, o^t) \propto p(o_t|s^t, u^t, o^{t-1})p(s^t|u^t, o^{t-1})$$

This part can be estimated using a particle filter:

$$\left\{s^{t,[i]}\right\} \sim p(s^t|s^{t-1,[i]}, u^{t-1}, o^{t-1})$$

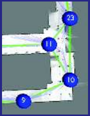
This process includes:

- Detecting when the robot enters into new areas.
- Generating topological loop closure hypotheses.



Advantages of HMT-SLAM

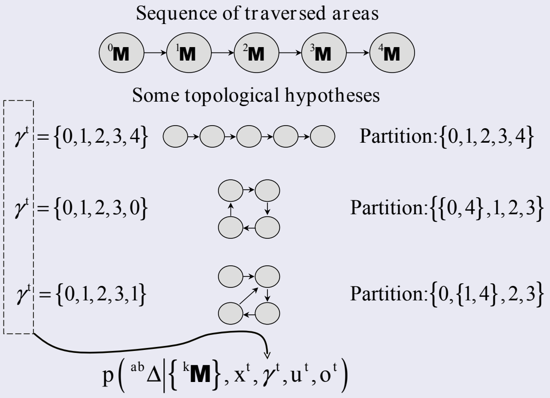
- More accurate loop closures.
- Capable of handling localization in partially known environments.

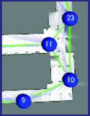


Advantages of HMT-SLAM

More accurate loop closures.

In HMT-SLAM, loop closure becomes a partition in the robot topological path:

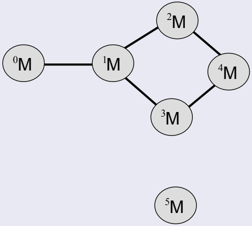


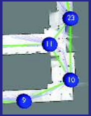


Advantages of HMT-SLAM

Localization with partially unknown maps

An example:

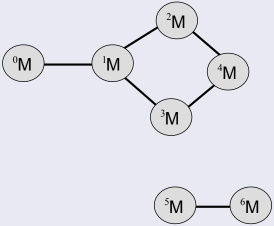




Advantages of HMT-SLAM

Localization with partially unknown maps

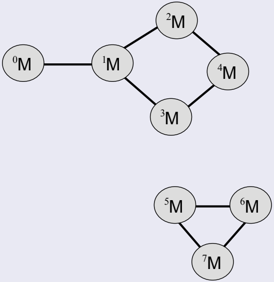
An example:



Advantages of HMT-SLAM

Localization with partially unknown maps

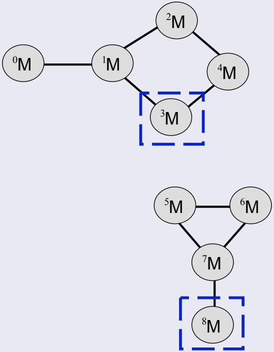
An example:

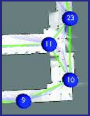


Advantages of HMT-SLAM

Localization with partially unknown maps

An example:

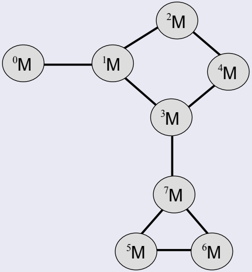




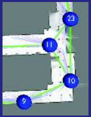
Advantages of HMT-SLAM

Localization with partially unknown maps

An example:



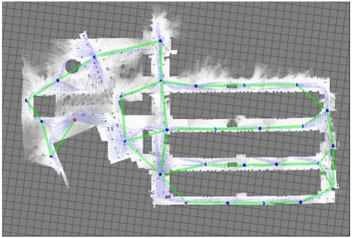
Experiments

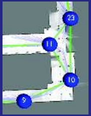


Experiments

Dataset: Málaga

- 2Km robot path. About 4700 laser scans.
- Covered area: 30 000 m2
- Several loop closures.
- Acquisition time: 21min

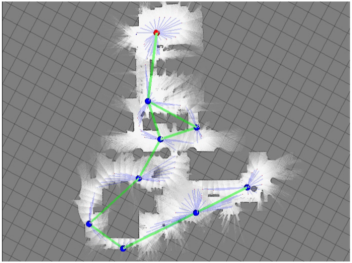




Experiments

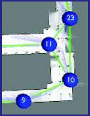
Dataset: Edmonton

- 1Km robot path. About 3000 laser scans.
- Covered area: 7 000 m2
- One loop closure. Acquisition time: 12min
- Gathered at the “Edmonton Convention Centre” by Nick Roy.



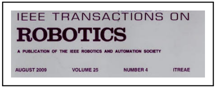
Experiments

(HMT-SLAM_malaga.avi)

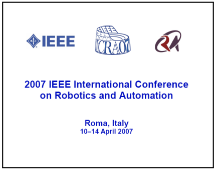


Papers

- TRO: *Towards a Unified Bayesian Approach to Hybrid Metric-Topological SLAM.*



- ICRA 2007 (Rome, Italy): *A New Approach for Large-Scale Localization and Mapping: Hybrid Metric-Topological SLAM.*



Outline of Part III

Introduction

Sub maps

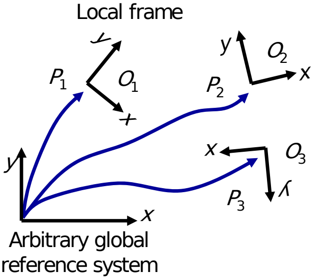
To build hybrid maps, we need to **cluster** areas of the space into discrete “entities” $\longrightarrow$ nodes in the HMT maps.
In the literature the common approach is to group landmarks into metric sub-maps.

Our proposal

Instead of clustering the map elements, clustering the robot **observations**.
This has advantages since segments of the robot path become **conditional independent** while submaps still **overlap**.

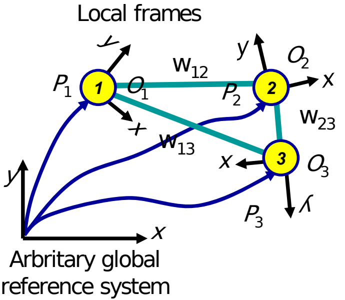
Graphs of observations

We propose to build an auxiliary graph of observations:



Graphs of observations

We propose to build an auxiliary graph of observations:



$$\mathbf{W} = \begin{pmatrix} w_{11} \; \ldots \; w_{1N} \\ \ldots \\ w_{M1} \; \ldots \; w_{MN} \end{pmatrix}$$

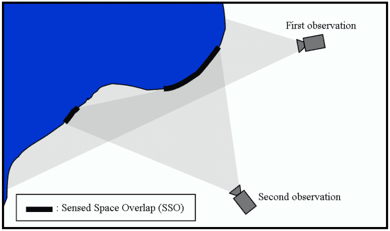
Nodes $\longrightarrow$ observations

Edges $\longrightarrow$ similarity or overlap between observations

Graphs of observations

The Sensed Space Overlap (SSO) is a function that assigns a similarity measurement to any pair of observations:

$$SSO : V \times V \rightarrow [0, 1]$$

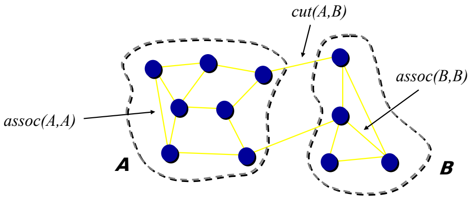


Partitioning graphs

Normalized cut (Ncut)

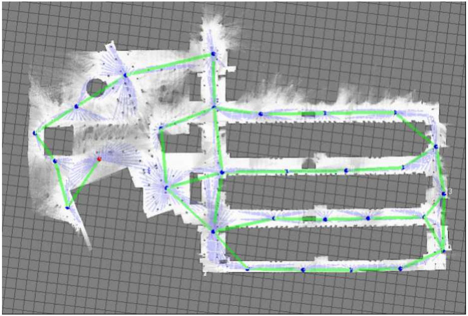
- The N-cut was first introduced by Shi & Malik in the context of image segmentation.
- This partitioning method finds bisections of a graph being a trade-off between their sizes and their mutual connections.

$$Ncut(A,B) = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)}$$



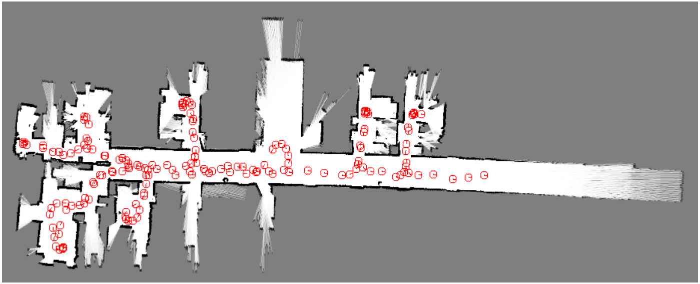
Experiments

This technique has been already shown in action within HMT-SLAM.

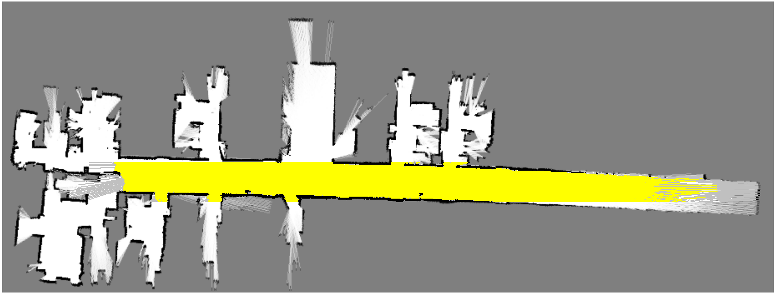


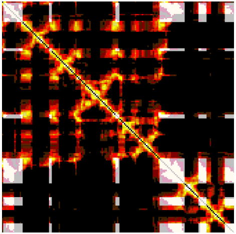
Experiments

Another experiment: recursive bisection of a global indoor map:



Experiments

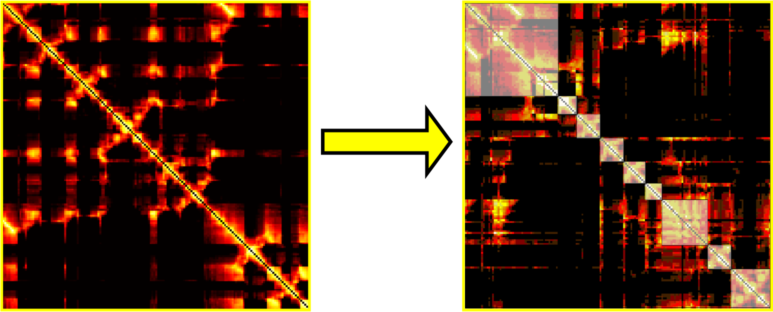




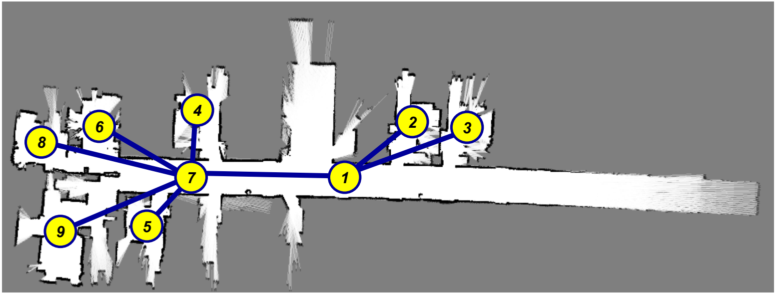
Cluster 1 out of 9

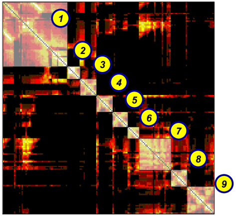
Experiments

Finally, if we rearrange the weight matrix according to the clustering, it is closer to being block diagonal:

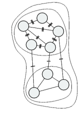


Experiments



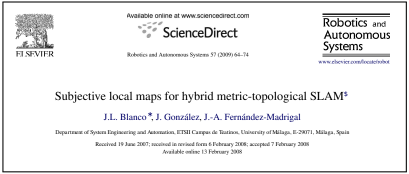


All the sub-maps



Papers

- RAS: *Subjective Local Maps for Hybrid Metric-Topological SLAM.*



Available online at www.sciencedirect.com

ScienceDirect

Robotics and Autonomous Systems 57 (2009) 64–74

ELSEVIER

Robotics and Autonomous Systems

www.elsevier.com/locate/robot

Subjective local maps for hybrid metric-topological SLAM☆

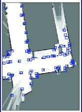
J.L. Blanco*, J. González, J.-A. Fernández-Madrigal

Department of System Engineering and Automation, ETSII Campus de Teatinos, University of Málaga, E-29071, Málaga, Spain

Received 19 June 2007; received in revised form 6 February 2008; accepted 7 February 2008
Available online 13 February 2008

- ICRA 2006 (Orlando, USA): *Consistent Observation Grouping for Generating Metric-Topological Maps that Improves Robot Localization.*

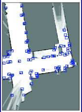
Outline of Part III



Introduction

Why to match grid maps?

- In "standard" global mapping $\rightarrow$ metric loop closure.
- In multi-robot mapping $\rightarrow$ to join maps.
- In HMT-SLAM $\rightarrow$ to detect topological loop closures.



Introduction

Pros

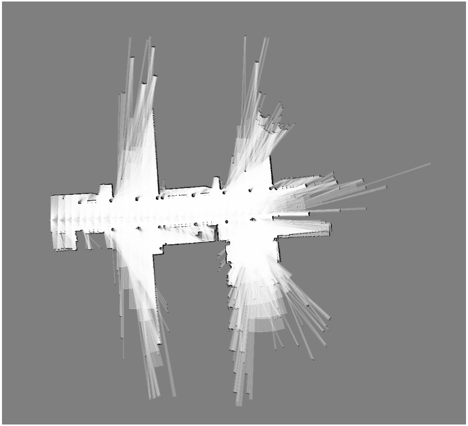
- Rigid transformations only (no scale changes).
- Grids carry very useful information.

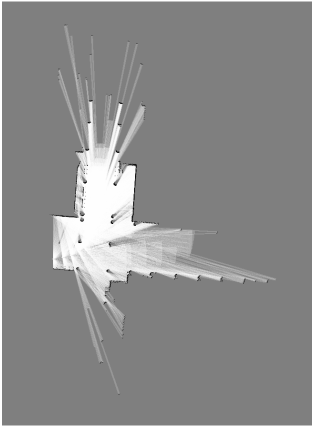
Cons

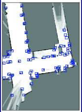
- Ambiguity: large portions of the grids look the same.
- Data-association is not applicable to raw grids.

Introduction

The problem of the ambiguity:







Introduction

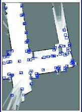
Previous methods

Correlation (costly).

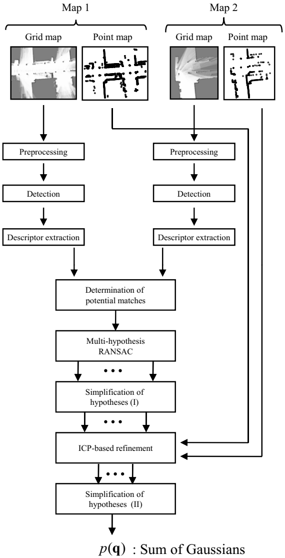
Histograms (requires a robot with a perfect compass).

Our proposal

To employ Computer Vision techniques previously applied to matching images.



Overview



We employ hybrid maps:

grid + points

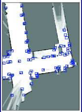
The points are just employed for a final refinement of the matching.

PhD dissertation

Jose Luis Blanco Claraco

Overview

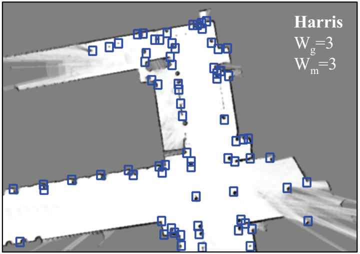
- First steps: Gaussian + median filter, then detect points.
- We have evaluated four detectors:
  - Harris.
  - Kanade-Lucas-Tomasi (KLT).
  - SIFT.
  - SURF.

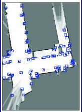


Overview

- First steps: Gaussian + median filter, then detect points.
- We have evaluated four detectors:
  - Harris.
  - Kanade-Lucas-Tomasi (KLT).
  - SIFT.
  - SURF.

Overview



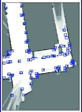


Descriptors

- Each feature is then assigned a descriptor.
- We have evaluated four detectors:
    - Log-polar patch.
    - Linear-polar patch.
    - Spin images (polar histogram of intensities).
    - SIFT.
    - SURF.

Descriptors

- Each feature is then assigned a descriptor.
- We have evaluated four detectors:
  - Log-polar patch.
  - Linear-polar patch.
  - Spin images (polar histogram of intensities).
  - SIFT.
  - SURF.

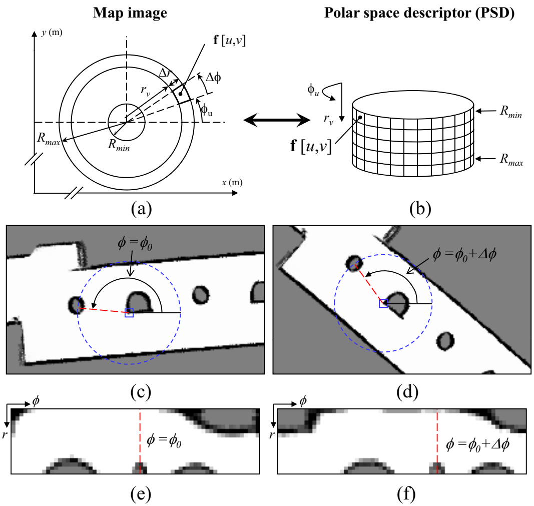


PhD dissertation

Jose Luis Blanco Claraco

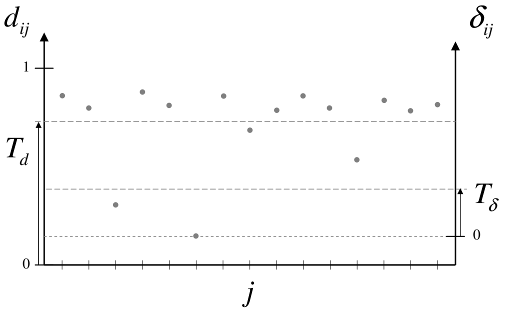
Descriptors

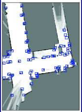
Linear-polar patch:



Descriptors

Then potential pairings are determined by thresholding the inter-feature distances with two thresholds T_δ and T_d:



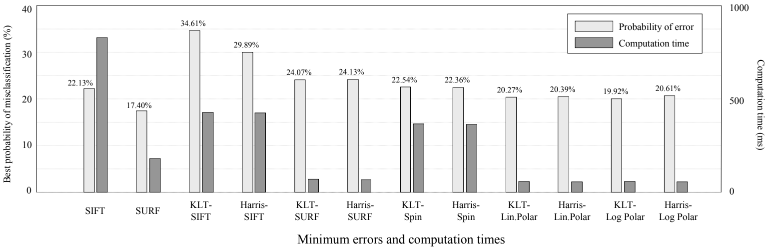


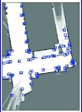
Descriptors

How to pick a combination of detector+descriptor+their parameters?

Descriptors

How to pick a combination of detector+descriptor+their parameters?

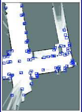




Descriptors

Establishing hypotheses

- A modified version of RANSAC is used.
- Instead of 1 hypothesis → a multitude of them.
- All the tests are probabilistic (chi-square tests).



Descriptors

Establishing hypotheses

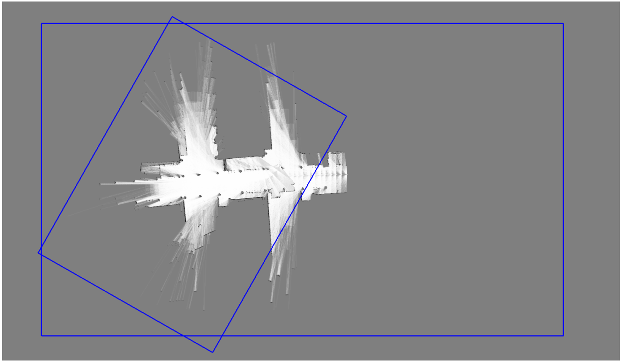
- A modified version of RANSAC is used.
- Instead of 1 hypothesis $\rightarrow$ a multitude of them.
- All the tests are probabilistic (chi-square tests).

Consequences

- The output of the RANSAC step is a SOG.
- Compatible with HMT-SLAM: multiple hypotheses for the arcs between areas ${}^{a}_{b}\Delta$.

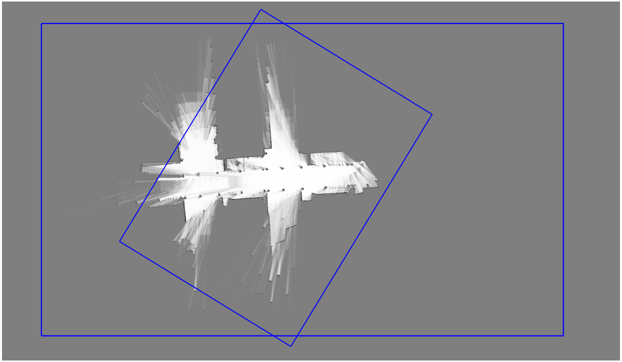
Results: Multiple hypotheses

An example of multiple hypotheses with real sub-maps:



Results: Multiple hypotheses

An example of multiple hypotheses with real sub-maps:



Results: Benchmark

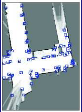
The method has been intensively tested against a testbed of 59 sub-maps obtained from 4 different public datasets:

The datasets:

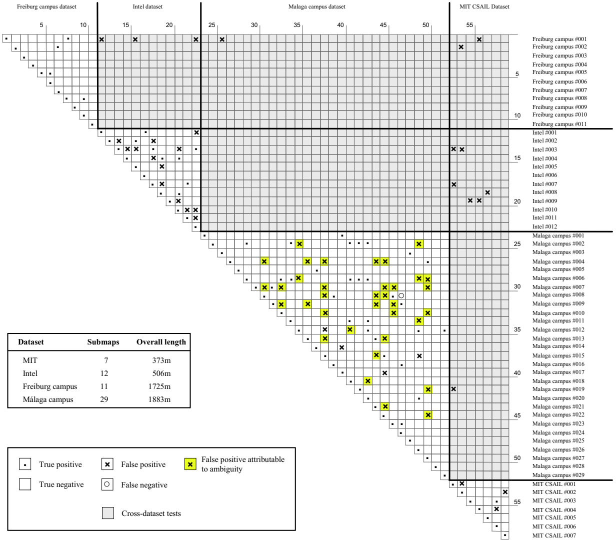
- Intel offices dataset.
- Freiburg campus.
- MIT dataset.
- Málaga campus.

The 3 first ones available at: `http://radish.sf.net/`

The last one available at: `http://mrpt.sf.net/`



Results: Benchmark



Dataset	Submaps	Overall length
MIT	7	373m
Intel	12	506m
Freiburg campus	11	1725m
Málaga campus	29	1883m

True positive | False positive | False positive attributable to ambiguity

True negative | False negative

Cross-dataset tests

Results: Benchmark

Results of the benchmark

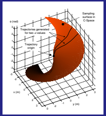
	Result	Disregarding ambiguity
True positives	97.56% (40/41)	·
False positives	3.47% (58/1670)	1.38% (23/1670)
True negatives	96.53% (1612/1670)	98.62% (1647/1670)
False negatives	2.44% (1/41)	·

Papers

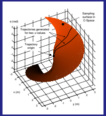
- TRO: *A Robust, Multi-Hypothesis Approach to Matching Occupancy Grid Maps* (Under review).

Part IV

Robot navigation



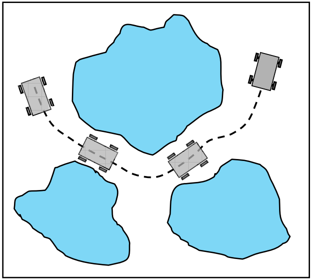
Outline of Part IV



Introduction

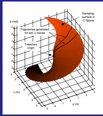
Robot navigation

Problem: Take the robot from point A to point B.



Paradigms

- Planned
- Reactive
- Mixed



Introduction

Why is this complicated?

Two main independent sources of problems:

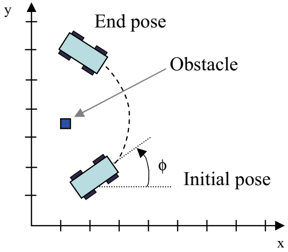
- Kinematic restrictions: Most robots cannot move in any arbitrary direction!
- Robot shape: Colission must be avoided with any part of the robot, which can **turn** $\longrightarrow$ C-Space

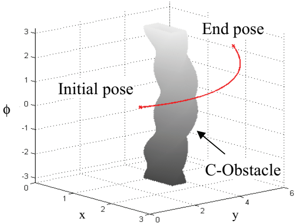
Introduction

Why is this complicated?

Two main independent sources of problems:

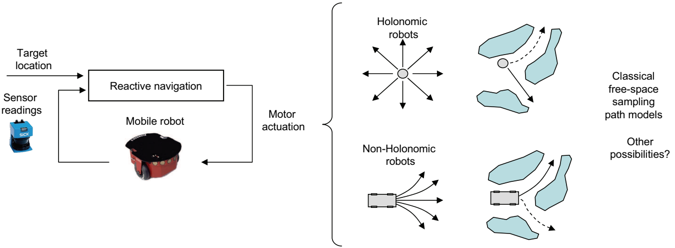
- Kinematic restrictions: Most robots cannot move in any arbitrary direction!
- Robot shape: Colission must be avoided with any part of the robot, which can **turn** → C-Space





Introduction

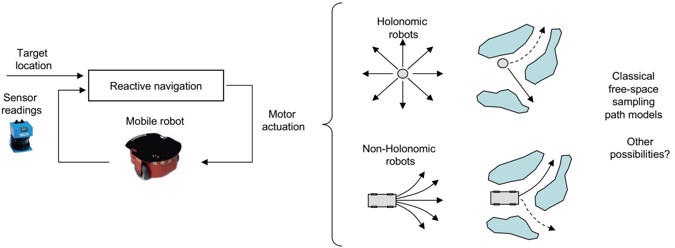
We will focus on **reactive** methods:



How does a robot decide in which direction to move?

Introduction

We will focus on **reactive** methods:

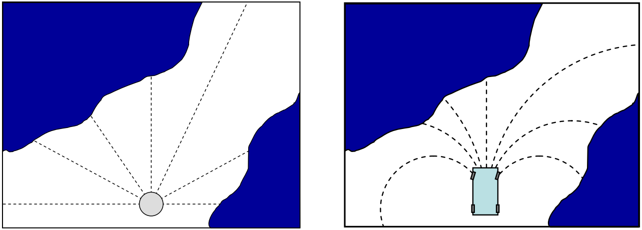


How does a robot decide in which direction to move?

Sampling the obstacles around with some path model

Introduction

"Sampling the obstacles" has been done in all previous works (never put explicitly):



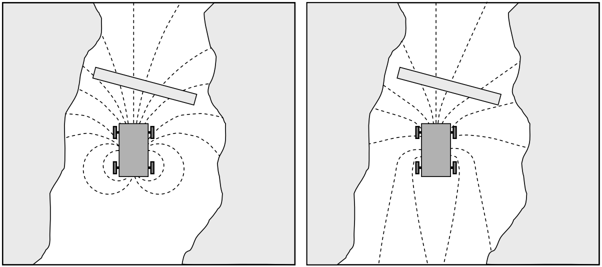
For non-holonomic robots: only circular arcs have been employed.

Introduction

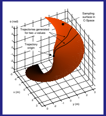
Why considering other models apart from arcs?

Reactive navigation decides just from the free-space it "samples":

We must provide it a **motivation** to start moving into any given direction.



(a) (b)



Introduction

Obstacle avoidance algorithms are either:

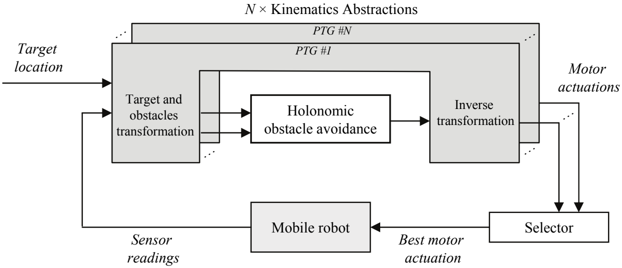
- **For non-holonomic robots:** They assume circular arcs.
- **For holonomic robots:** Well known methods (VFF, Nearness Diagram), but not applicable to many real robots.

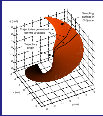
The solution introduced by [Minguez & Montano, 2006]: to abstract the robot shape and kinematics with a *space transformation*.

Introduction

Our contribution:

A **generalization** of this abstraction by means of Parameterized Trajectory Generators (PTGs)



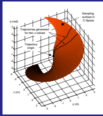


What are PTGs?

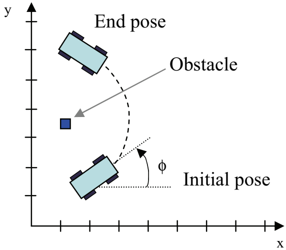
Definition

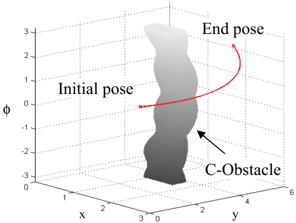
A PTG maps a 2-d space, the Trajectory Parameter (TP) space, into a manifold of the robot C-Space.
An inverse PTG does exactly the opposite operation.

The key idea is reducing the dimensionality to 2-d, then apply "standard" holonomic methods.



What are PTGs?





Planning in C-Space is complicated: robot trajectories are in 3-d !

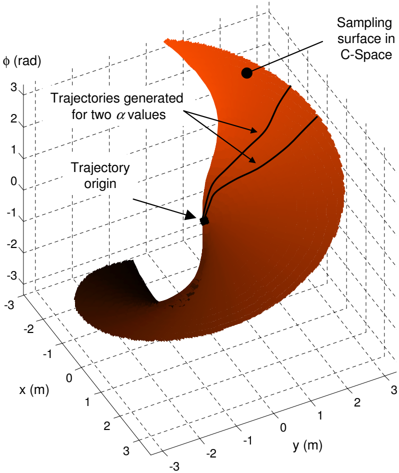
Visualize a family of trajectories and a surface will emerge...

PhD dissertation

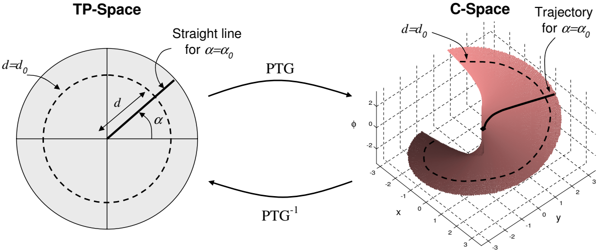
Jose Luis Blanco Claraco

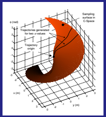
What are PTGs?

It's the PTG sampling surface, and it's parameters are α and d.
(This surface is a manifold of C-Space)



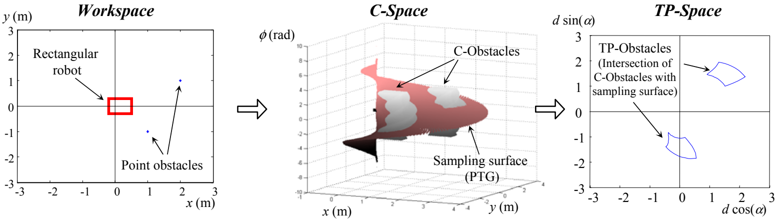
Relation of PTGs with the TP-Space:





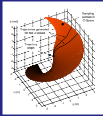
What are PTGs?

The name "sampling surfaces" comes from this:



Theoretical results

Given all these definitions, we have proven that many PTGs are suitable for reactive navigation (not trivial!):



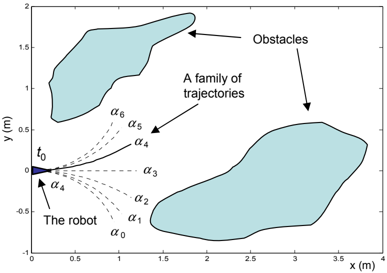
Theoretical results

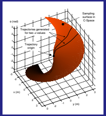
Definition of a **valid** PTG

It must fulfil:

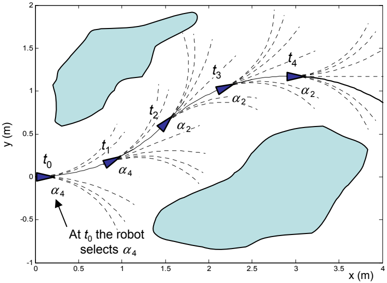
- **C1.** It generates *consistent reactive trajectories* (the memoryless nature of the movement decision process).

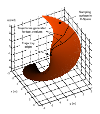
Theoretical results





Theoretical results



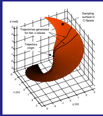


Theoretical results

Definition of a valid PTG

It must fulfil:

- **C1.** It generates *consistent reactive trajectories* (the memoryless nature of the movement decision process).

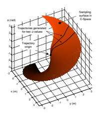


Theoretical results

Definition of a **valid** PTG

It must fulfil:

- **C1.** It generates *consistent reactive trajectories* (the memoryless nature of the movement decision process).
- **C2.** It is *WS-bijective* for each WS location (x, y).

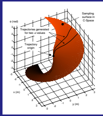


Theoretical results

Definition of a **valid** PTG

It must fulfil:

- **C1.** It generates *consistent reactive trajectories* (the memoryless nature of the movement decision process).
- **C2.** It is *WS-bijective* for each WS location (x, y).
- **C3.** It is *continuous*.



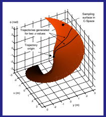
Theoretical results

Definition of a **valid** PTG

It must fulfil:

- **C1.** It generates *consistent reactive trajectories* (the memoryless nature of the movement decision process).
- **C2.** It is *WS-bijective* for each WS location (x, y).
- **C3.** It is *continuous*.

C2 + C3 $\rightarrow$ the PTG will not modify the topology of the real workspace around the robot.



Theoretical results

Theorem

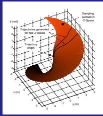
A sufficient, but not necessary condition for a PTG to be valid is its velocity vector $\mathbf{u}$ *being of the form:*

$$\mathbf{u}(\alpha, t) = \begin{bmatrix} v_m \cdot f_v\left(a\alpha + b\phi(\alpha, t)\right) \\ \omega_m \cdot \left(a\alpha + b\phi(\alpha, t)\right) \end{bmatrix}$$

where v_m *and* ω_m *settle the desired maximum linear and angular velocities in absolute value, respectively,* $f_v(\alpha, t)$ *is any Lipschitz continuous function which evaluates to non-zero over the whole domain, and* a, b *are arbitrary constants with the restrictions* $0 < |a/b| \leq 1$ *and* $b < 0$.

Corollary

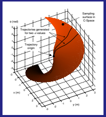
Furthermore, a velocity vector of this form becomes fully defined by just specifying its value for $t = 0$.



Theoretical results

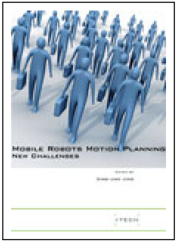
The proofs imply the derivation of 4 auxiliary lemmas. All the details can be found in the thesis.

To remark it again: this is the first time it is proven that reactive navigation is possible with other models apart from circular arcs.



Papers

- **Book chapter**: *Foundations of Parameterized Trajectories-based Space Transformations for Obstacle Avoidance*. (Mobile Robots Motion Planning – New challenges), 2008.



Outline of Part IV

Applications of PTGs

In the last slides we have discussed PTGs as a theoretical tool.

Applications of PTGs

- To reactive navigation.
- To planned or mixed schemes.

Applications of PTGs

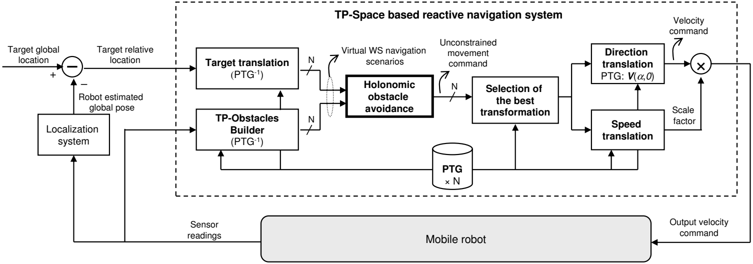
In the last slides we have discussed PTGs as a theoretical tool.

Applications of PTGs

- To reactive navigation.
- To planned or mixed schemes.

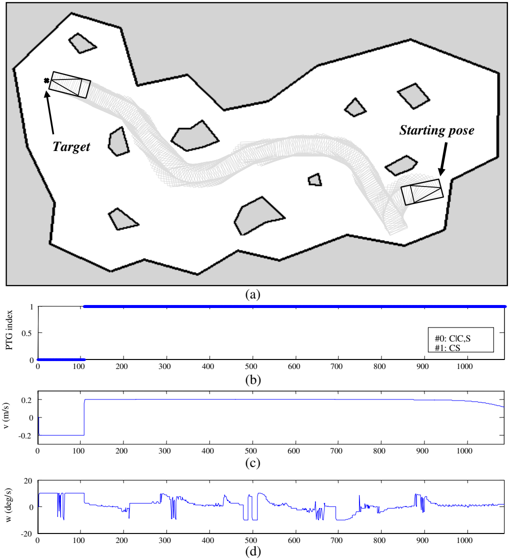
A PTG-based reactive navigator

A system has been proposed to integrate multiple PTGs into a reactive navigator:

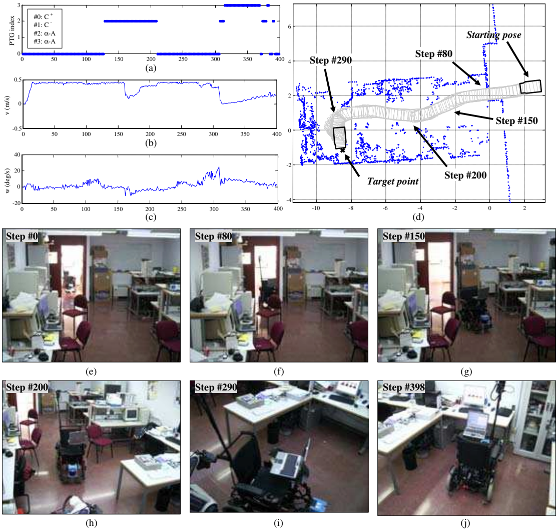


The holonomic navigator running on the TP-Spaces is the *Nearness Diagram* [Minguez & Montano 2004].

Experiment #1



Experiment #2



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

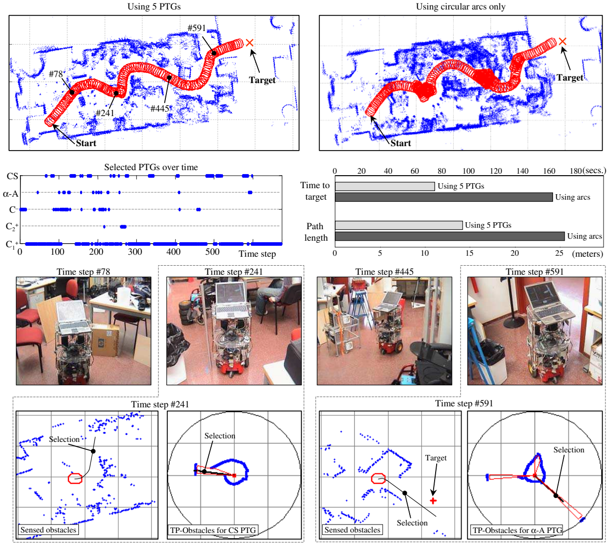
Experiment #2

(sena-nav.avi)

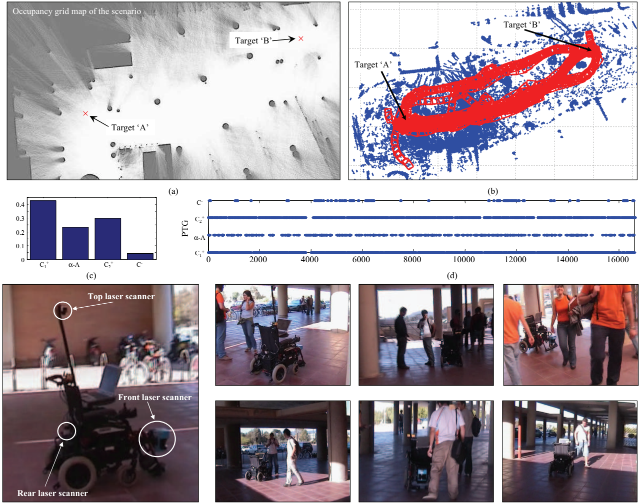
Experiment #3

(sancho-nav.avi)

Experiment #3



Experiment #4



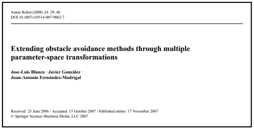
(a) (b) (c) (d) (e) (f)

Experiment #4

(sena-nav-out.avi)

Papers

- AR: *Extending Obstacle Avoidance Methods through Multiple Parameter-Space Transformations*, 2008.


Auton Robot (2008) 24: 29–48
DOI 10.1007/s10514-007-9062-7

Extending obstacle avoidance methods through multiple parameter-space transformations

Jose-Luis Blanco · Javier González · Juan-Antonio Fernández-Madrigal

Received: 23 June 2006 / Accepted: 15 October 2007 / Published online: 17 November 2007
© Springer Science+Business Media, LLC 2007

- IROS 2006 (Beijing, China): *The Trajectory Parameter Space (TP-Space): A New Space Representation for Non-Holonomic Mobile Robot Reactive Navigation.*

Part V

Conclusions

Thesis conclusions

This thesis comprises several important contributions to the fields of mobile robotics and estimation theory:

- First unified Bayesian approach to hybrid SLAM.
- Optimal particle filtering $\rightarrow$ localization & SLAM.
- Introduction of PTGs $\rightarrow$ reactive navigation.
- Efficient solution to RO-SLAM with SOG.
- New information metrics for RBPF-SLAM & exploration.
- Range scan likelihood fusion via Consensus theory.
- New probabilistic, multi-hypotheses grid-to-grid matching.
- Grounded method for partitioning sub-maps.
- Fusion odometry $+$ IMU.

Paper references

Some citation statistics according to Google Scholar (2009/NOV/1):

- TRO 2008: HMT-SLAM $\rightarrow$ 14 citations.
- ICRA 2006: Observation clustering $\rightarrow$ 10 citations.
- ICRA 2007: HMT-SLAM $\rightarrow$ 4 citations.
- The rest of papers: average of $\sim$ 2 citations.

(Not counting self-references)

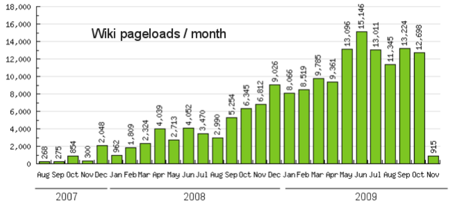
MRPT statistics

The MRPT is an open-source initiative to release the works in this thesis, among others, as a coherent C++ programming framework.

MRPT statistics

The MRPT is an open-source initiative to release the works in this thesis, among others, as a coherent C++ programming framework.

- It's been integrated into major GNU/Linux repositories (Debian, Ubuntu, Fedora).
- Integrated into our group's **BABEL** development framework.
- >1300 source files, > 610,000 lines of code.
- ∼20 downloads/day during 2009.
- Web traffic:



Contributions to Localization, Mapping and Navigation in Mobile Robotics

PhD dissertation

Jose Luis Blanco Claraco

Advisors: J. González, J.A. Fernández-Madrigal

UNIVERSIDAD DE MÁLAGA

University of Málaga

November 13th, 2009