Mobile Robot Simultaneous Localization and Mapping
Using DP-SLAM with a Single Laser Range Finder

Dissertation presented to the Postgraduate Program in Mechanical Engineering of the Departamento de Engenharia Mecânica, PUC-Rio, as partial fulfillment of the requirements for the degree of Mestre em Engenharia Mecânica.

Advisor: Prof. Marco Antonio Meggiolaro

Rio de Janeiro
April 2011
Luis Ernesto Ynoquio Herrera

Mobile Robot Simultaneous Localization and Mapping
Using DP-SLAM with a Single Laser Range Finder

Dissertation presented to the Postgraduate Program in Mechanical Engineering of the Departamento de Engenharia Mecânica do Centro Técnico Científico da PUC-Rio, as partial fulfillment of the requirements for the degree of Mestre em Engenharia Mecânica.

Prof. Marco Antonio Meggiolaro
Advisor
Departamento de Engenharia Mecânica – PUC-Rio

Prof. Karla Tereza Figueiredo Leite
Departamento de Engenharia Elétrica – PUC-Rio

Prof. Liu Hsu
Departamento de Engenharia Elétrica – UFRJ

Prof. José Eugenio Leal
Coordenador Setorial
Centro Técnico Científico – PUC-Rio

Rio de Janeiro, 7 de abril de 2011
Luis Ernesto Ynoquio Herrera

Graduated in Electronics Engineering at Universidad Privada Antenor Orrego of Perú in 2002 (Trujillo - Perú).

Bibliographic data

Ynoquio Herrera, Luis Ernesto


167 f.: il. (color.) ; 30 cm

Dissertação (mestrado) – Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Mecânica, 2011.

Inclui bibliografia.


CDD: 621
Thanks to

My advisor Marco Antonio Meggiolaro, for friendship, patience, help and support that motivated me to perform this work.

PUC-Rio for the opportunity and the great academic environment.

CNPQ for the financial support, without which it would not have been possible to do this work.

My parents, Eumenia and Gilber, and my siblings, who however distant, always support me.

My friend Juan Gerardo Castillo Alva, who facilitated the opportunity to come to Brasil and for his wise counsel, thanks.
Abstract

Simultaneous Localization and Mapping (SLAM) is one of the most widely researched areas of Robotics. It addresses the mobile robot problem of generating a map without prior knowledge of the environment, while keeping track of its position. Although technology offers increasingly accurate position sensors, even small measurement errors can accumulate and compromise the localization accuracy. This becomes evident when programming a robot to return to its original position after traveling a long distance, based only on its sensor readings. Thus, to improve SLAM’s performance it is necessary to represent its formulation using probability theory. The Extended Kalman Filter SLAM (EKF-SLAM) is a basic solution and, despite its shortcomings, it is by far the most popular technique. Fast SLAM, on the other hand, solves some limitations of the EKF-SLAM using an instance of the Rao-Blackwellized particle filter. Another successful solution is to use the DP-SLAM approach, which uses a grid representation and a hierarchical algorithm to build accurate 2D maps. All SLAM solutions require two types of sensor information: odometry and range measurement. Laser Range Finders (LRF) are popular range measurement sensors and, because of their accuracy, are well suited for odometry error correction. Furthermore, the odometer may even be eliminated from the system if multiple consecutive LRF scans are matched. This works presents a detailed implementation of these three SLAM solutions, focused on structured indoor environments. The implementation is able to map 2D environments, as well as 3D environments with planar terrain, such as in a typical indoor application. The 2D application is able to automatically generate a stochastic grid map. On the other hand, the 3D problem uses a point cloud representation of the map, instead of a 3D grid, to reduce the SLAM computational effort. The considered mobile robot only uses a single LRF, without any odometry information. A Genetic Algorithm is presented to optimize the matching of LRF scans taken at different instants. Such matching is able not only to map the environment but also localize the robot, without the need for odometers or other sensors. A simulation program is implemented in Matlab® to generate virtual LRF readings of a mobile robot in a 3D environment. Both simulated readings and experimental data from the literature are independently used to validate the proposed methodology, automatically generating 3D maps using just a single LRF.
Key Words

Mobile Robots, Bayesian Filter, Scan Matching, Simultaneous Localization and Mapping, Laser Range Finder.
Resumo


SLAM (Mapeamento e Localização Simultânea) é uma das áreas mais pesquisadas na Robótica móvel. Trata-se do problema, num robô móvel, de construir um mapa sem conhecimento prévio do ambiente e ao mesmo tempo manter a sua localização nele. Embora a tecnologia ofereça sensores cada vez mais precisos, pequenos erros na medição são acumulados comprometendo a precisão na localização, sendo estes evidentes quando o robô retorna a uma posição inicial depois de percorrer um longo caminho. Assim, para melhoria do desempenho do SLAM é necessário representar a sua formulação usando teoria das probabilidades. O SLAM com Filtro Extendido de Kalman (EKF-SLAM) é uma solução básica, e apesar de suas limitações é a técnica mais popular. O Fast SLAM, por outro lado, resolve algumas limitações do EKF-SLAM usando uma instância do filtro de partículas conhecida como Rao-Blackwellized. Outra solução bem sucedida é o DP-SLAM, o qual usa uma representação do mapa em forma de grade de ocupação, com um algoritmo hierárquico que constrói mapas 2D bastante precisos. Todos estes algoritmos usam informação de dois tipos de sensores: odômetros e sensores de distância. O Laser Range Finder (LRF) é um medidor laser de distância por varredura, e pela sua precisão é bastante usado na correção do erro em odômetros. Este trabalho apresenta uma detalhada implementação destas três soluções para o SLAM, focalizado em ambientes fechados e estruturados. Apresenta-se a construção de mapas 2D e 3D em terrenos planos tais como em aplicações típicas de ambientes fechados. A representação dos mapas 2D é feita na forma de grade de ocupação. Por outro lado, a representação dos mapas 3D é feita na forma de nuvem de pontos ao invés de grade, para reduzir o custo computacional. É considerado um robô móvel equipado com apenas um LRF, sem nenhuma informação de odometria. O alinhamento entre varreduras laser é otimizado fazendo o uso de Algoritmos Genéticos. Assim, podem-se construir mapas e ao mesmo tempo localizar o robô sem necessidade de odômetros ou outros sensores. Um simulador em Matlab® é implementado para a geração de varreduras virtuais de um LRF em um ambiente 3D (virtual). A metodologia proposta é validada com os dados simulados, assim como com dados experimentais obtidos da literatura, demonstrando a possibilidade de construção de mapas 3D com apenas um sensor LRF.
Palavras-Chave

# Summary

## 1 Introduction and Problem Definition

1.1. Introduction 22

1.1.1. Robotics 22

1.1.2. Uncertainty in Robotics 23

1.2. Problem Definition 23

1.2.1. Localization overview 24

1.2.2. Mapping overview 26

1.2.3. Simultaneous Localization and Mapping 27

1.3. Motivation 28

1.4. Objective 29

1.5. Organization of the Thesis 30

## 2 Theoretical Basis

2.1. Probabilistic Robotics 31

2.1.1. Bayes Filter and SLAM 32

2.1.2. Motion Model 35

2.1.3. Perception Model 38

2.2. Map Representation 40

2.2.1. Landmark Maps 40

2.2.2. Grid Maps 42

2.3. Scan Matching 43

2.3.1. Point to Point Correspondence Methods. 44

2.3.2. Feature to Feature Correspondence Methods. 45

2.3.3. The Normal Distribution Transform 47

2.4. Genetic Algorithms 51

2.4.1. Chromosome Representation 52
2.4.2. The Fitness Function  52
2.4.3. Fundamental Operators  53
2.4.4. Genetic Algorithms to Solve Problems  54
2.4.5. Differential Evolution  54
2.4.6. Different Strategies of DE  56

3. SLAM Solutions  58
3.1. Gaussian Filter SLAM Solutions  58
3.1.1. Kalman Filter SLAM  58
3.1.2. Extended Kalman Filter SLAM  63
3.2. Particle Filter SLAM Solutions  65
3.2.1. Particle Filter Overview  65
3.2.2. Fast SLAM  69
3.2.3. DP-SLAM  73
3.3. 3D SLAM Review  86

4. Detailed Implementation  88
4.1. EKF SLAM  88
4.2. FastSLAM  95
4.3. Simulator  103
4.3.1. 3D Environment Simulation  103
4.3.2. LRF Simulation  104
4.3.3. Error introduction in virtual data  108
4.4. Scan Matching  109
4.4.1. Differential Evolution Optimization for NDT  110
4.4.2. Parameters and Considerations  113
4.4.3. Scan Filtering  115
4.5. DP-SLAM  117
4.5.1. Motion Model  120
4.5.2. High Motion Model  124
5. Tests and results

5.1. Scan Matching
5.1.1. Optimization Parameters Influence
5.1.2. LRF Error Influence
5.1.3. Scan Matching in Real Data

5.2. Motion Model
5.3. DP-SLAM
5.4. 3D Mapping

6. Conclusions

6.1. DP-SLAM Conclusions
6.2. Scan Matching Conclusions
6.3. 3D Mapping Conclusions

7. References
List of figures

Figure 1.1: Localization Overview (search for landmarks) ......................... 25
Figure 1.2: Localization Overview (location updated) ............................. 25
Figure 1.3: Mapping Overview ............................................................. 26
Figure 1.4: Simultaneous Localization and Mapping ............................... 27
Figure 2.1: SLAM like a Dynamic Bayes Network ................................. 35
Figure 2.2: Robot pose ....................................................................... 36
Figure 2.3: The motion model, showing posterior distributions of the robot’s pose upon executing the motion command illustrated by the red striped line. The darker a location, the more likely it is. ................................................................. 37
Figure 2.4: Robot in a map getting measurements from its LRF. ................. 39
Figure 2.5: Given an actual measurement and an expected distance, the probability of getting that measurement is given by the red line in the graph ............................................................... 39
Figure 2.6: Simulated Landmark Map .................................................. 41
Figure 2.7: Grid Map: White regions mean unknown areas, light gray unoccupied areas, and darker gray to black represent increasingly occupied areas [8]. ........................................................................ 43
Figure 2.8: An example of NTD: the original laser scan (left) and the resulting probability density (right) ......................................................... 48
Figure 2.9: The crossover operator ....................................................... 53
Figure 2.10: The mutation operator ...................................................... 54
Figure 2.11: Differential Evolution Process[22] ..................................... 56
Figure 3.1: Representation of a Gaussian by a set of particles ................. 66
Figure 3.2: Particle Filter idea ............................................................. 68
Figure 3.3: Landmark correlation ......................................................... 70
Figure 3.4: Occupancy grid prediction based on a movement of one cell to the right.

Figure 3.5: Square representation.

Figure 3.6: Interaction between the laser ray and the square representation.

Figure 3.7: Example of application of eq. (3.37) for a given square $j=3$.

Figure 3.8: Distance between the square $i$ and the stopping point of the laser ray.

Figure 3.9: Example of computing the probability of a laser ray given two sampled robot poses.

Figure 3.10: Example of particle ancestry tree maintenance.

Figure 3.11: Simulated environment (60 x 40 m).

Figure 3.12: Mapping closing a loop. Each black dot is the perturbed endpoints of trajectories.

Figure 3.13: Map after ambiguities are resolved.

Figure 4.1: Robot position (red fill circle) at time $t-1$, and predicted robot position (white filled circle).

Figure 4.2: Predicted landmark position seen from the predicted robot position.

Figure 4.3: New landmark $L_{\text{pn}}$ is added to the state vector.

Figure 4.4: Sampling. Each black dot represents a possible robot position.

Figure 4.5: Distance and rotation between particle $i$ and landmark $j$.

Figure 4.6: Updated landmark $j$ for the particle $i$.

Figure 4.7: Distance and rotation between particle $i$ and updated landmark $j$.

Figure 4.8: Cumulative probability distribution and a random number $r$.

Figure 4.9: New observed landmark $L_q$.

Figure 4.10: Simulated structured environment using rectangles.

Figure 4.11: Laser ray from a simulated LRF.
Figure 4.12: Simulated LRF rotation to acquire 3D data .................106

Figure 4.13: Virtual LRF readings in a simulated environment (top view) ........................................................................................................107

Figure 4.14: Virtual LRF readings in a simulated environment, from two different points of view. .............................................................108

Figure 4.15: Density regions produced by a robot situated close to the right wall. ...................................................................................115

Figure 4.16: Mismatched scans, showing the NDT of a first scan (grayscale) and a second scan (red dots). The right-bottom wall produces a high number of readings, bringing down the second scan and compromising the match. ....................................116

Figure 4.17: Scan filtering. Original scan with 181 points (left) and filtered scan with 59 points (right) ...........................................................117

Figure 4.18: DP-SLAM flow chart ................................................................118

Figure 4.19: Low (level) DP-SLAM flow chart ........................................119

Figure 4.20: Two consecutive robot positions in an environment ..........121

Figure 4.21: Environment seen from the current robot position (left) and second robot position (right) ..........................................................122

Figure 4.22: Aligned scans in the first scan coordinate system .............122

Figure 4.23: High Motion Model ................................................................125

Figure 5.1: Error in displacement, $\Delta x$, $\Delta y$, $\Delta \theta$, influenced by population size (20 and 40) in DE optimization ..........................................................127

Figure 5.2: Error in displacement, $\Delta x$, $\Delta y$, $\Delta \theta$, influenced by population size (60 and 100) in DE optimization ..........................................................127

Figure 5.3: Error in displacement, $\Delta x$, $\Delta y$, $\Delta \theta$, influenced by number of generations (15 and 30) in DE optimization ..........................................................128

Figure 5.4: Error in displacement, $\Delta x$, $\Delta y$, $\Delta \theta$, influenced by number of generations (50 and 75) in DE optimization ..........................................................128

Figure 5.5: Displacement error, $\Delta x$, $\Delta y$, $\Delta \theta$, influenced by LRF error (stdv= 15mm and stdv=25mm) in DE optimization ..........................................................129

Figure 5.6: Displacement error, $\Delta x$, $\Delta y$, $\Delta \theta$, influenced by LRF error (stdv= 40mm and stdv=80mm) in DE optimization ..........................................................129
Figure 5.7: Accumulated error in robot position due to imperfect
scan matching. ................................................................................ 130

Figure 5.8: Robot trajectories, showing the true path (red line) and
estimated path (blue line). a) Robot starts from zero position
(red dot), goes through positions A, B, C and A. b) Robot
traveling A, D, B. c) Robot completes the course through B, A,
C, B and D. .......................................................................................... 131

Figure 5.9: Map acquired using scan matching process in the
simulated environment. .................................................................. 132

Figure 5.10: Fitness values for “D-Wing” experiment. ..................... 133

Figure 5.11: “D-Wing” experiment acquired using the proposed
scan matching process, without the use of odometry data. .......... 134

Figure 5.12: Fitness values for “C-Wing” experiment .................... 135

Figure 5.13: “C-Wing” experiment acquired using the proposed
scan matching process, without the use of odometry data. .......... 136

Figure 5.14: Fitness values for “Diiga” experiment. ....................... 137

Figure 5.15: “Diiga” experiment acquired using the proposed scan
matching process, without the use of odometry data. ............... 138

Figure 5.16: Fitness values for “Mine” experiment ....................... 139

Figure 5.17: Poor map of “Mina” experiment. ........................... 139

Figure 5.18: “Mina” experiment acquired using scan matching
process .............................................................................................. 141

Figure 5.19: Error distribution in displacement: a) \( \Delta x \), b) \( \Delta \theta \) and
c) \( \Delta y \) .......................................................................................... 142

Figure 5.20: Misalignment in \( \Delta y \) (respect to the current robot
position) .......................................................................................... 143

Figure 5.21: a) The true robot displacement. b) The displacement
given by scan matching, showing that the most common error
is in \( \Delta y \). .................................................................................. 144

Figure 5.22: Misalignment in \( \Delta y \) (D-Wing experiment) ............. 145

Figure 5.23: Larger displacement in \( \Delta y \) correspond to small
rotations \( \Delta \theta \) (D-Wing experiment) ............................................. 146
Figure 5.24: 2D grid map from the simulated environment experiment, obtained with DP-SLAM using the proposed motion model.................................148

Figure 5.25: Accumulated error position obtained with DP-SLAM using the proposed motion model, on simulated experiment. ........148

Figure 5.26: Error distribution in $\Delta x$ obtained with DP-SLAM using the proposed motion model............................................................149

Figure 5.27: Error distribution in $\Delta y$ obtained with DP-SLAM using the proposed motion model............................................................149

Figure 5.28: Error distribution in $\Delta \theta$ obtained with DP-SLAM using the proposed motion model............................................................150

Figure 5.29: 2D grid map from simulated environment presented in Figure 3.11.....................................................................................150

Figure 5.30: 2D grid map of “D-wing” experiment, acquired with DP-SLAM using the proposed motion model.................................151

Figure 5.31: 2D grid map of “C-wing” experiment, acquired with DP-SLAM using the proposed motion model.........................................152

Figure 5.32: 2D grid map of “Diiga” experiment, acquired with DP-SLAM using the proposed motion model.........................................153

Figure 5.33: 2D grid map of “Mine” experiment, acquired with DP-SLAM using the proposed motion model.........................................154

Figure 5.34: 3D point cloud map of the simulated environment..............155

Figure 5.35: 3D point cloud of the simulated environment (only three 3D scans are shown) .............................................................156

Figure 5.36: 3D point cloud of the simulated environment (only four 3D scans are shown).................................................................156

Figure 5.37: 3D point cloud of the “Mine” experiment............................157

Figure 5.38: 3D point cloud of the “Mine” experiment............................158

Figure 5.39: 3D point cloud of the “Mine” experiment (top view). .........158

Figure 5.40: 3D point cloud of the “Mine” experiment............................159
List of tables

Table 3.1: The Kalman Filter Algorithm [1]. .............................................. 61
Table 3.2: The EKF Algorithm [1] ............................................................. 65
Table 3.3: Particle Filter Algorithm [1] ...................................................... 67
Table 4.1: Sample Motion Model Algorithm [1]................................. 96
Table 4.2: Compute weights algorithm; .................................................. 100
Table 4.3: Adjustable parameters on simulated LRF ......................... 105
Table 4.4: Search space for vector $p$ ...................................................... 112
Table 4.5: DE Optimization Parameters. .............................................. 113
Table 4.6: SICK LMS-200-3016 features. .............................................. 114
Table 4.7: Sample scan matching motion model algorithm, where
$\text{atan2}(\Delta y, \Delta x)$ is defined as the generalization of the arc-tangent
function of $\Delta y/\Delta x$ over $[0, 2\pi]$. .................................................. 123
Table 5.1: Proposed Scan Matching Motion Model. ............................ 146
Table 5.2: Approximate algorithm to sampling from normal
distribution [1].................................................................................. 147
Table 5.3: Approximated Algorithm to sample from the $\Delta y$ distribution
error. .................................................................................................... 147
List of Variables

\( A_t \): State transition matrix

\( B_t \): Matrix that translates control input into a predicted change in state

\( d \): Measured data

\( f \): Function that represents the motion model in EKF-SLAM

\( h \): Function that represents the perception model in EKF-SLAM

\( h_i^j \): Function that represents the perception model for the particle \( i \)

\( H_i^j \): The Jacobian of \( h_i^j \) at \( L_j \)

\( H_t \): The Jacobian of \( h \) at \( \overline{L}_i \)

\( J_f u_t \): The Jacobian of \( f \) at \( u_t \)

\( K_t \): Kalman gain

\( L \): Set of landmarks with known exact location

\( L_{n.t}^i \): Position of landmark \( n \), related with particle \( i \), at time \( t \)

\( L_{t1} L_{t2} \ldots L_{tn} \): n-th landmark estimated at time \( t \)

\( L_q \): New observed landmark

\( \eta \): Normalizer

\( p \): Vector of the parameters to estimate (in DE)

\( p_c \): Probability of crossover

\( p_m \): Probability of mutation

\( P_t \): Covariance of the process noise
\( P_r \): Reference robot position

\( P_n \): New robot position

\( Q \): Covariance matrix for the sensor noise in EKF-SLAM

\( R_\theta, R_1, \ldots R_t \): Robot position at time \( t \)

\( Rx, Ry, R\theta \): Robot position in two-dimensional planar coordinates

\( R^{i,t} \): Robot path, related with particle \( i \), until time \( t \)

\( S_{new} \): New scan

\( S_{ref} \): Reference scan

\( t_x \): Translation in \( x \)

\( t_y \): Translation in \( y \)

\( U \): Uncertainty of the control \( u_t \) in EKF-SLAM

\( u_t \): Control at time \( t \)

\( w_i^t \): Weight of particle \( i \) at time \( t \)

\( x_i^t \): Particle \( i \) at time \( t \)

\( Z_0, Z_1, Z_t \): Map estimated at time \( t \)

\( x_i \): Distance that the laser ray travels through the square \( i \)

\( x_i^t \): State variable at time \( t \)

\( z_i \): Sensor measurement at time \( t \)

\( z_{j,t} \): The \( j^{th} \) landmark sensor observation in \( z_t \)

\( \Delta x, \Delta y, \Delta \theta \): Displacements that are referenced to the current robot position

\( \lambda_i \): Gaussian mean at time \( t \)

\( \lambda^i_{a,t} \): Mean related with the landmark position \( L^i_{a,t} \)

\( \rho_i \): Opacity of the square \( i \)
$\Sigma_t$: Gaussian covariance at time $t$

$\Sigma_t, \bar{X}_t$: Predicted covariance and mean at time $t$

$\Sigma_{nt}^i$: Covariance related with the landmark position $L_{nt}^i$

$\phi$: Rotation in $z$

$\Phi_t$: Set of particles at time $t$
List of Abbreviations

SLAM : Simultaneous Localization and Mapping
LRF : Laser Range Finder
GPS: Global Positioning System
KF : Kalman Filter
PF : Particle Filter
EKF-SLAM : Extended Kalma Filter SLAM
FastSLAM : Fast SLAM
DP-SLAM : Distributed Particle SLAM
ICP : Iterative Closest Point
IDC : Iterative Dual Correspondence
ICL : Iterative Closest Line
HAYAI : The Highspeed and Yet Accurate Indoor/outdoor-tracking
NDT : Normal Distributed Transform
GA : Genetic Algorithm
GP : Genetic Programming
DE : Differential Evolution
LSLAM : Low SLAM
HSLAM: High SLAM
Stdv : Standard Deviation